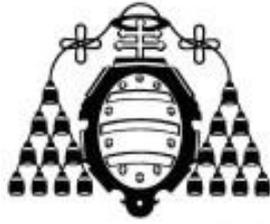


UNIVERSIDAD DE OVIEDO



ESCUELA DE INGENIERÍA INFORMÁTICA

TRABAJO FIN DE GRADO

“Copernicus: Plataforma de comunicaciones”

Vº Bº del Director del
Proyecto

DIRECTOR: Vicente García Díaz

DIRECTOR: Cristian González García

AUTOR: Carlos Villa Blanco

Agradecimientos

En primer lugar, quisiera agradecer a mis padres todo lo que han hecho por mí para que hoy esté terminando esta carrera. Ellos son los que me han dado todas las posibilidades para lograr esto. También a mis hermanos por toda la ayuda y apoyo que me han dado durante todos estos años.

No puedo olvidar tampoco al resto de mi familia. En especial a mis abuelos por sus cuidados desde que era un niño y a mi tía y madrina por su ayuda desde que me mudé a Oviedo.

Agradecer también a aquellos que me apoyaron durante los momentos más difíciles de este proyecto, especialmente a Elena.

Este trabajo no hubiese sido posible sin la ayuda y consejos de los directores de este proyecto. Ellos son los que han hecho posible el buen devenir de este trabajo.

Finalmente, también quiero agradecer a mis compañeros y profesores que me he encontrado durante toda la carrera. Ellos, directa o indirectamente, han influenciado sobre el camino que he tomado en mis estudios.

Resumen

El famoso astrónomo Nicolás Copérnico dedicó gran parte de su trabajo al desarrollo del modelo heliocéntrico, el cual explica que los planetas giran alrededor del sol y que este se encuentra en el centro del universo. De esta premisa proviene el nombre “Copernicus”, una plataforma web de comunicaciones cuyo principal objetivo es ser el centro de todas las comunicaciones de sus usuarios.

La rápida evolución de las tecnologías web y la reducción de las barreras de acceso a internet han permitido en los últimos años la aparición de nuevas formas de trabajo y de ocio a través de comunicaciones remotas. Sin embargo, la mayoría de las aplicaciones que lo hacen posible solo se centran en proporcionar pocas funcionalidades. Copernicus busca ser una plataforma web que se pueda ampliar fácilmente con nuevos servicios, pero que no implique una saturación de sus interfaces. Una plataforma que permita a sus usuarios decidir qué quieren utilizar en cada momento y cómo lo quieren utilizar.

El objetivo es crear una plataforma web que mejore las comunicaciones entre personas y ofrezca en una sola interfaz varios servicios (videollamada, chat de texto, dibujos colaborativos...) de la forma más eficiente posible. Para ello se ha diseñado una interfaz a la que se ha denominado “Modular” que le permite al usuario configurarla añadiendo y eliminando servicios. Con esto se pretende mejorar la productividad y el tiempo de ocio de sus usuarios.

Los servicios proporcionados por Copernicus son:

- Chat de texto
- Chat de video
- Dibujos colaborativos
- Videos compartidos
- Radio/Música compartida
- Presentaciones compartidas

Actualmente estos son los servicios proporcionados, pero, gracias a esta interfaz modular, terceros desarrolladores podrían incluir sus propios servicios.

Palabras Clave

Comunicaciones, videollamadas, redes sociales, Node.js, WebRTC, API REST, WebSockets

Abstract

The famous astronomer Nicolaus Copernicus devoted part of his work to develop the heliocentric model. This model explains that the planets orbit around the Sun and the latter is in the center of the universe. From this premise was created “Copernicus”, a web platform whose main objective is being the center of all the communications of its users.

The rapid evolution of the web technologies and the reduction of the internet access barriers allowed in the recent years the appearance of new ways of working and leisure through remote communications. However, most of the applications that make this possible are only focused on providing a few functionalities. Copernicus wants to be a web platform that can be easily extended with new services, but avoiding a saturation of its interfaces. A platform that allows its users to decide what services they want to use and how they want to use them.

The goal is to create a web platform that improves the communications between people and displays in a single interface several services (Video chat, text chat, collaborative drawings...) in the most efficient way. To achieve this, it was created an interface called “Modular” that allows the user to personalize it by adding and removing services. This wants to improve the productivity and leisure time of its users.

The services provided by Copernicus are:

- Text chat
- Video chat
- Collaborative drawings
- Shared videos
- Shared radio/music
- Shared presentations

Currently these are all the provided services, but, thanks to the modular interface, third-party developers could include their own services.

Keywords

Communications, video calls, social networks, Node.js, WebRTC, REST API, WebSockets

Índice General

CAPÍTULO 1. MEMORIA DEL PROYECTO	25
1.1 RESUMEN DE LA MOTIVACIÓN, OBJETIVOS Y ALCANCE DEL PROYECTO	25
1.2 RESUMEN DE TODOS LOS ASPECTOS	26
CAPÍTULO 2. INTRODUCCIÓN	27
2.1 JUSTIFICACIÓN DEL PROYECTO	27
2.2 OBJETIVOS DEL PROYECTO	27
2.3 ESTUDIO DE LA SITUACIÓN ACTUAL	28
2.3.1 <i>Aplicaciones estudiadas</i>	28
2.3.2 <i>Tabla comparativa</i>	33
2.3.3 <i>Evaluación de Alternativas</i>	33
CAPÍTULO 3. ASPECTOS TEÓRICOS	38
3.1 NODE.JS.....	38
3.1.1 <i>Descripción</i>	38
3.1.2 <i>Justificación de su uso</i>	38
3.1.3 <i>Fuentes</i>	39
3.2 ANGULARJS	39
3.2.1 <i>Descripción</i>	39
3.2.2 <i>Justificación de su uso</i>	39
3.2.3 <i>Fuentes</i>	40
3.3 EXPRESS.....	40
3.3.1 <i>Descripción</i>	40
3.3.2 <i>Justificación de su uso</i>	40
3.3.3 <i>Fuentes</i>	40
3.4 BASE DE DATOS ORIENTADAS A GRAFOS	41
3.4.1 <i>Descripción</i>	41
3.4.2 <i>Justificación de su uso</i>	41

3.4.3	Fuentes	41
3.5	WEBSOCKETS.....	42
3.5.1	Descripción	42
3.5.2	Justificación de su uso.....	42
3.5.3	Fuentes	43
3.6	WEBRTC	43
3.6.1	Descripción	43
3.6.2	Justificación de su uso.....	43
3.6.3	Fuentes	43
3.7	EJS (EMBEDDED JAVASCRIPT TEMPLATES)	44
3.7.1	Descripción	44
3.7.2	Justificación de su uso.....	44
3.7.3	Fuentes	45
3.8	CSS (CASCADING STYLESHEETS).....	45
3.8.1	Descripción	45
3.8.2	Justificación de su uso.....	45
3.8.3	Fuentes	46
3.9	API REST	46
3.9.1	Descripción	46
3.9.2	Justificación de su uso.....	46
3.9.3	Fuentes	47
3.10	INTELLIJ IDEA ULTIMATE EDITION.....	47
3.10.1	Descripción	47
3.10.2	Justificación de su uso	47
3.10.3	Fuentes.....	48
3.11	BOOTSTRAP	48
3.11.1	Descripción	48
3.11.2	Justificación de su uso	49

3.11.3	Fuentes	49
3.12	GRIDSTACK.JS.....	49
3.12.1	Descripción	49
3.12.2	Justificación de su uso	50
3.12.3	Fuentes	50
3.13	NPM.....	50
3.13.1	Descripción	50
3.13.2	Justificación de su uso	50
3.13.3	Fuentes	50
3.14	BOWER.....	51
3.14.1	Descripción	51
3.14.2	Justificación de su uso	51
3.14.3	Fuentes	51
3.15	OTROS ASPECTOS TEÓRICOS	51
3.15.1	Fuentes	56
CAPÍTULO 4.	PLANIFICACIÓN DEL PROYECTO Y PRESUPUESTO INICIALES.....	58
4.1	PLANIFICACIÓN INICIAL.....	58
4.1.1	Calendario de los recursos de trabajo.....	61
4.2	PRESUPUESTO INICIAL.....	63
4.2.1	Presupuesto interno.....	63
4.2.2	Presupuesto para el cliente.....	64
CAPÍTULO 5.	ANÁLISIS	66
5.1	DEFINICIÓN DEL SISTEMA.....	66
5.1.1	Determinación del Alcance del Sistema	66
5.2	REQUISITOS DEL SISTEMA	68
5.2.1	Obtención de los Requisitos del Sistema.....	68
5.2.2	Identificación de Actores del Sistema	74
5.2.3	Especificación de Casos de Uso.....	75

5.3	IDENTIFICACIÓN DE LOS SUBSISTEMAS EN LA FASE DE ANÁLISIS	86
5.3.1	<i>Descripción de los Subsistemas</i>	86
5.3.2	<i>Descripción de los Interfaces entre Subsistemas</i>	86
5.4	DIAGRAMA DE CLASES PRELIMINAR DEL ANÁLISIS.....	87
5.4.1	<i>Diagrama de Clases</i>	87
5.4.2	<i>Descripción de las Clases</i>	88
5.5	ANÁLISIS DE CASOS DE USO Y ESCENARIOS.....	102
5.5.1	<i>Caso de Uso: Registrarse</i>	102
5.5.2	<i>Caso de Uso: Iniciar sesión</i>	102
5.5.3	<i>Caso de Uso: Cambiar idioma</i>	103
5.5.4	<i>Caso de Uso: Modificar datos cuenta</i>	103
5.5.5	<i>Caso de Uso: Crear sala</i>	104
5.5.6	<i>Caso de Uso: Añadir contactos</i>	105
5.5.7	<i>Caso de Uso: Enviar solicitud contacto</i>	105
5.5.8	<i>Caso de Uso: Aceptar solicitud contacto</i>	106
5.5.9	<i>Caso de Uso: Ignorar solicitud contacto</i>	106
5.5.10	<i>Caso de Uso: Aceptar solicitud de unión a sala</i>	106
5.5.11	<i>Caso de Uso: Ignorar solicitud de unión a sala</i>	107
5.5.12	<i>Caso de Uso: Acceder a sala</i>	107
5.5.13	<i>Caso de Uso: Filtrar salas</i>	108
5.5.14	<i>Caso de Uso: Filtrar contactos</i>	108
5.5.15	<i>Caso de Uso: Bloquear contacto</i>	108
5.5.16	<i>Caso de Uso: Desbloquear contacto</i>	109
5.5.17	<i>Caso de Uso: Eliminar cuenta</i>	110
5.5.18	<i>Caso de Uso: Abrir servicio</i>	110
5.5.19	<i>Caso de Uso: Cerrar servicio</i>	111
5.5.20	<i>Caso de Uso: Mover servicio</i>	111
5.5.21	<i>Caso de Uso: Redimensionar servicio</i>	111

5.5.22	Caso de Uso: Crear dibujo colaborativo	112
5.5.23	Caso de Uso: Compartir URL video	112
5.5.24	Caso de Uso: Compartir presentación	112
5.5.25	Caso de Uso: Participar en videollamada	113
5.5.26	Caso de Uso: Enviar mensaje de texto.....	114
5.5.27	Caso de Uso: Enviar fichero	114
5.5.28	Caso de Uso: Descargar fichero.....	114
5.5.29	Caso de Uso: Abandonar sala.....	115
5.5.30	Caso de Uso: Cambiar permisos solicitud.....	115
5.5.31	Caso de Uso: Eliminar solicitud.....	116
5.5.32	Caso de Uso: Cambiar permisos miembros	116
5.5.33	Caso de Uso: Enviar solicitud de unión a sala.....	116
5.5.34	Caso de Uso: Eliminar miembro	117
5.5.35	Caso de Uso: Eliminar sala	117
5.5.36	Caso de Uso: Cambiar permisos moderadores.....	118
5.5.37	Caso de Uso: Eliminar moderadores.....	118
5.6	ANÁLISIS DE INTERFACES DE USUARIO	119
5.6.1	Descripción de la Interfaz	119
5.6.2	Descripción del Comportamiento de la Interfaz	137
5.6.3	Diagrama de Navegabilidad	138
5.7	ESPECIFICACIÓN DEL PLAN DE PRUEBAS.....	139
5.7.1	Pruebas unitarias.....	139
5.7.2	Pruebas de integración y del sistema	139
5.7.3	Pruebas de usabilidad.....	139
5.7.4	Pruebas de accesibilidad.....	140
5.7.5	Pruebas de rendimiento.....	140
5.7.6	Pruebas a partir de casos de uso	140
CAPÍTULO 6.	DISEÑO DEL SISTEMA.....	148

6.1	ARQUITECTURA DEL SISTEMA	148
6.1.1	<i>Diagramas de Paquetes</i>	148
6.1.2	<i>Diagramas de Componentes</i>	152
6.1.3	<i>Diagramas de Despliegue</i>	154
6.2	DISEÑO DE CLASES	157
6.2.1	<i>Diagrama de Clases</i>	157
6.3	DIAGRAMAS DE INTERACCIÓN.....	168
6.3.1	<i>Crear sala</i>	168
6.3.2	<i>Mostrar moderadores y miembros de una sala y eliminarlos</i>	169
6.3.3	<i>Abrir chat de texto y mandar mensaje</i>	170
6.4	DIAGRAMAS DE ACTIVIDADES	171
6.5	DISEÑO DE LA BASE DE DATOS.....	172
6.5.1	<i>Descripción del SGBD Usado</i>	172
6.5.2	<i>Integración del SGBD en Nuestro Sistema</i>	173
6.5.3	<i>Diagrama de la base de datos</i>	174
6.6	DISEÑO DE LA INTERFAZ	175
6.6.1	<i>Interfaz de inicio de sesión y registro</i>	175
6.6.2	<i>Interfaz de la página principal</i>	176
6.6.3	<i>Interfaz de las salas</i>	182
6.6.4	<i>Interfaz para la gestión de salas</i>	189
6.6.5	<i>Interfaz de ajustes de la cuenta</i>	192
6.6.6	<i>Interfaz con información sobre el creador</i>	192
6.6.7	<i>Interfaz aviso legal</i>	192
6.6.8	<i>Interfaz de error en la navegación</i>	193
6.7	ESPECIFICACIÓN TÉCNICA DEL PLAN DE PRUEBAS.....	193
6.7.1	<i>Pruebas Unitarias</i>	194
6.7.2	<i>Pruebas de Integración y del Sistema</i>	201
6.7.3	<i>Pruebas de Usabilidad y Accesibilidad</i>	208

6.7.4	Pruebas de Rendimiento.....	214
CAPÍTULO 7. IMPLEMENTACIÓN DEL SISTEMA		216
7.1	ESTÁNDARES Y NORMAS SEGUIDOS	216
7.1.1	CSS3	216
7.1.2	JSON.....	216
7.2	LINGÜAJES	217
7.2.1	Lingüajes de Programación	217
7.2.2	Lingüajes de etiquetas	217
7.2.3	Lingüajes de diseño gráfico	218
7.3	HERRAMIENTAS Y PROGRAMAS USADOS PARA EL DESARROLLO.....	218
7.3.1	IntelliJ IDEA Ultimate Edition	218
7.3.2	Node.js.....	219
7.3.3	AngularJS	219
7.3.4	Neo4j.....	220
7.3.5	Express.....	220
7.3.6	Otras herramientas.....	220
7.4	CREACIÓN DEL SISTEMA	221
7.4.1	Problemas Encontrados.....	221
7.4.2	Descripción Detallada de las Clases.....	225
CAPÍTULO 8. DESARROLLO DE LAS PRUEBAS		226
8.1	PRUEBAS UNITARIAS.....	226
8.2	PRUEBAS DE INTEGRACIÓN Y DEL SISTEMA	234
8.3	PRUEBAS DE USABILIDAD Y ACCESIBILIDAD.....	243
8.3.1	Pruebas de Usabilidad	243
8.3.2	Pruebas de Accesibilidad	266
8.4	PRUEBAS DE RENDIMIENTO.....	270
8.4.1	Pruebas con navegador Chrome.....	270
8.4.2	Pruebas con Apache JMeter	272

CAPÍTULO 9. MANUALES DEL SISTEMA.....	276
9.1 MANUAL DE INSTALACIÓN	276
9.2 MANUAL DE EJECUCIÓN	277
9.3 MANUAL DE USUARIO	278
9.3.1 <i>Modificar idioma de la aplicación.....</i>	278
9.3.2 <i>Iniciar sesión y registrarse en la aplicación</i>	278
9.3.3 <i>Añadir contactos.....</i>	280
9.3.4 <i>Aceptar o ignorar solicitudes de contacto</i>	281
9.3.5 <i>Bloquear contactos.....</i>	284
9.3.6 <i>Crear salas</i>	285
9.3.7 <i>Aceptar o ignorar solicitudes de unión a salas</i>	287
9.3.8 <i>Gestionar salas</i>	288
9.3.9 <i>Utilización de las salas y sus servicios.....</i>	289
9.3.10 <i>Ajustes de la cuenta</i>	296
9.3.11 <i>Cerrar sesión.....</i>	297
9.3.12 <i>Otros datos de interés</i>	298
9.4 MANUAL DEL PROGRAMADOR.....	299
9.4.1 <i>Primeros pasos para crear un servicio</i>	299
9.4.2 <i>Añadir funcionalidad y comunicación en tiempo real a nuestro servicio</i>	300
CAPÍTULO 10. CONCLUSIONES Y AMPLIACIONES.....	301
10.1 CONCLUSIONES.....	301
10.2 AMPLIACIONES	302
10.2.1 <i>Enviar mensajes sin utilizar salas</i>	302
10.2.2 <i>Aceptar solicitudes de salas automáticamente.....</i>	302
10.2.3 <i>Aplicación para móviles.....</i>	302
10.2.4 <i>Ampliación del número de servicios</i>	303
10.2.5 <i>Recuperar contraseña por email</i>	303
CAPÍTULO 11. PLANIFICACIÓN DEL PROYECTO Y PRESUPUESTO FINALES	304

11.1	PLANIFICACIÓN FINAL	304
11.2	PRESUPUESTO FINAL.....	308
11.2.1	<i>Presupuesto Interno</i>	308
11.2.2	<i>Presupuesto para el cliente</i>	309
CAPÍTULO 12.	REFERENCIAS BIBLIOGRÁFICAS	310
12.1	LIBROS Y ARTÍCULOS.....	310
12.2	REFERENCIAS EN INTERNET	311
CAPÍTULO 13.	APÉNDICES	313
13.1	GLOSARIO Y DICCIONARIO DE DATOS	313
13.2	CONTENIDO ENTREGADO EN EL ARCHIVO ADJUNTO	314
13.2.1	<i>Contenidos</i>	314
13.2.2	<i>Código Ejecutable e Instalación</i>	315
13.2.3	<i>Ficheros de Configuración</i>	316
13.3	ÍNDICE ALFABÉTICO	317
13.4	CÓDIGO FUENTE	319

Índice de Figuras

Figura 2.1. Skype.....	29
Figura 2.2. Google Slides	30
Figura 2.3. GoToMeeting y su función de dibujo.....	31
Figura 2.4. Talky	31
Figura 2.5. FaceTime.....	32
Figura 3.1. Node.js	38
Figura 3.2. AngularJS.....	39
Figura 3.3. Express	40
Figura 3.4. Neo4j.....	41
Figura 3.5. WebSockets.....	42
Figura 3.6. WebRTC.....	43
Figura 3.7. EJS	44
Figura 3.8. Traza generada por un error en un fichero EJS	45
Figura 3.9. CSS3	45
Figura 3.10. Arquitectura REST.....	46
Figura 3.11. IntelliJ IDEA Ultimate.....	47
Figura 3.12. Creación de proyecto en IntelliJ IDEA Ultimate Edition	48
Figura 3.13. Bootstrap.....	49
Figura 3.14. Gridstack.js.....	49
Figura 3.15. npm	50
Figura 3.16. Bower	51
Figura 4.1. Tareas planificación inicial	59
Figura 4.2. Diagrama de Gantt planificación inicial	62
Figura 5.1. Caso de uso de usuario no registrado	75
Figura 5.2. Caso de uso de usuario registrado	76
Figura 5.3. Caso de uso compartidos por miembro, moderador y administrador	80

Figura 5.4. Casos de uso compartidos por miembro y moderador	82
Figura 5.5. Casos de uso compartidos por administrador y moderador	83
Figura 5.6. Caso de uso administrador	85
Figura 5.7. Diagrama de clases preliminar	87
Figura 5.8. Interfaz inicio de sesión.....	119
Figura 5.9. Interfaz para registrarse.....	120
Figura 5.10. Interfaz página principal	121
Figura 5.11. Interfaz para gestionar salas.....	122
Figura 5.12. Interfaz para gestionar salas como administrador	122
Figura 5.13. Interfaz para gestionar salas como moderador	123
Figura 5.14. Interfaz para gestionar salas como miembro	124
Figura 5.15. Interfaz salas	125
Figura 5.16. Interfaz salas mostrando listado de servicios	125
Figura 5.17. Interfaz salas mostrando listado de participantes conectados	126
Figura 5.18. Interfaz para ajustes de la cuenta	127
Figura 5.19. Barra de navegación	127
Figura 5.20. Diálogo añadir contacto	128
Figura 5.21. Diálogo solicitudes de contactos.....	129
Figura 5.22. Diálogo bloquear y desbloquear contacto.....	130
Figura 5.23. Diálogo crear sala.....	131
Figura 5.24. Diálogo solicitudes de salas	132
Figura 5.25. Interfaz servicio de videollamada	133
Figura 5.26. Interfaz servicio de chat de texto	134
Figura 5.27. Interfaz servicio de presentaciones	135
Figura 5.28. Interfaz servicio de video compartido.....	135
Figura 5.29. Interfaz servicio de dibujos.....	136
Figura 5.30. Interfaz servicio de radio	136
Figura 5.31. Mensaje de error de angular-growl-2.....	137

Figura 5.32. Interfaz para errores de navegación	137
Figura 5.33. Diagrama de navegabilidad	138
Figura 6.1. Diagrama de paquetes del cliente	148
Figura 6.2. Diagrama de paquetes del servidor	151
Figura 6.3. Diagrama de componentes del cliente.....	152
Figura 6.4. Diagrama de componentes del servidor	154
Figura 6.5. Diagrama de despliegue	155
Figura 6.6. Clases controllers (Paquete public).....	158
Figura 6.7. Clases services	158
Figura 6.8. Clases utils.....	159
Figura 6.9. Clases managers	160
Figura 6.10. Diagrama de clases cliente.....	161
Figura 6.11. Clases config	162
Figura 6.12. Clases controllers (Paquete api).....	163
Figura 6.13. Clases routes (Paquete api)	163
Figura 6.14. Clases utils	164
Figura 6.15 Diagrama de clases api	165
Figura 6.16. Clases controllers (Paquete server).....	166
Figura 6.17. Clases routes (Paquete server)	166
Figura 6.18. Clases servers	166
Figura 6.19 Diagrama de clases server	167
Figura 6.20. Diagrama de Interacción crear sala.....	168
Figura 6.21. Diagrama de Interacción mostrar moderadores y miembros de una sala y eliminarlos ..	169
Figura 6.22. Diagrama de Interacción chat de texto	170
Figura 6.23. Diagrama de Actividad crear sala.....	171
Figura 6.24. Diagrama de la base de datos.....	174
Figura 6.25. Interfaz de inicio de sesión	175
Figura 6.26. Interfaz registro	176

Figura 6.27. Interfaz página principal	177
Figura 6.28. Diálogo añadir usuario.....	177
Figura 6.29. Listado de usuarios	178
Figura 6.30. Diálogo solicitudes de contacto	178
Figura 6.31. Diálogo bloquear y desbloquear contactos.....	179
Figura 6.32. Diálogo crear sala	180
Figura 6.33. Diálogo solicitudes de salas	181
Figura 6.34. Diálogo datos de contacto	181
Figura 6.35. Interfaz sala	182
Figura 6.36. Servicios sala	183
Figura 6.37. Participantes conectados en la sala	183
Figura 6.38. Icono para redimensionar servicios	183
Figura 6.39. Interfaz videollamada.....	184
Figura 6.40. Interfaz videollamada con tres usuarios	185
Figura 6.41. Interfaz chat de texto	185
Figura 6.42. Interfaz radio	186
Figura 6.43. Interfaz presentaciones	187
Figura 6.44. Interfaz video compartido	187
Figura 6.45. Interfaz dibujos.....	188
Figura 6.46. Sala con 4 servicios abiertos	189
Figura 6.47. Sala con 3 servicios abiertos48	189
Figura 6.49. Interfaz gestión salas	190
Figura 6.50. Interfaz información de la sala	190
Figura 6.51. Interfaz usuarios de la sala	191
Figura 6.52. Interfaz acciones peligrosas.....	191
Figura 6.53. Interfaz ajustes de la cuenta	192
Figura 6.54. Interfaz con información del creador	192
Figura 6.55. Interfaz aviso legal.....	193

Figura 6.56. Interfaz error en la navegación	193
Figura 7.1. CSS validado por W3C.....	216
Figura 7.2. JSON.....	216
Figura 7.3. JavaScript	217
Figura 7.4. EJS	217
Figura 7.5. CSS3	218
Figura 7.6. IntelliJ IDEA Ultimate.....	219
Figura 7.7. Node.js	219
Figura 7.8. AngularJS.....	219
Figura 7.9. Neo4j.....	220
Figura 7.10. Express	220
Figura 8.1. Pruebas unitarias con Mocha	234
Figura 8.2. Menú lateral página principal.....	258
Figura 8.3. Barra de navegación	258
Figura 8.4. Página principal deuteranopía.....	266
Figura 8.5. Sala deuteranopía	267
Figura 8.6. Página principal tritanopía	267
Figura 8.7. Sala tritanopía	268
Figura 8.8. Página principal visión monocromática	268
Figura 8.9. Sala visión monocromática.....	269
Figura 8.10. Entorno Chrome pruebas rendimiento	270
Figura 8.11. Resultados prueba rendimiento iniciar sesión	271
Figura 8.12. Resultados prueba rendimiento registro	272
Figura 8.13. Petición iniciar sesión JMeter	273
Figura 8.14. Gráfico petición iniciar sesión JMeter	273
Figura 8.15. Petición registrar JMeter	274
Figura 8.16. Gráfico petición registrar JMeter	275
Figura 9.1. Manual usuario - Localización banderas idiomas	278

Figura 9.2. Manual de usuario - Iniciar sesión	279
Figura 9.3. Manual de usuario - Registro.....	279
Figura 9.4. Manual de usuario - Página principal	280
Figura 9.5. Manual de usuario - Añadir contacto.....	281
Figura 9.6. Manual de usuario - Listado para seleccionar usuario	281
Figura 9.7. Manual de usuario - Notificación solicitudes de contacto.....	282
Figura 9.8. Manual de usuario - Solicitudes de contacto	282
Figura 9.9. Manual de usuario - Listado de contactos	283
Figura 9.10. Manual de usuario - Información sobre el contacto.....	284
Figura 9.11. Manual de usuario - Bloquear contacto.....	284
Figura 9.12. Manual de usuario - Crear sala	285
Figura 9.13. Manual de usuario - Listado de salas	286
Figura 9.14. Manual de usuario - Notificación solicitudes de salas	287
Figura 9.15. Manual de usuario - Solicitudes de salas	287
Figura 9.16. Manual de usuario - Gestionar salas.....	288
Figura 9.17. Manual de usuario - Sala	289
Figura 9.18. Manual de usuario - Menú de servicios	290
Figura 9.19. Manual de usuario - Menú de participantes	290
Figura 9.20. Manual de usuario - Desplazar servicio.....	291
Figura 9.21. Manual de usuario - Flechas redimensionar	291
Figura 9.22. Manual de usuario - Símbolo cerrar servicio.....	291
Figura 9.23. Manual de usuario - Videollamada	292
Figura 9.24. Manual de usuario - Chat de texto.....	293
Figura 9.25. Manual de usuario - Presentación HTML	294
Figura 9.26. Manual de usuario - Presentación PDF	294
Figura 9.27. Manual de usuario - Video compartido.....	295
Figura 9.28. Manual de usuario - Dibujos	295
Figura 9.29. Manual de usuario – Radio	296

Figura 9.30. Manual de usuario - Ajustes de la cuenta	297
Figura 9.31. Manual de usuario - Cerrar sesión menú lateral	298
Figura 9.32. Manual de usuario - Cerrar sesión menú desplegable.....	298
Figura 11.1. Tareas planificación final	305
Figura 11.2. Diagrama de Gantt planificación final.....	307

Índice de Tablas

Tabla 2.1. Comparación de aplicaciones actuales con este proyecto	33
Tabla 3.1. Otras tecnologías utilizadas.....	56
Tabla 4.1. Presupuesto interno inicial	63
Tabla 4.2. Presupuesto cliente inicial	65
Tabla 5.1. Requisitos usuarios no registrados.....	68
Tabla 5.2. Requisitos usuarios registrados.....	70
Tabla 5.3. Requisitos usuarios que participan en una sala	72
Tabla 5.4. Requisitos miembros y moderadores	72
Tabla 5.5. Requisitos administradores y moderadores.....	72
Tabla 5.6. Requisitos administradores	73
Tabla 5.7. Requisitos tecnológicos	73
Tabla 5.8. Requisitos de usabilidad	73
Tabla 5.9. Requisitos de seguridad.....	73
Tabla 11.1. Presupuesto interno final	308
Tabla 11.2. Presupuesto cliente final	309

Capítulo 1. Memoria del Proyecto

1.1 Resumen de la Motivación, Objetivos y Alcance del Proyecto

Uno de los problemas principales de las actuales plataformas de comunicaciones es que proporcionan unos servicios limitados a sus usuarios, lo que les obligan a abrirse cuentas en diversas plataformas para poder realizar todas aquellas comunicaciones que necesitan. Esto también deriva en la necesidad de descargar varios programas en sus ordenadores o utilizar diferentes Webs de manera simultánea.

Otro problema es que algunas aplicaciones aún siguen estancadas en tecnologías propietarias que no proporcionan las ventajas que se pueden conseguir con herramientas un poco más novedosas. Las comunicaciones de este proyecto estarán sostenidas por dos tecnologías que actualmente están teniendo una gran acogida y de las cuales se espera un gran futuro.

En este proyecto se construirá una plataforma de comunicaciones web cuya funcionalidad será extensible, pero que esto no derivará en un aumento de la complejidad de la misma. Para ello se diseñará una interfaz sencilla pero que, a su vez, pueda disponer de todas aquellas formas de comunicaciones que existan o vayan a existir. De esta forma los usuarios dispondrán de videollamadas, chats de texto, dibujos colaborativos, videos compartidos, etc., en un único lugar y sin necesidad de software adicional.

El motivo por el cual se decidió realizar este proyecto fue la posibilidad de trabajar en un área tan interesante y necesaria en nuestra sociedad como es la comunicación de forma remota. Además, este proyecto me da la posibilidad de trabajar con algunas tecnologías que no han podido ser estudiadas durante la carrera, pero que son muy útiles y probablemente lo serán aún más en el futuro.

1.2 Resumen de Todos los Aspectos

Esta memoria está compuesta por 13 capítulos que a continuación explicaremos brevemente:

- **Capítulo 1.** Resumen de la motivación y objetivos del proyecto.
- **Capítulo 2.** Justificación del proyecto, listado de sus objetivos y estudio de la situación actual de las comunicaciones y tecnologías.
- **Capítulo 3.** Explicación de las tecnologías utilizadas.
- **Capítulo 4.** Planificación y presupuesto iniciales del proyecto.
- **Capítulo 5.** Estudio de los requisitos, casos de uso, identificación de subsistemas, diagramas de clases, escenarios, diseño inicial de interfaces y especificación del plan de pruebas.
- **Capítulo 6.** Diagramas realizados para facilitar la implementación del sistema, interfaces definitivas realizadas y especificación del plan de pruebas.
- **Capítulo 7.** Explicación de tecnologías utilizados para la implementación, problemas encontrados durante esa fase y localización de la documentación del código.
- **Capítulo 8.** Resultados del plan de pruebas definido durante el análisis y diseño.
- **Capítulo 9.** Manuales de instalación, ejecución, usuario y programador.
- **Capítulo 10.** Conclusiones alcanzadas tras realizar el proyecto y futuras ampliaciones.
- **Capítulo 11.** Estudio de la planificación y presupuestos finales.
- **Capítulo 12.** Referencias bibliográficas.
- **Capítulo 13.** Glosario, explicación del contenido entregado en el archivo adjunto e índice alfabético.

Capítulo 2. Introducción

2.1 Justificación del Proyecto

Copernicus es una plataforma web de comunicaciones cuyo principal objetivo es ser el centro de todas las comunicaciones de sus usuarios, es decir, busca reunir todas aquellas formas de comunicación que necesitan sus usuarios sin necesidad de utilizar un software alternativo.

Es habitual disponer de múltiples cuentas en plataformas como Skype, Telegram, GoToMeeting, etc., ya que cada una de ellas ofrece diferentes formas de comunicarse. Esto conlleva en ocasiones la necesidad de descargar varios programas en nuestro ordenador y memorizar varios nombres de usuario y contraseñas. Copernicus busca simplificar las comunicaciones del usuario mediante una única aplicación web en la que se incluyen todas aquellas formas de comunicación necesarias.

Un problema de añadir mucha funcionalidad a una única plataforma es que esto puede derivar en una sobrecarga de sus interfaces y hacer el uso del software más complejo para sus usuarios. Copernicus evita este problema mediante el uso de una interfaz con un diseño “modular” que permite al usuario gestionar las interfaces de los servicios. De este modo se da la posibilidad de abrir, cerrar y mover por la pantalla los servicios como el chat de texto, videollamadas o dibujos colaborativos de manera independiente y sencilla.

Gracias a la interfaz diseñada para esta plataforma también se busca que terceros desarrolladores puedan incluir sus propios servicios de una forma sencilla y que no perjudique la interfaz de usuario. De modo que la funcionalidad que ofrece esta plataforma siempre estaría en evolución.

2.2 Objetivos del Proyecto

El objetivo principal de este proyecto es proporcionar a sus usuarios una plataforma en la que puedan mantener un listado de contactos y salas. Estas salas, además, les permitirán realizar diferentes comunicaciones en tiempo real.

Las comunicaciones en las salas serán posible gracias a lo que se ha denominado “Servicios”. Estos servicios proporcionarán diferentes formas de comunicarse a los usuarios en tiempo real, estas son:

1. Videollamadas
2. Dibujos colaborativos
3. Chat de texto
4. Videos compartidos

5. Radio y música compartida
6. Presentaciones compartidas

Además, se busca que la aplicación pueda ser accedida por un gran número de personas. Por ello, esta será traducida al inglés y español.

2.3 Estudio de la Situación Actual

En este apartado se han evaluado diferentes aplicaciones que proveen unos servicios similares al software desarrollado en este proyecto. Esto es, aplicaciones de videoconferencias, chats, intercambio de archivos u otras funciones cooperativas como el desarrollo de presentaciones o textos.

En general, todas las aplicaciones comparten un mismo objetivo. Facilitar las reuniones y el trabajo cooperativo entre sus usuarios, permitiendo su utilización desde cualquier lugar con acceso a internet y evitando así cualquier tipo de desplazamiento físico.

El problema de algunas aplicaciones estudiadas es la necesidad de descargar un software para su utilización, o el reducido abanico de posibilidades que proporciona. Con este proyecto se pretende integrar varios servicios en una única aplicación.

Este estudio se ha realizado antes comenzar el desarrollo del proyecto, con el fin de identificar funcionalidades útiles, y estudiar las carencias de las aplicaciones actuales para descubrir nuevas funciones que puedan diferenciar este proyecto en el mercado.

2.3.1 Aplicaciones estudiadas

Existen una gran variedad de aplicaciones de comunicaciones, y el primer objetivo de este proyecto es tratar de identificar sus posibles carencias y problemas. Por ello en los siguientes puntos se describirán algunas de las plataformas más famosas.

2.3.1.1 Skype

Skype es una de las aplicaciones de videollamadas y llamadas de voz más populares. Se basa en que cada usuario posee una lista de contactos con los cuales puede realizar llamadas de voz, video o utilizar un chat de texto. Además, Skype posee otras funciones interesantes como, por ejemplo, la creación de grupos o compartir pantalla nuestra pantalla.

Skype posee una aplicación de escritorio, otra para dispositivos móviles y además una aplicación web que, en estos momentos, está en fase beta. Gracias a esto es posible utilizar sus servicios desde un gran número de dispositivos.

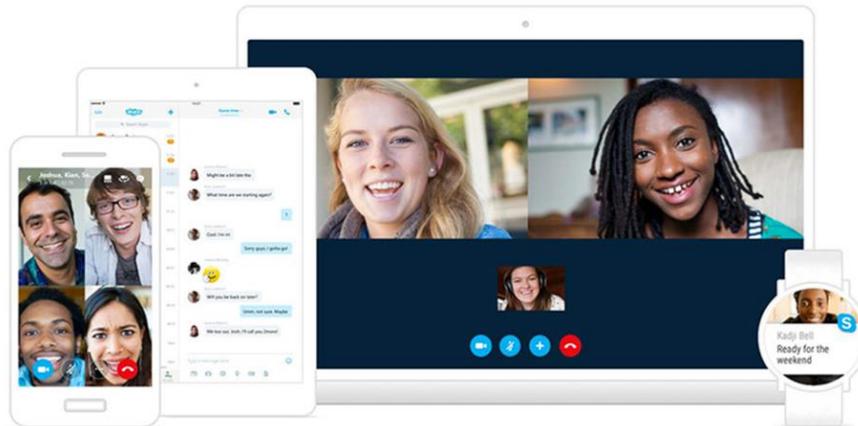


Figura 2.1. Skype

2.3.1.1.1 Aportaciones a este proyecto

La posibilidad de crear grupos de usuarios a los cuales puede asignarse un nombre y foto es una función interesante que será incluidas en este proyecto. Por motivos de rendimiento, estos grupos (“salas” en esta aplicación) serán limitados a 4 personas. Se estudió la posibilidad de incluir un servicio de compartición de nuestra pantalla, pero debido a los plazos para realizar el proyecto se priorizaron otros servicios en su lugar.

Skype es una aplicación “freemium”, ya que provee servicios gratuitos, pero algunas de sus características son de pago. Esto llevó a evaluar la estrategia de precios para la aplicación desarrollada en este proyecto. Finalmente se decidió que todos los servicios de la aplicación serían gratuitos, pero no se cierra la posibilidad de añadir servicios de pago en un futuro.

2.3.1.2 Google Slides

Google provee una suite ofimática en línea en la que encontramos aplicaciones web como Google Docs, Google Sheets y, en el caso que nos interesa, Google Slides.

Google Slides permite la creación y edición de presentaciones por varios usuarios al mismo tiempo. Esto permite tener y editar presentaciones en un mismo lugar, evitando así inconsistencias.

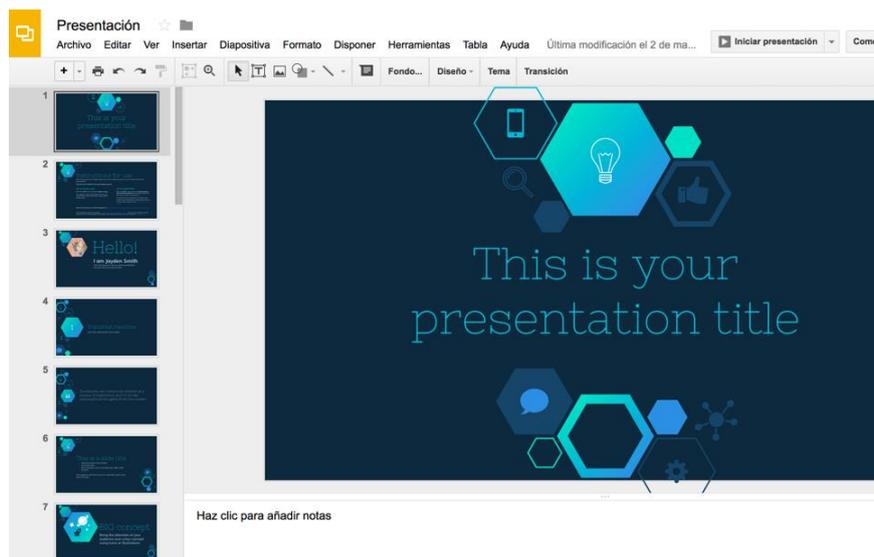


Figura 2.2. Google Slides

2.3.1.2.1 Aportaciones a este proyecto

En este proyecto no se añadirá la edición de presentaciones, pero permitirá que sus usuarios compartan presentaciones en formato HTML. Además, el servicio de presentaciones será también utilizado para el visionado de ficheros PDF.

2.3.1.3 GoToMeeting

GoToMeeting es una aplicación de videoconferencias basada en la utilización de URLs e identificadores para acceder a chats en grupo. A parte de proveer las funciones típicas de videollamadas, llamadas de voz y de texto, GoToMeeting tiene interesantes servicios como compartir dibujos o controlar remotamente el ordenador de otro miembro del grupo.

El principal problema de esta plataforma es que su servicio es de pago, además de que no permite la compartir videos y presentaciones. Debido a que esta es una aplicación dirigida al sector de negocios, podría ser interesante esta opción.

2.3.1.3.1 Aportaciones a este proyecto

GoToMeeting es una aplicación dirigida en particular hacia el área de negocios que proporciona características interesantes, pero obligando a pagar una suscripción mensual. Existe una versión gratuita pero sus servicios son reducidos considerablemente. Uno de los servicios diferenciadores de GoToMeeting, y que es de pago, es la posibilidad de compartir dibujos con otros miembros de la reunión.

Después de evaluar GoToMeeting se decidió que la función de compartir dibujos puede ser útil para nuestra aplicación, por ello se incluirá una versión más simple debido a los

cortos plazos de tiempo para la realización de este proyecto. Se dejará para el futuro la realización de la función de control remoto de otros ordenadores.

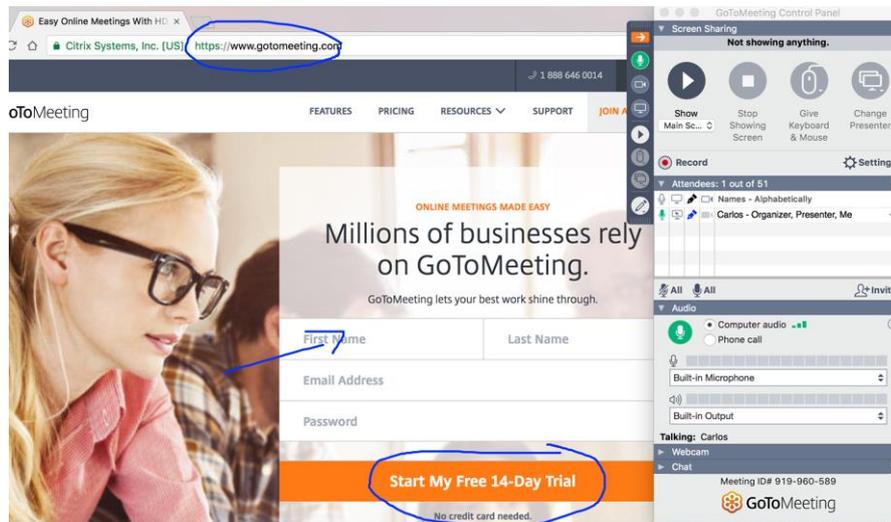


Figura 2.3. GoToMeeting y su función de dibujo

2.3.1.4 Talky

Talky es una aplicación web de videoconferencias que utiliza WebRTC. Esta aplicación permite la creación de grupos de usuarios sin necesidad de registrarse. Se basa en la creación de una URL a la que podemos añadir un nombre que identifique el grupo. Después esa URL es comunicada al resto de usuarios que queremos invitar, los cuales únicamente tienen que introducir en el navegador para acceder al grupo. También incluye la posibilidad de introducir una contraseña en el grupo para evitar que personas ajenas que conozcan la URL puedan acceder a la conversación.

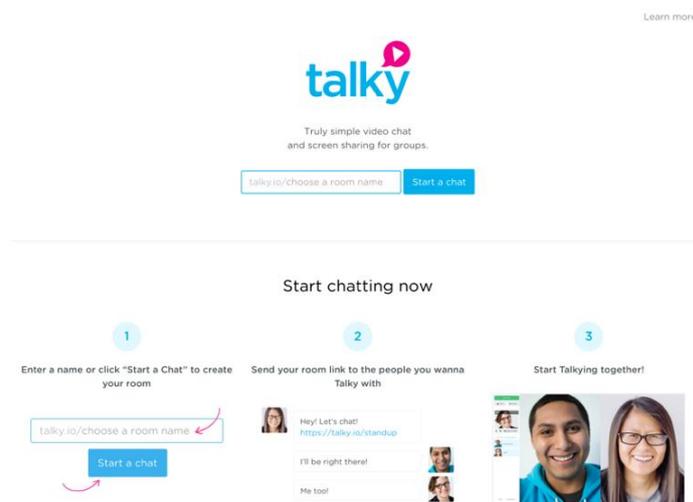


Figura 2.4. Talky

2.3.1.4.1 Aportaciones a este proyecto

Una de las ventajas de Talky es que no requiere la descarga de ningún software, por lo que puede utilizarse desde una gran variedad de dispositivos que dispongan de navegador web Chrome, Firefox u Opera. Esta característica es muy interesante y, por tanto, será utilizada en este proyecto.

Otra ventaja de Talky es que permite utilizar sus servicios sin necesidad de registrarse. Tras el estudio de esta aplicación se evaluó la posibilidad de eliminar el proceso de registro simplificando el uso de la aplicación. Sin embargo, esta ventaja conlleva dos inconvenientes: la imposibilidad de mantener una lista de contactos para cada usuario y la necesidad de que los usuarios dispongan de un segundo servicio para poder comunicar la URL de nuestro grupo. No es igual de sencillo recordar una URL potencialmente compleja que un nombre de usuario. Por lo tanto, esta opción fue desestimada.

2.3.1.5 FaceTime

FaceTime es una aplicación de videoconferencias para dispositivos de la marca Apple que se caracteriza por su simplicidad. Esta aplicación únicamente nos permite realizar llamadas de video o de voz con una sola persona, dando la posibilidad de mantener una lista de contactos.



Figura 2.5. FaceTime

2.3.1.5.1 Aportaciones a este proyecto

Copernicus incluirá nuestro listado de contactos en la pantalla principal con un campo de filtrado de contactos del mismo modo que FaceTime. Esto hará que los usuarios puedan ver los datos de sus contactos y puedan bloquearlos de forma simple, sin que tengan que navegar por diversas páginas de la aplicación.

La simplicidad de FaceTime nos hace ver la necesidad de realizar una aplicación lo más intuitiva y simple posible. Por lo que en la página principal de Copernicus también se incluirá el listado de todas las salas en las que participa el usuario y todas las páginas de la plataforma podrán ser accedidas en un solo clic desde la página principal.

2.3.2 Tabla comparativa

En la siguiente tabla se muestran algunas de las características que se buscan en este proyecto y su situación respecto a los competidores analizados que proveen un servicio de videoconferencias:

Característica \ Aplicación	Skype	GoToMeeting	Talky	FaceTime	Copernicus
Sin software adicional	Si ¹	Si	Si	No	Si
Listado de Contactos	Si	No	No	Si	Si
Creación de grupos/salas	Si	Si ²	Si ³	No	Si
Dibujos colaborativos	No	Si ⁴	No	No	Si
Compartición de video, audio y presentaciones	No	No	No	No	Si
Posibilidad de incluir servicios de terceros	No	No	No	No	Si
Gratuito	Si ⁵	No ⁶	Si	Si	Si

Tabla 2.1. Comparación de aplicaciones actuales con este proyecto

2.3.3 Evaluación de Alternativas

A pesar de que este proyecto fue concebido para ser desarrollado utilizando tecnologías como Node.js, WebSockets y WebRTC, se realizó un estudio para valorar la utilización de otras alternativas. Sin embargo, debido al auge de frameworks como MEAN stack

¹ La aplicación web de Skype está en fase beta

² Permite la creación de reuniones, pero se necesita disponer de un ID de la reunión para acceder a ellas

³ Permite la creación de salas temporales, no permite mantener un listado de salas

⁴ Está en beta y solo disponible para la versión de escritorio

⁵ Las funciones principales son gratuitas, hay funciones adicionales que son de pago

⁶ Existe una suscripción gratuita pero los servicios proporcionados son reducidos

(MongoDB–Express–AngularJS–Node.js) y ANNE stack (AngularJS–Node.js–Neo4J–Express), la elección de las tecnologías a utilizar fue un proceso sencillo. El único punto que requirió una evaluación un poco más exhaustiva fue la elección del framework entre los dos propuestos anteriormente. Finalmente se decidió utilizar Neo4j como sistema de base de datos y, por lo tanto, utilizar el framework ANNE.

2.3.3.1 Alternativas de lenguaje de programación

Existen muchos lenguajes de programación que podrían ser considerados como alternativas a utilizar en este proyecto. Ejemplos de ellos son Ruby, Python o C#. Sin embargo, se presentarán dos de las alternativas más populares actualmente.

2.3.3.1.1 Java

2.3.3.1.1.1 Descripción

Java es un lenguaje de programación de propósito general orientado a objetos. Este es actualmente uno de los lenguajes de programación más populares, que se encuentra presente, por ejemplo, en aplicaciones móviles y servidores. Una de las características más llamativas de los programas desarrollados en Java es son desarrollados una vez, pero pueden ser ejecutadas en varias plataformas gracias al JRE (Java Runtime Environment).

2.3.3.1.1.2 Ventajas

Una de las principales ventajas de utilizar Java es que dispongo de amplios conocimientos de este lenguaje ya que ha sido utilizado en numerosas ocasiones a lo largo de toda la carrera de ingeniería informática. Esta ventaja repercutiría en necesitar un menor tiempo para el aprendizaje de las tecnologías utilizadas.

2.3.3.1.1.3 Inconvenientes

A pesar de que el uso del lenguaje de programación Java sea un punto positivo, también implica la utilización de más de un lenguaje de programación en el mismo proyecto, ya que el lado del cliente sería implementado utilizando también JavaScript. Este inconveniente es solucionado utilizando Node.js, ya que el lado del servidor es realizado utilizando únicamente JavaScript, lenguaje de programación también utilizado de manera exclusiva en la capa del cliente.

2.3.3.1.2 PHP

2.3.3.1.2.1 Descripción

PHP es un lenguaje de programación utilizado en el lado del servidor para la creación de páginas web dinámicas.

2.3.3.1.2.2 *Ventajas*

Algunas de las ventajas de PHP es su estabilidad, velocidad, flexibilidad, sus múltiples extensiones o la rapidez para desarrollar un script. Además, PHP es un lenguaje muy popular que tiene una gran comunidad, algo que puede ser realmente útil cuando tienes unos conocimientos básicos de un lenguaje.

2.3.3.1.2.3 *Inconvenientes*

Al igual que cuando se evaluó la utilización de Java, utilizar PHP conlleva el uso de más de un lenguaje de programación en el mismo proyecto. Al utilizar únicamente JavaScript se puede reducir considerablemente el tiempo de desarrollo de la aplicación.

Otro inconveniente de utilizar PHP es la necesidad de un periodo de tiempo para el aprendizaje de esta tecnología. Este paso sería imprescindible para poder realizar el proyecto correctamente.

2.3.3.2 *Alternativas de sistemas de gestión de bases de datos*

Al igual que ocurría con los lenguajes de programación, existen muchas alternativas a la hora de seleccionar la base de datos del sistema. En esta sección nos centraremos en estudiar dos de las alternativas más importantes en cuanto a base de datos NoSQL y relacionales.

2.3.3.2.1 **MongoDB**

2.3.3.2.1.1 *Descripción*

MongoDB es un sistema de base de datos NoSQL que almacena estructuras de datos en documentos con formato BSON.

2.3.3.2.1.2 *Ventajas*

La principal ventaja en la utilización de MongoDB respecto a Neo4j, es que ya poseo conocimientos suficientes para trabajar con MongoDB. Por el contrario, mis conocimientos sobre el lenguaje de consultas Cypher (Neo4j) son más limitados.

2.3.3.2.1.3 *Inconvenientes*

MongoDB no permite una representación tan “natural” de una red social como lo puede hacer una base de datos en grafo. Una base de datos en grafo como Neo4J, nos permite representar cada usuario y sala como un nodo, y cada conexión entre estos nodos mediante simples relaciones. En el caso de utilizar MongoDB, necesitaríamos documentos para asociar usuarios con salas, usuarios con sus contactos, usuarios con usuarios bloqueados, etc., creando documentos potencialmente grandes. En el caso de utilizar una base de datos en grafo el proceso se simplifica enormemente, teniendo únicamente que crear relaciones con atributos entre los nodos. De esta forma también se simplifica la realización de

consultas como la búsqueda de participantes de una sala, contactos de un usuario, sus contactos bloqueados, moderadores de una sala, etc.

2.3.3.2.2 MySQL

2.3.3.2.2.1 Descripción

MySQL es un sistema de gestión de bases de datos relacionales, las cuales se basan en utilizar tablas para almacenar y organizar la información. Es uno de los sistemas más populares actualmente y de código abierto.

2.3.3.2.2.2 Ventajas

Entre sus ventajas encontramos la posibilidad de ser ejecutado en máquinas con escasos recursos, que es soportado en una gran variedad de sistemas operativos o que es considerado uno de los sistemas de gestión de bases de datos más seguros.

2.3.3.2.2.3 Inconvenientes

Al igual que MongoDB, la creación y mantenimiento de las conexiones de una red social se hace un poco más compleja que utilizando grafos.

2.3.3.3 Alternativas de frameworks Web basados en Node.js

2.3.3.3.1 Sails

2.3.3.3.1.1 Descripción

Sails es un framework MVC para Node.js que facilita el desarrollo de aplicaciones web.

2.3.3.3.1.2 Ventajas

Sails tiene unas ventajas similares a Express, el cuál es el framework MVC seleccionado para realizar este proyecto. Entre sus ventajas encontramos el soporte para WebSockets, su compatibilidad con cualquier base de datos y permitir la creación de una API RESTful de manera automática.

2.3.3.3.1.3 Inconvenientes

Debido a los reducidos conocimientos sobre Sails, se ha optado por utilizar Express. Además, Express está en un estado de desarrollo más avanzado, es más sencillo de utilizar y tiene una comunidad mucho mayor que lo utiliza. Por lo tanto, sería más fácil solventar cualquier problema que nos encontremos durante el desarrollo del proyecto.

2.3.3.3.2 hapi

2.3.3.3.2.1 Descripción

Framework de código abierto basado en Node.js para la creación de aplicaciones y servicios Web.

2.3.3.3.2.2 Ventajas

La estructura de las aplicaciones desarrolladas con hapi hace más simple el desarrollo de proyectos en equipo.

2.3.3.3.2.3 Inconvenientes

Dispone de una comunidad muy pequeña en comparación con Express o Sails.

Capítulo 3. Aspectos Teóricos

En este capítulo se describirán las herramientas y tecnologías utilizadas para la realización de este proyecto.

3.1 Node.js

3.1.1 Descripción

Node.js es un entorno de ejecución de JavaScript orientado a eventos asíncronos, multiplataforma y de código abierto. Este permite la ejecución de JavaScript en el lado del servidor y crear aplicaciones altamente escalables.

Node.js fue creado por Ryan Lienhart Dahl, sin embargo, cualquiera puede contribuir en el desarrollo de este proyecto. En el momento de realización de esta memoria, existen 1388 colaboradores en su página de GitHub.



Figura 3.1. Node.js

3.1.2 Justificación de su uso

Node.js fue utilizado para la configuración del servidor utilizado por la plataforma desarrollada en este proyecto. Son tres las principales razones por las que se decidió utilizar Node.js en el lado del servidor:

- La posibilidad de escalar grandes volúmenes de clientes gracias a que da soporte a la programación asíncrona. Esto es realmente necesario en este proyecto ya que al ser una red social implicaría su uso simultaneo por un número potencialmente alto de usuarios.
- La posibilidad de utilizar únicamente JavaScript como lenguaje de programación para el desarrollo de todo el proyecto. Esto implica una reducción del tiempo de

aprendizaje de otras tecnologías como PHP, además de fortalecer mis conocimientos sobre uno de los lenguajes más populares del momento.

- El auge de la creación de aplicaciones WEB utilizando Node.js en el lado del servidor. Además de mejorar mis conocimientos sobre este entorno que me serán útiles en la creación de futuras aplicaciones web, existe una amplia comunidad muy útil para resolver todos aquellos que me pueda encontrar durante el desarrollo de la plataforma.

3.1.3 Fuentes

<https://nodejs.org> [Accedido por última vez el 09/02/17]

<https://es.wikipedia.org/wiki/Node.js> [Accedido por última vez el 09/02/17]

3.2 AngularJS

3.2.1 Descripción

AngularJS es un framework MVC para JavaScript de código abierto que proporciona librerías útiles para el desarrollo de aplicaciones web. AngularJS ha sido desarrollado por Google Inc. junto a una amplia comunidad gracias a su modelo de código abierto.

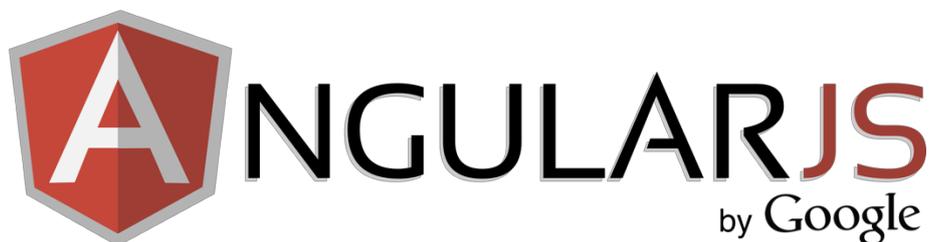


Figura 3.2. AngularJS

3.2.2 Justificación de su uso

Su gran popularidad fue una de las razones por las que ya me había planteado la utilización de este framework antes de realizar un estudio más exhaustivo de las posibilidades existentes. AngularJS cuenta con una gran comunidad que hace más fácil la solución de posibles problemas.

Existen dos grandes versiones de Angular que son completamente diferentes entre ellas, la versión 1 denominada AngularJS y la versión 2 llamada simplemente Angular. Actualmente ambas versiones tienen soporte, aunque se dice que la versión 2 viene a reemplazar la 1. Sin embargo, ya he realizado aplicaciones web utilizando AngularJS y

las diferencias entre ambos frameworks me obligarían a dedicar una gran parte del tiempo para la realización de este proyecto al aprendizaje de Angular en su versión 2. Por lo tanto, se ha decidido utilizar AngularJS en este proyecto.

3.2.3 Fuentes

<https://angularjs.org/> [Accedido por última vez el 09/02/17]

<https://es.wikipedia.org/wiki/AngularJS> [Accedido por última vez el 09/02/17]

<https://desarrolloweb.com/articulos/que-es-angularjs-descripcion-framework-javascript-conceptos.html> [Accedido por última vez el 09/02/17]

3.3 Express

3.3.1 Descripción

Express es un framework web rápido, minimalista, abierto y flexible para Node.js, que está construido sobre Connect, un framework extensible de manejo de servidores HTTP. Express fue desarrollado por TJ Holowaychuk y StrongLoop entre otros.



Figura 3.3. Express

3.3.2 Justificación de su uso

Uno de los frameworks más populares de Node.js que simplifica enormemente la creación de servidores.

3.3.3 Fuentes

<http://expressjs.com/es/> [Accedido por última vez el 09/02/17]

<https://en.wikipedia.org/wiki/Express.js> [Accedido por última vez el 09/02/17]

<https://www.genbetadev.com/frameworks/introduccion-a-la-programacion-asincrona-con-nodejs-desarrollo-web-con-expressjs> [Accedido por última vez el 09/02/17]

3.4 Base de datos orientadas a grafos

3.4.1 Descripción

Las bases de datos orientadas a grafos son una de las clasificaciones de los sistemas de gestión de bases de datos NoSQL. Estas se basan en representar la información utilizando nodos y las relaciones entre estos mediante el uso de aristas.

En este proyecto se utilizará la base de datos orientada a grafos Neo4j. Una base de datos altamente escalable desarrollada por Neo Technology.



Figura 3.4. Neo4j

3.4.2 Justificación de su uso

La principal razón por la que se utiliza una base de datos orientada a grafo es por la facilidad de representar la estructura de una red social en una de estas bases de datos ya que esta estructura es, básicamente, un grafo. Una base de datos en grafo como Neo4J nos permite representar cada usuario y sala como un nodo, y cada conexión entre estos nodos mediante simples relaciones.

Existen diversas ventajas de las bases de datos en grafo frente a otros tipos de bases de datos como, por ejemplo, frente a las relacionales. Entre estas ventajas encontramos un mejor rendimiento, flexibilidad y escalabilidad.

También influyeron en su uso la simplicidad del lenguaje de consultas Cypher de Neo4j, además de la facilidad para gestionar una base de datos de Neo4j con el software que nos es proporcionado.

3.4.3 Fuentes

<https://neo4j.com/use-cases/social-network/> [Accedido por última vez el 09/02/17]

<https://neo4j.com/blog/3-advantages-neo4j-alongside-oracle-rdbms/> [Accedido por última vez el 09/02/17]

3.5 WebSockets

3.5.1 Descripción

Esta tecnología permite el envío bidireccional de datos entre el cliente y el servidor sobre un único socket TCP. El servidor de WebSockets permite así solventar el problema de AJAX de ser necesario mandar peticiones cada cierto tiempo por parte del cliente al servidor para saber si existen actualizaciones.

Como se puede observar en la siguiente figura, su funcionamiento se basa en utilizar el protocolo HTTP únicamente para establecer la conexión entre el cliente y servidor. Desde ese momento se establece una comunicación bidireccional y full-duplex sobre un único socket TCP.

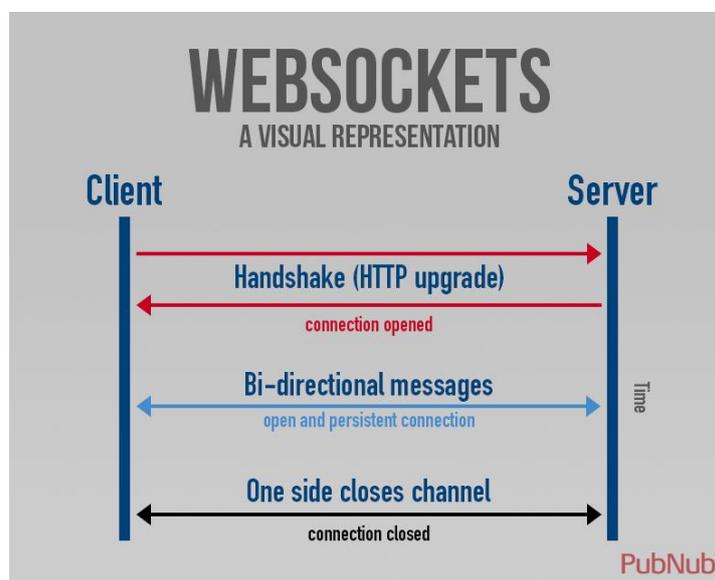


Figura 3.5. WebSockets

En este proyecto se ha utilizado el módulo angular-websocket, creado por AngularClass, para la gestión del servidor de WebSockets.

3.5.2 Justificación de su uso

Esta tecnología es necesaria para poder realizar las comunicaciones entre los usuarios de esta plataforma. Permite que un usuario envíe, por ejemplo, un mensaje de texto al servidor y que este lo redirija a todos los participantes conectados en la misma sala. De este modo, no se necesita que cada cliente mande una petición al servidor para comprobar si existen datos nuevos, lo que implicaría un aumento de las peticiones hacia el servidor y la imposibilidad de realizar comunicaciones en tiempo real.

3.5.3 Fuentes

<https://es.wikipedia.org/wiki/WebSocket> [Accedido por última vez el 09/02/17]

<https://github.com/AngularClass/angular-websocket> [Accedido por última vez el 09/02/17]

<https://www.pubnub.com/blog/2013-09-11-what-are-websockets/> [Accedido por última vez el 09/02/17]

3.6 WebRTC

3.6.1 Descripción

WebRTC es una colección de protocolos de comunicaciones y APIs dirigidos a simplificar el desarrollo de aplicaciones web basadas en comunicaciones en tiempo real. Está basada en la utilización de conexiones peer-to-peer, sin la necesidad de plug-ins, que permite la creación de aplicaciones altamente escalables.

Esta tecnología ha sido creada por Google Inc. y actualmente está siendo estandarizada por la World Wide Web Consortium (W3C) y el Grupo de Trabajo de Ingeniería en Internet (IETF).



Figura 3.6. WebRTC

3.6.2 Justificación de su uso

Esta tecnología permite reducir la sobrecarga del servidor mediante la realización de comunicaciones peer-to-peer. A su vez, simplifica enormemente la realización del sistema de videollamadas de esta plataforma.

3.6.3 Fuentes

<https://webrtc.org/> [Accedido por última vez el 09/02/17]

<https://en.wikipedia.org/wiki/WebRTC> [Accedido por última vez el 09/02/17]

https://es.wikipedia.org/wiki/World_Wide_Web_Consortium [Accedido por última vez el 09/02/17]

https://en.wikipedia.org/wiki/Internet_Engineering_Task_Force [Accedido por última vez el 09/02/17]

<https://www.3cx.es/webrtc/que-es-webrtc/> [Accedido por última vez el 09/02/17]

3.7 EJS (Embedded JavaScript templates)

3.7.1 Descripción

EJS es un lenguaje de etiquetas que nos permite generar código HTML combinando la sintaxis de HTML y la salida de información procesada mediante código JavaScript. Para ello se utilizan las etiquetas “<%” y “%>” para indicar código JavaScript a ejecutar, y las etiquetas “<%=” “%>” para indicar código JavaScript cuya salida se tiene que incluir en el fichero HTML generado.



Figura 3.7. EJS

3.7.2 Justificación de su uso

Este lenguaje nos permite utilizar la misma sintaxis que utilizamos al generar ficheros HTML manualmente, pero además nos da la posibilidad de incluir código JavaScript para ser ejecutado o para incluir su salida directamente en el fichero HTML generado. Además, nos ayuda a depurar de una forma sencilla nuestra aplicación en el caso de que ocurra algún error, ya que te indica la línea donde ocurrió el error y te muestra su traza, tal y como podemos ver en la figura 3.1.

```

ReferenceError: /Users/carlosvillablanca/Workspace/WebApp_Videoconferencias/views/register.ejs:3
 1 | <div>
 2 |
>> 3 |     Hello My name is <%= name %>
 4 |
 5 | </div>

name is not defined
at eval (eval at <anonymous> (/Users/carlosvillablanca/Workspace/WebApp_Videoconferencias/node_modules/ejs/lib/ejs.js:481:12), <anonymous>:11:24)
at returnedFn (/Users/carlosvillablanca/Workspace/WebApp_Videoconferencias/node_modules/ejs/lib/ejs.js:512:17)
at View.exports.renderFile [as engine] (/Users/carlosvillablanca/Workspace/WebApp_Videoconferencias/node_modules/express/lib/view.js:364:31)
at View.render (/Users/carlosvillablanca/Workspace/WebApp_Videoconferencias/node_modules/express/lib/view.js:126:8)
at tryRender (/Users/carlosvillablanca/Workspace/WebApp_Videoconferencias/node_modules/express/lib/application.js:639:10)
at EventEmitter.render (/Users/carlosvillablanca/Workspace/WebApp_Videoconferencias/node_modules/express/lib/application.js:591:3)
at ServerResponse.render (/Users/carlosvillablanca/Workspace/WebApp_Videoconferencias/node_modules/express/lib/response.js:960:7)
at module.exports.register (/Users/carlosvillablanca/Workspace/WebApp_Videoconferencias/server/controllers/controllers.js:22:9)
at Layer.handle [as handle_request] (/Users/carlosvillablanca/Workspace/WebApp_Videoconferencias/node_modules/express/lib/router/layer.js:95:5)
at next (/Users/carlosvillablanca/Workspace/WebApp_Videoconferencias/node_modules/express/lib/router/route.js:131:13)

```

Figura 3.8. Traza generada por un error en un fichero EJS

3.7.3 Fuentes

<http://www.embeddedjs.com/> [Accedido por última vez el 09/02/17]

<http://ejs.co/> [Accedido por última vez el 09/02/17]

3.8 CSS (Cascading Stylesheets)

3.8.1 Descripción

CSS es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado, en este caso, EJS.



Figura 3.9. CSS3

3.8.2 Justificación de su uso

CSS permite cambiar el diseño de las vistas de la aplicación, modificando las formas, colores, tamaños, etc., de sus elementos. Esta es una tecnología esencial para crear interfaces fáciles de entender y utilizar por sus usuarios, y para controlar la respuesta de la interfaz de usuario ante modificaciones en el tamaño de la ventana del navegador.

3.8.3 Fuentes

https://es.wikipedia.org/wiki/Hoja_de_estilos_en_cascada [Accedido por última vez el 09/02/17]

3.9 API REST

3.9.1 Descripción

El término REST proviene de “Representational State Transfer”, un estilo de arquitectura para la comunicación entre clientes y servidores creado por Roy Fielding. Mientras que una API es la abreviatura de “Application programming interface”, y son aquellas subrutinas, funciones y procedimientos que posibilitan las comunicaciones entre componentes software.

Conociendo estos dos conceptos, podemos concluir que una API REST es aquella API Web que debe cumplir unos requisitos impuestos por el estilo de arquitectura REST. Entre estos requisitos encontramos una separación entre cliente y servidor, no guardar estado entre dos peticiones y que las respuestas pueden ser almacenada en cache por el cliente.

Las comunicaciones con la API REST se realizan utilizando el protocolo HTTP.

3.9.2 Justificación de su uso

La creación de una API REST nos permite la separación de la aplicación en clientes y un servidor fácilmente, ya que ni los clientes ni el servidor necesitan saber cómo funciona el otro. Simplemente se intercambian información mediante peticiones y respuestas HTTP. Este hecho es interesante ya que permitiría que otras aplicaciones utilizarasen esta API de forma sencilla.

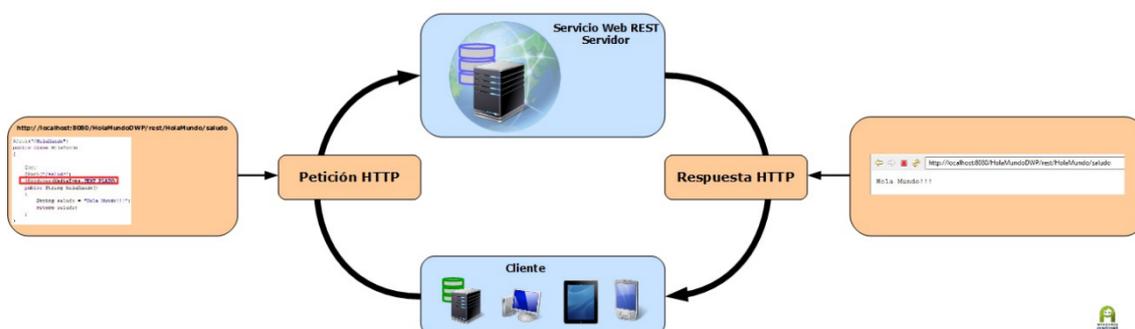


Figura 3.10. Arquitectura REST

3.9.3 Fuentes

<https://www.sitepoint.com/developers-rest-api/> [Accedido por última vez el 09/02/17]

https://en.wikipedia.org/wiki/Representational_state_transfer [Accedido por última vez el 09/02/17]

https://es.wikipedia.org/wiki/Interfaz_de_programaci%C3%B3n_de_aplicaciones [Accedido por última vez el 09/02/17]

<https://academiaandroid.com/servicios-web-arquitectura-rest/> [Accedido por última vez el 09/02/17]

3.10 IntelliJ IDEA Ultimate Edition

3.10.1 Descripción

IntelliJ IDEA Ultimate Edition es un IDE (entorno de desarrollo integrado) desarrollado por JetBrains. Este entorno permite el desarrollo de productos software utilizando diversos lenguajes como JavaScript, Java, HTML, CSS, Scala, Perl, SQL, entre otros. Además de poder integrarse con sistemas de control de versiones como Git o Mercurial.



Figura 3.11. IntelliJ IDEA Ultimate

3.10.2 Justificación de su uso

IntelliJ IDEA es un potente entorno de desarrollo que proporciona interesantes ayudas para el desarrollo de esta aplicación. En primer lugar, simplifica enormemente la creación de proyectos creando un esqueleto con todas las tecnologías necesarias en su última versión. También nos permite seleccionar el lenguaje de marcado y de diseño gráfico que vamos a utilizar tal y como podemos ver en la figura 3.2. De este modo el comienzo de nuestro proyecto se vuelve realmente sencillo.

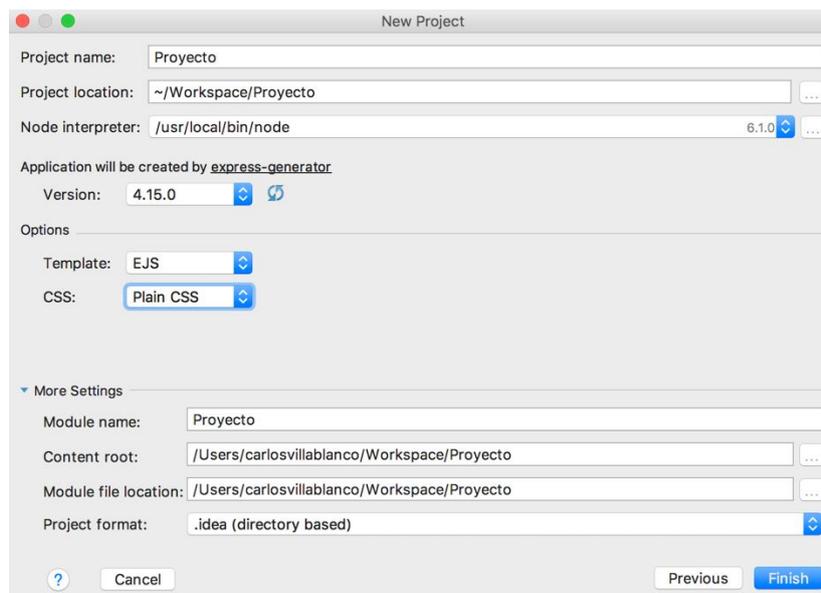


Figura 3.12. Creación de proyecto en IntelliJ IDEA Ultimate Edition

Este entorno nos ofrece un terminal y herramientas para ejecución de pruebas muy útiles para la realización de este proyecto, también posee otras características como autocompletado inteligente de nuestro código, detención de código duplicado, etc.

Este entorno de desarrollo es de pago, sin embargo, gracias a mi condición de estudiante universitario puedo acceder a todas sus funciones de manera gratuita.

3.10.3 Fuentes

<https://www.jetbrains.com/idea/features/> [Accedido por última vez el 09/02/17]

3.11 Bootstrap

3.11.1 Descripción

Bootstrap es un framework desarrollado por Twitter para el diseño de sitios y aplicaciones web. Este framework está pensado para el desarrollo de aplicaciones web con un diseño adaptable de modo que ajusta el diseño de sus componentes a las características de la pantalla del dispositivo.

El framework proporciona una gran variedad de componentes como botones, inputs, menús de navegación, paneles, iconos, etc.



Figura 3.13. Bootstrap

3.11.2 Justificación de su uso

Este framework facilita el desarrollo de aplicaciones web con un diseño adaptable, además de poseer componentes con un diseño muy cuidado. De este modo se pueden crear interfaces profesionales sin tener que dedicar mucho tiempo al desarrollo de estas.

3.11.3 Fuentes

<http://getbootstrap.com/> [Accedido por última vez el 09/02/17]

[https://es.wikipedia.org/wiki/Bootstrap_\(framework\)](https://es.wikipedia.org/wiki/Bootstrap_(framework)) [Accedido por última vez el 09/02/17]

3.12 Gridstack.js

3.12.1 Descripción

Gridstack.js es un plugin de jQuery que permite la creación de interfaces en las que se puede mover, soltar y redimensionar sus elementos. Este plugin es actualmente mantenido por Pavel Reznikov y Dylan Weiss, pero su modelo de código abierto permite la colaboración de otras personas.



Figura 3.14. Gridstack.js

3.12.2 Justificación de su uso

Esta es una herramienta clave para el desarrollo de las salas en las que se comunican los usuarios. Permite mover los diferentes servicios proporcionados por la plataforma alrededor de la interfaz, además de permitir su redimensionamiento.

3.12.3 Fuentes

<https://github.com/troolee/gridstack.js> [Accedido por última vez el 11/02/17]

<https://troolee.github.io/gridstack.js/> [Accedido por última vez el 11/02/17]

3.13 npm

3.13.1 Descripción

La herramienta npm es el gestor de módulos por defecto para Node.js. Permite a los desarrolladores compartir sus módulos, de modo que sean reutilizados en otras aplicaciones. Fue desarrollado por Isaac Z. Schlueter como un proyecto de código abierto.



Figura 3.15. npm

3.13.2 Justificación de su uso

Esta herramienta facilita enormemente la descarga de módulos, además de mantener estos y sus dependencias actualizadas. También facilita realizar la ejecución y testing de la aplicación, de modo que ejecutando “npm start” se ejecutan unos determinados comandos que hemos establecido, mientras que si ejecutamos “npm test” se ejecutan otros. De este modo he conseguido que el sistema de integración continua que utilizo “Travis CI” ejecute unos determinados comandos para realizar las pruebas de la aplicación, mientras que la plataforma “Heroku” ejecuta los necesarios para desplegar la aplicación.

3.13.3 Fuentes

<https://www.npmjs.com/about> [Accedido por última vez el 11/02/17]

3.14 Bower

3.14.1 Descripción

Bower, al igual que npm, es una herramienta para gestión de paquetes. En este caso, Bower está dirigido a gestionar todas aquellas dependencias necesarias para el lado del cliente. Esta herramienta sigue un modelo de desarrollo de código abierto y es desarrollada por Twitter y otros colaboradores.



Figura 3.16. Bower

3.14.2 Justificación de su uso

Al igual que npm, Bower permite descargar y mantener actualizado todas aquellos frameworks, librerías, etc., que son necesarios para el correcto funcionamiento del lado del cliente de Copernicus.

3.14.3 Fuentes

<https://github.com/bower/bower> [Accedido por última vez el 11/02/17]

<https://desarrolloweb.com/articulos/uso-bower-gestor-dependencias.html> [Accedido por última vez el 11/02/17]

3.15 Otros aspectos teóricos

A continuación, se incluye una tabla en la que se describen algunas herramientas, librerías, frameworks, directivas, etc., que, aun siendo necesarias para el correcto funcionamiento y desarrollo de este proyecto, no se les dedicará el mismo grado de detenimiento que los puntos anteriores.

Cabe destacar que en la tabla se especifican las personas que fundaron o crearon la tecnología descrita o sus desarrolladores principales. Sin embargo, muchos de estos proyectos son de código abierto y en ellos han trabajado otros colaboradores.

Nombre	Creador/es	Descripción
angucomplete	Daryl Rowland	Directiva de AngularJS utilizada para la creación de campos de texto con autocompletado. Ha sido utilizado para el autocompletado de nombres de usuario. Un punto importante de esta herramienta
angular-cookies	Google Inc.	Componente de angular utilizado para la gestión de cookies.
angular-growl-v2	Marco Rinck and Jan Stevens	Utilizado para mostrar notificaciones.
angular-translate	The angular-translate team and Pascal Precht	Módulo utilizado para la internacionalización de la aplicación. Copernicus está traducido al español y al inglés.
angular-translate-storage-cookie	The angular-translate team and Pascal Precht	Angular-translate permite modificar el idioma de la aplicación, sin embargo, no almacena datos referentes a cuál fue el último lenguaje escogido. Este servicio utiliza cookieStore como almacenamiento de esta información.
angular-sanitize	Google Inc.	Este módulo analizando el código HTML en tokens, de modo que mediante una lista blanca se buscan tokens seguros. Estos tokens son de nuevo serializados, mientras que se descartan los tokens no seguros.
angular-youtube-mb	Matthew Brandly y Arnaud Breton	Reproductor para videos de YouTube. Debido a problemas encontrados con el plugin para reproducir videos de YouTube en videogular, se decidió contar con un segundo reproductor para poder reproducir estos videos.
Balsamiq Mockup	Giacomo Guilizzoni	Software utilizado para la creación de los prototipos de las interfaces.
Body-parser	Douglas Christopher Wilson y Jonathan Ong	Middleware utilizado para el parseo del cuerpo de las solicitudes HTTP.
Cloudinary	Itai Lahan, Tal Lev-Ami y Nadav Soferman	Servicio en la nube utilizado para el almacenamiento de las fotos de perfil de los usuarios y las fotos de las salas.

Cookie-parser	Douglas Christopher Wilson y TJ Holowaychuk	Middleware utilizado para el parseo de cookies
Crypto	Irakli Gozalishvili	Utilizado para el cifrado de las contraseñas de usuario, mediante la generación de una salt y un hash que se almacenan en la base de datos. También se utiliza para validar un inicio de sesión generando un hash mediante la nueva contraseña introducida y la salt almacenada en la base de datos. Este hash resultante se compara con el hash almacenado en la base de datos y si coinciden el acceso es autorizado.
Dotenv	Scott Motte	Utilizado para cargar las variables de entorno del fichero .env a process.env.
fabric.js	Juriy Zaytsev	Framework utilizado para la creación del servicio de dibujos. Permite la creación y manipulación de figuras en un canvas.
font-awesome	Dave Gandy	Colección de iconos.
Force-ssl	Geza Kovacs	Utilizado para redirigir las peticiones HTTP a HTTPS.
GanttProject	GanttProject Team	Software utilizado para la creación de las planificaciones de este proyecto.
Git	Linus Torvalds	Software de control de versiones
GitHub	Tom Preston-Werner, Chris Wanstrath y PJ Hyett	Forja para el almacenamiento de proyectos basado en el sistema de control de versiones Git.
GrapheneDB	Alberto Perdomo y Juanjo Andres	Se utilizarán dos bases de datos alojadas en servidores de GrapheneDB. Estas bases serán utilizadas para la aplicación en producción y para los tests realizados en Travis CI.
Heroku	Adam Wiggins y James Lindenbaum	Plataforma como servicio de computación en la nube en la cual está desplegado la versión de producción de Copernicus.

jQuery	John Resig	Librería de JavaScript utilizada para la manipulación del árbol DOM, creación de animaciones, manejar eventos, etc.
jQuery UI Touch Punch	David Furfero	Utilizado para mover los servicios por la interfaz en pantallas táctiles.
Jwt-simple	Kazuhiro Hokamura	Módulo para codificación y decodificación JWT (JSON Web Token). Este módulo, junto a una variable de entorno, han sido utilizados para codificar los nombres y los nombres de usuario de cada usuario en un token. Este token se crea y envía al usuario en forma de cookie cada vez que este inicia sesión o se registra, y se utiliza para su identificación. Además, se le ha dado un periodo de validez de un día, de modo que solo sirven para identificar a los usuarios en ese tiempo. Una vez pasado el tiempo es necesario volver a iniciar sesión.
Métrica 3	Ministerio de Administraciones Públicas de España	Métrica 3 es una metodología de planificación, desarrollo y mantenimiento de proyectos software. Se ha utilizado una versión adaptada en esta documentación.
Mocha	JS Foundation	Framework para la realización de pruebas para Node.js.
Moment	JS Foundation	Librería para el análisis, validación, manipulación y formateo de fechas.
Morgan	Douglas Christopher Wilson y Jonathan Ong	Módulo utilizado para poder hacer un seguimiento de las peticiones HTTP realizadas a la aplicación.
ng-file-upload	Danial Farid	Directiva de AngularJS utilizada para permitir a los usuarios cargar sus ficheros en la aplicación.
ng-img-crop	Alex Kaul	Directiva de AngularJS que permite recortar imágenes.
Nodemon	Remy Sharp	Esta herramienta detecta cualquier cambio realizado en el código y, en ese caso, reinicia el servicio. Es muy útil ya que evita que tengamos que reiniciar el servidor manualmente y perder tiempo

		cada que queramos probar un nuevo cambio.
Papertrail	Eric Lindvall y Troy Davis	Gestor de logs utilizado para la aplicación en producción.
Passport	Jared Hanson	Middleware para autenticación de usuarios utilizado junto a crypto.
Passport-local	Jared Hanson	Proporciona una estrategia para la autenticación de usuarios utilizando nombre de usuario y contraseña.
reveal.js	Hakim El Hattab	Framework para la creación de presentaciones HTML.
Seraph	BRIK Teknologier AS	Driver utilizado para realizar consultas contra una base de datos Neo4j.
Serve-favicon	Sencha Inc., LearnBoost, TJ Holowaychuk y Douglas Christopher Wilson	Utilizado para añadir un favicon a la aplicación.
Should (Testing)	TJ Holowaychuk y Denis Bardadym	Librería utilizada para realizar aseercciones en los tests.
Sim Daltonism	Michel Fortin	Herramienta que permite visualizar los colores de las interfaces como aquellas personas con algún tipo de daltonismo.
Supertest (Testing)	TJ Holowaychuk	Librería utilizada para realizar pruebas contra un servidor HTTP.
Travis CI	Konstantin Haase, Mathias Meyer, Fritz Thielemann, Sven Fuchs y Josh Kalderimis	Sistema de integración continua.
videogular	2fdevs	Reproductor de video para HTML5.
videogular-controls	2fdevs	Plugin para videogular que proporciona botones para controlar la reproducción del video.
videogular-themes-default	2fdevs	Temas para videogular.

ws	Einar Otto Stangvik	Librería de WebSocket que proporciona una implementación de un servidor de WebSockets.
----	---------------------	--

Tabla 3.1. Otras tecnologías utilizadas

3.15.1 Fuentes

<https://es.wikipedia.org/wiki/JQuery> [Accedido por última vez el 11/02/17]

<https://api.jquery.com/> [Accedido por última vez el 11/02/17]

<https://www.npmjs.com/> [Accedido por última vez el 12/02/17]

<http://cloudinary.com/about> [Accedido por última vez el 12/02/17]

<https://github.com/alexk111/ngImgCrop> [Accedido por última vez el 12/02/17]

<https://angular-translate.github.io/> [Accedido por última vez el 12/02/17]

<https://github.com/angular-translate/angular-translate> [Accedido por última vez el 12/02/17]

<https://www.crunchbase.com> [Accedido por última vez el 12/02/17]

<https://balsamiq.com/products/mockups/> [Accedido por última vez el 26/02/17]

<https://nodemon.io/> [Accedido por última vez el 22/03/17]

<https://github.com/expressjs/body-parser> [Accedido por última vez el 22/03/17]

<https://neo4j.com/developer/javascript/> [Accedido por última vez el 23/03/17]

<https://github.com/brikteknologier/seraph> [Accedido por última vez el 23/03/17]

<https://github.com/danialfarid/ng-file-upload> [Accedido por última vez el 25/03/17]

<https://github.com/JanStevens/angular-growl-2> [Accedido por última vez el 25/03/17]

<https://github.com/GoZala/crypto> [Accedido por última vez el 26/03/17]

<https://github.com/hokaccha/node-jwt-simple> [Accedido por última vez el 26/03/17]

<https://github.com/shouldjs/should.js> [Accedido por última vez el 26/03/17]

<https://github.com/visionmedia/supertest> [Accedido por última vez el 26/03/17]

<https://github.com/moment/moment> [Accedido por última vez el 27/03/17]

<https://github.com/expressjs/cookie-parser> [Accedido por última vez el 27/03/17]

<https://docs.angularjs.org/api/ngCookies> [Accedido por última vez el 27/03/17]

<https://github.com/darylrowland/angucomplete> [Accedido por última vez el 08/04/17]

<https://github.com/angular-translate/bower-angular-translate-storage-cookie> [Accedido por última vez el 08/04/17]

<https://github.com/expressjs/serve-favicon> [Accedido por última vez el 22/04/17]

<https://github.com/mochajs/mocha> [Accedido por última vez el 23/04/17]

<https://github.com/kangax/fabric.js/> [Accedido por última vez el 04/06/17]

<http://www.videogular.com/> [Accedido por última vez el 10/06/17]

<https://docs.angularjs.org/api/ngSanitize> [Accedido por última vez el 11/06/17]

<https://github.com/brandly/angular-youtube-embed/commits/master> [Accedido por última vez el 14/06/17]

<https://github.com/websockets/ws> [Accedido por última vez el 08/05/17]

<http://www.mmfilesi.com/blog/introduccion-a-ngdocs/> [Accedido por última vez el 16/06/2017]

<https://papertrailapp.com/> [Accedido por última vez el 25/06/2017]

<https://www.heroku.com> [Accedido por última vez el 26/06/17]

<https://www.graphenedb.com/> [Accedido por última vez el 26/06/17]

Capítulo 4. Planificación del Proyecto y Presupuesto Iniciales

4.1 Planificación Inicial

Una vez definido el alcance y los objetivos de este proyecto, se ha definido una planificación inicial para la realización de este proyecto. Esta planificación será seguida durante el desarrollo de la aplicación y finalmente se plasmará en una nueva “planificación” todas aquellas posibles variaciones en los tiempos.

Para poder estimar el tiempo necesario y las funciones que se pueden implementar antes de la defensa de este proyecto, fue necesaria la creación de una planificación inicial usando un diagrama Gantt.

En la figura 4.1 se muestran las tareas planificadas para la realización del proyecto, así como sus fechas previstas de realización.

ID	Nombre	Inicio	Fin
1	Documentación inicial	30/1/17	11/2/17
2	Reunión con director del proyecto	30/1/17	30/1/17
3	Motivación y objetivos	31/1/17	2/2/17
4	Planificación	3/2/17	5/2/17
5	Presupuesto	3/2/17	5/2/17
6	Estudio de la situación actual	6/2/17	7/2/17
7	Busqueda y estudio de las tecnologías	8/2/17	11/2/17
8	Análisis	12/2/17	2/3/17
9	Requisitos del sistema y casos de usos	12/2/17	15/2/17
10	Definición de las clases	16/2/17	20/2/17
11	Interfaces de usuario	21/2/17	25/2/17
12	Plan de pruebas	26/2/17	2/3/17
13	Diseño	3/3/17	12/3/17
14	Arquitectura del sistema y diagramas	3/3/17	7/3/17
15	Diseño de la base de datos	8/3/17	10/3/17
16	Plan de pruebas	11/3/17	12/3/17
17	Implementación	13/3/17	9/6/17
18	Estructura de la aplicación web	13/3/17	25/3/17
19	Inicio de sesión y registro	26/3/17	28/3/17
20	Interfaz modular de las salas	29/3/17	4/4/17
21	Gestión de contactos	5/4/17	19/4/17
22	Gestión de salas	20/4/17	4/5/17
23	Navegación entre las vistas de la aplicación	5/5/17	12/5/17
24	Servicio de videollamadas	13/5/17	20/5/17
25	Servicio de chat de texto	21/5/17	27/5/17
26	Servicio de dibujo	28/5/17	31/5/17
27	Servicio de presentaciones	1/6/17	3/6/17
28	Servicio de audio y video compartido	4/6/17	9/6/17
29	Pruebas	13/3/17	10/6/17
30	Reuniones de seguimiento	30/3/17	1/6/17
31	Muestra de prototipo inicial	30/3/17	30/3/17
32	Reunión plataforma terminada	1/6/17	1/6/17
33	Manuales del sistema	11/6/17	14/6/17
34	Finalización de la documentación	15/6/17	25/6/17
35	Revisión de la documentación	26/6/17	2/7/17
36	Entrega de la documentación	3/7/17	7/7/17
37	Preparación de la defensa	8/7/17	16/7/17
38	Defensa	17/7/17	21/7/17
39	Fin del proyecto	21/7/17	21/7/17

Figura 4.1. Tareas planificación inicial

Como se puede apreciar en la figura 4.1, existen 39 tareas a completar de las cuales 5 son tareas resumen. Antes de realizar esta planificación ya se asistió a una reunión con el director para conocer su propuesta de trabajo y establecer unos primeros objetivos.

Tras la reunión inicial se dedicaron unos días para ampliar los objetivos y definir con claridad cuál es la razón de ser de este proyecto. Una vez definido que se quería construir, ya se pudo comenzar con esta planificación inicial.

Junto con la planificación, se está desarrollando el presupuesto inicial. Posteriormente se realizará un estudio de cuál es la situación actual en el área de las comunicaciones remotas, así como una búsqueda y pequeño estudio de las tecnologías que van a ser necesarias para la realización del proyecto.

Concluida la primera fase de documentación inicial, se comenzará la especificación de requisitos y definición de casos de uso, un primer estudio de las clases necesarias, la creación de los prototipos de las interfaces de usuario y una definición del plan de pruebas.

En la fase de diseño se construirá todos aquellos diagramas necesarios para una correcta implementación de la aplicación, así como el diseño de la base de datos orientada a grafos que se pretende utilizar y una ampliación del plan de pruebas definido en el análisis.

La fase de implementación duraría casi 3 meses, y en ellas se pondrán en práctica la documentación generada en las fases de análisis y diseño. Primero se comenzará por definir la estructura del proyecto, realizando una separación entre las interfaces de usuario, controladores, API REST y servidores. En esta primera tarea también se comenzará la creación de las interfaces de usuario, mediante la creación de ficheros EJS, y la creación de la base de datos, así como los ficheros necesarios para poder realizar la comunicación entre la aplicación y la base. Posteriormente se refinará las interfaces y aquella lógica dirigida a permitir el registro y el inicio de sesión de usuarios.

La tarea con ID 20 es muy importante para cumplir uno de los objetivos principales de este proyecto, la creación de una interfaz simple que permita a sus usuarios modificarla. En esta tarea se dejará preparada la interfaz de las salas en las que los usuarios se podrán comunicar.

En las tareas con ID 21 y 22 se definirán los controladores, el código necesario para enlazar estos controladores con las vistas y las llamadas a la API REST necesarias, para realizar todas las gestiones relativas a los contactos y salas que fueron definidas como requisitos en la fase de análisis. Posteriormente en la tarea 23 se definirán todos los mecanismos necesarios para una correcta navegación entre las diferentes vistas.

De la tarea 24 a la 28 se implementarán los servicios, determinados en los objetivos de la aplicación, para permitir diferentes comunicaciones entre los usuarios.

La tarea de pruebas durará desde el inicio de la fase de implementación, hasta unos días después de la finalización del último servicio. De este modo, mediante un sistema de integración continua se realizarán las pruebas de los nuevos cambios desplegados y al finalizar la implementación se podrán hacer algunas pruebas con usuarios reales. Con esto también se pretende seguir un desarrollo guiado por pruebas (TDD) con las pruebas unitarias. De este modo, antes de implementar la API REST se definen los parámetros que necesitaran sus llamadas y las respuestas que deberían devolver.

También se contemplan en la planificación dos reuniones que se esperan realizar los días 30 de marzo y 1 de junio. La primera reunión consistirá en presentar al director del proyecto un prototipo con el que se le explicará cómo funcionará la aplicación una vez finalizada, y en la siguiente reunión se le mostrará la aplicación ya terminada.

Finalmente, las últimas tareas están dirigidas a la creación de los manuales del sistema, a la finalización de las partes de esta memoria que no se pudiesen haber hecho con anterioridad, a la revisión y entrega de la memoria, y, por último, a la preparación y defensa del proyecto.

4.1.1 Calendario de los recursos de trabajo

Seré la única persona encargada de realizar los roles de analista, arquitecto del software, programador, tester y responsable de documentación. Por ello, solo se definirá un calendario para un recurso de trabajo. Debido a la compaginación del proyecto con las clases de la universidad y con un trabajo, el rango de horas es muy limitado y podría verse afectado debido a imprevistos durante el desarrollo del proyecto.

- Lunes: de 17:00h a 19:00h
- Jueves: de 17:00h a 19:00h
- Sábado: de 09:00h a 14:00h y de 16:00h a 17:00h
- Domingo: de 09:00h a 14:00h

Con este calendario se prevén realizar 15 horas a la semana, completando así alrededor de 364 horas de trabajo en este proyecto. Estas son las horas previstas para la realización del proyecto, aunque, como se explicaba anteriormente, estas pueden variar.

Todas las tareas disponen de un periodo de tiempo para su realización, pero se espera que sean realizadas los días que coincidan con los especificados en este calendario.

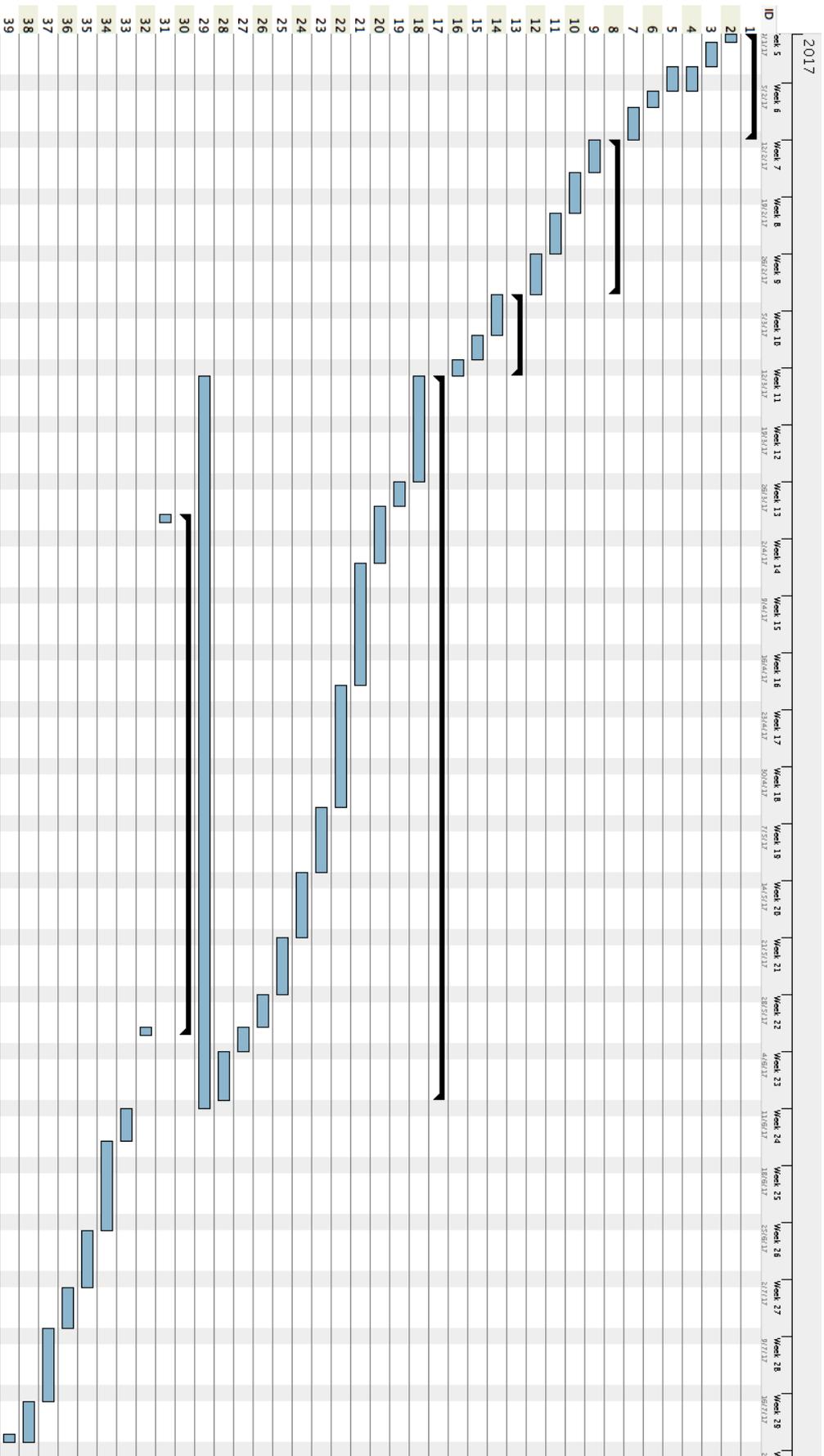


Figura 4.2. Diagrama de Gantt planificación inicial

4.2 Presupuesto Inicial

4.2.1 Presupuesto interno

Para la realización del presupuesto inicial, se tendrá en cuenta las tecnologías que se prevén utilizar, los rangos de tiempo definidos para cada tarea y el calendario que se pretende seguir. A continuación, se muestra el presupuesto inicial.

Item	Concepto	Cantidad	Amortización	Precio unitario (€)	Total (€)
1	Recursos humanos				
1.1	Documentación inicial	25	100%	25,00 €	625,00 €
1.2	Análisis del sistema	39	100%	40,00 €	1.560,00 €
1.3	Diseño del sistema	26	100%	50,00 €	1.300,00 €
1.4	Implementación del sistema	124	100%	30,00 €	3.720,00 €
1.5	Pruebas	66	100%	35,00 €	2.310,00 €
1.6	Finalización del proyecto	84	100%	25,00 €	2.100,00 €
2	Recursos Hardware				
2.1	Portátil	1	1,33%	1.570,00 €	20,88 €
2.2	Ordenador de sobremesa	1	1,33%	800,00 €	10,64 €
2.3	Servidor Heroku	1	100,00%	0,00 €	0,00 €
3	Licencias				
3.1	Windows 10	1	1,33%	279,00 €	3,71 €
3.2	macOS Sierra	1	1,33%	0,00 €	0,00 €
3.3	Microsoft Office 365	6	100,00%	4,20 €	25,20 €
3.4	Neo4j Community Edition	1	100%	0,00 €	0,00 €
3.5	AngularJS 1.6	1	100%	0,00 €	0,00 €
3.6	Node.js v7.5.0	1	100%	0,00 €	0,00 €
3.7	Express 4	1	100%	0,00 €	0,00 €
3.8	Bootstrap v3.3.7	1	100%	0,00 €	0,00 €
3.9	GanttProject v2.8.1	1	100%	0,00 €	0,00 €
3.10	IntelliJ IDEA Ultimate Edition	6	100%	49,90 €	299,40 €
3.11	Otro software gratuito	-	100%	0,00 €	0,00 €
4	Gastos indirectos				
4.1	Internet	6	8,33%	50,00 €	24,99 €
4.2	Electricidad	364	100,00%	0,12 €	43,68 €
4.3	Pizarra blanca	1	2,00%	60,00 €	1,20 €
4.4	Rotuladores Pizarra	4	50,00%	2,00 €	4,00 €
4.5	Papel	100	50,00%	0,07 €	3,25 €
4.6	Lápices	2	50,00%	0,60 €	0,60 €
4.7	Impresora	1	2,00%	80,00 €	1,60 €
4.8	Tóner	1	50,00%	37,00 €	18,50 €
<i>Subtotal</i>					12.072,65 €
<i>Beneficio (10%)</i>					1.207,27 €
Total					13.279,92 €

Tabla 4.1. Presupuesto interno inicial

Los recursos humanos son aquellos trabajos necesarios para completar las tareas del proyecto. La cantidad se especifica en horas, y el precio unitario fue obtenido al realizar un pequeño estudio de los salarios de los siguientes puestos de trabajo:

- Responsable de Documentación: 25 €
- Analista: 40 €
- Arquitecto del software: 50 €
- Programador: 30 €
- Tester: 35 €

Cabe destacar que como las horas dedicadas a las pruebas y a la implementación se solapan, pero solo existe una persona trabajando en el proyecto, durante este periodo las 15 horas semanales se dividirán en 5 horas para la definición de pruebas y 10 para la implementación.

Se prevé que el portátil y el ordenador de sobremesa utilizados (junto a sus sistemas operativos), tengan una vida útil de alrededor de 3 años. Si podrán ser utilizados durante las 364 horas que dura el proyecto (esto es 0.04 años), realizando una regla de tres obtenemos una amortización del 1.33%. En el caso de la impresora y la pizarra blanca, se prevé una vida útil de 2 años y por tanto se definió una amortización del 2%.

Para obtener el costo de Internet, se ha utilizado la duración del proyecto en meses (6 meses), el coste por mes y se han pasado las horas que dura el proyecto a meses. Con una simple regla de tres se puede calcular una amortización del 8,33%.

El monto resultante del consumo eléctrico se ha calculado utilizando la media del coste del kWh en España a febrero de 2017 (0,12€/kWh), y las horas totales que dura la realización del proyecto (364 horas).

El software IntelliJ IDEA Ultimate Edition y Microsoft Office 365 fueron contratados el número de meses que dura el proyecto y solo serán utilizado para este. Por lo tanto, se carga el coste total de ambas suscripciones.

Para los rotuladores, lápices, papel disponible y tóner, se prevé un uso del 50% de las cantidades disponibles. Por lo tanto, su amortización es del 50%.

Finalmente se incluye un beneficio del 10%, ya que se busca obtener un margen de beneficio de al menos 1200€.

4.2.2 Presupuesto para el cliente

El principal objetivo del presupuesto del cliente es que pueda ser entendido por el cliente de forma clara, aunque este no posea ningún conocimiento técnico. Este presupuesto debe ser conciso e incluir el importe total que se planea cobrarle una vez terminado el proyecto. A continuación, se muestra el presupuesto de cliente planteado para la realización de esta aplicación:

Concepto	Coste Total Concepto
<i>Analisis y diseño del sistema</i>	3.276,23 €
<i>Implementación</i>	4.136,23 €
<i>Pruebas</i>	2.726,23 €
<i>Documentación y presentación del proyecto</i>	3.141,23 €
<i>Subtotal</i>	13.279,92 €
<i>IVA (21%)</i>	2.788,78 €
TOTAL	16.068,70 €

Tabla 4.2. Presupuesto cliente inicial

Como se puede observar, se ha simplificado el presupuesto interno de forma que al cliente solo se le presenta 4 conceptos. Lo que se ha hecho es juntar los costos del análisis y diseño en un único concepto, igual que las tareas de la documentación inicial y la finalización del proyecto. Además, se ha dividido proporcionalmente entre todos los conceptos los gastos derivados del hardware, licencias y gastos indirectos, así como el margen de beneficios. Para finalizar, se ha aplicado el IVA del 21% vigente en febrero de 2017.

Capítulo 5. Análisis

5.1 Definición del Sistema

5.1.1 Determinación del Alcance del Sistema

En este proyecto se realizará una aplicación web, internacionalizada en dos idiomas, cuyo objetivo es permitir las comunicaciones entre sus usuarios registrados, los cuales pueden gestionar una lista de contactos y de salas.

Una gran parte de la lógica de esta aplicación será relegada a los controladores, managers, servicios y directivas que serán ejecutados en el lado del cliente. Esto se consigue gracias a AngularJS, y de este modo se disminuirá el trabajo del servidor. En la parte del servidor estarán localizados un servidor HTTP y un servidor WebSockets.

El servidor HTTP se encarga de gestionar peticiones dirigidas a persistir información o a devolver datos a los clientes. Esta gestión de peticiones se realizará mediante una API REST.

El servidor de WebSockets permitirá una comunicación bidireccional entre los clientes y el servidor, y será utilizado durante las comunicaciones entre usuarios registrados. Este servidor permitirá una disminución de peticiones al servidor durante el uso de las salas, ya que los clientes no necesitarán comprobar si existen actualizaciones cada cierto tiempo.

Para explicar de una forma clara el alcance de este proyecto, se dividirá la funcionalidad a implementar en tres apartados.

5.1.1.1 Cuenta de usuario

Una vez el usuario se ha registrado en la aplicación, se le permitirá iniciar y cerrar sesión en la misma. También tendrá la posibilidad de modificar ciertos datos de su cuenta como su nombre, contraseña o foto de perfil. También se le dará la posibilidad de eliminarla.

5.1.1.2 Contactos

Los usuarios registrados podrán gestionar su lista de contactos, enviando, aceptando o ignorando solicitudes de contacto. También tendrán la posibilidad de bloquear y desbloquear contactos.

5.1.1.3 Salas

Se podrán crear salas en las que los usuarios participantes podrán comunicarse entre sí. Estas salas albergarán un máximo de 4 participantes, los cuales podrán acceder a distintas funciones para la gestión de la sala dependiendo de los permisos que posean. Estos permisos pueden ser de “Administrador”, “Moderador” y “Miembro”.

Para que un usuario pueda acceder a una sala, este tiene que aceptar antes una solicitud de unión a la sala. Esta solicitud puede ser ignorada por el usuario, o incluso cancelada por moderadores o el administrador.

En las salas los usuarios podrán comprobar que otros usuarios están conectados y gestionar los servicios de comunicaciones. Se podrá decidir qué servicios se muestran en pantalla, además de poder moverlos y redimensionarlos.

5.1.1.4 Comunicaciones

Los usuarios podrán utilizar seis servicios para realizar diferentes tipos de comunicaciones en las salas. Estos servicios serán de videollamadas, chat de texto, dibujos colaborativos y tres servicios independientes para la compartición de presentaciones, videos y radios o música.

5.2 Requisitos del Sistema

5.2.1 Obtención de los Requisitos del Sistema

5.2.1.1 Requisitos funcionales

En las siguientes tablas se muestran los requisitos funcionales divididos según los roles del usuario. Se muestran tablas donde los requisitos son compartidos por varios roles y otras donde los requisitos solo afectan a un único rol.

5.2.1.1.1 Usuarios no registrados

Código	Nombre Requisito	Descripción del Requisito
RF1.1	Registro	Para realizar el registro el usuario debe proporcionar un nombre, un nombre de usuario que no esté siendo utilizado, un email con formato válido, y una contraseña que se debe introducir dos veces. De forma opcional puede introducir sus apellidos y una foto de perfil la cuál debe tener un tamaño igual o inferior a 8MB. Si no se indica foto de perfil, se le asignará una por defecto.
RF1.2	Iniciar sesión	Se introducirá un nombre de usuario y contraseña para iniciar sesión.
RF1.3	Internacionalización	El idioma de la plataforma podrá ser modificado a español o inglés en la página de inicio de sesión o registro.

Tabla 5.1. Requisitos usuarios no registrados

5.2.1.1.2 Usuarios registrados

Código	Nombre Requisito	Descripción del Requisito
RF2.1	Enviar solicitud de contacto	El usuario podrá enviar solicitudes de contacto a todos los usuarios exceptuando los que estén bloqueados, los que nos tengan bloqueados, los que ya sean contactos del usuario o a los que ya envió una solicitud de contacto. Esta solicitud puede contener un mensaje.
RF2.2	Listado de contactos	Se mostrará un listado con los contactos.
RF2.3	Filtrado de contactos	Se permitirá el filtrado de contactos por nombre en la lista de contactos.

RF2.4	Listado de solicitudes de contacto	Existirá un listado con las solicitudes de contacto recibidas que aún no se han aceptado o ignorado.
RF2.5	Listado de solicitudes de unión a salas	Existirá un listado con las solicitudes de unión a salas recibidas que aún no se han aceptado o ignorado.
RF2.6	Aceptar solicitudes de contacto	Permitir aceptar como contacto al usuario que envió la solicitud de contacto.
RF2.7	Ignorar solicitudes de contacto	Permitir ignorar una solicitud de contacto recibida.
RF2.8	Aceptar solicitudes de unión a salas	Permitir aceptar una solicitud de unión a una sala.
RF2.9	Ignorar solicitudes de unión a salas	Permitir ignorar una solicitud de unión a una sala.
RF2.10	Bloquear contacto	Los usuarios podrán bloquear a sus contactos, de modo que los usuarios bloqueados dejarán de ser parte de sus contactos y entre ambos no se podrán intercambiar solicitudes contacto o de unión a salas. Además, el usuario bloqueado será eliminado de todas las salas donde el usuario que lo bloqueó sea administrador.
RF2.11	Ver listado de contactos bloqueados	Se mostrará un listado con los usuarios bloqueados.
RF2.12	Información contactos	Se podrá acceder a la información básica de los contactos, esto es, nombres, apellidos, emails, nombres de usuario y fotos de perfil.
RF2.13	Desbloquear contacto	Se permitirá el desbloqueo de usuarios bloqueados. De esta forma los usuarios volverán a formar parte del listado de contactos y se podrán volver a intercambiar solicitudes de unión a salas.
RF2.14	Crear sala	Se podrá crear una sala especificando un nombre, descripción, foto para la sala y contactos que participarán y sus permisos. De estos datos únicamente será obligatorio el nombre. En el caso de adjuntar una foto para sala, se comprobará que su tamaño sea menor o igual que 8MB, en caso contrario se le asignará una por defecto.

RF2.15	Listado de salas	Se mostrará un listado con las salas en las que el usuario puede participar.
RF2.16	Acceder a salas	El usuario podrá acceder a las salas pulsando en un botón en el listado de salas.
RF2.17	Filtrado salas	Se permitirá el filtrado de salas por su nombre en el listado de salas en las que el usuario participa.
RF2.18	Enviar solicitudes de unión a sala	Durante la creación de una sala, se permitirá añadir contactos a la misma los cuales recibirán solicitudes de unión a la sala.
RF2.19	Modificar datos de la cuenta	Se permitirá modificar el nombre, apellidos, email, contraseña y foto de perfil del usuario. Se validará que el nombre no sea vacío, que el email tenga un formato correcto y que la foto de perfil sea de un tamaño menor o igual que 8MB. En el caso de cambiar la contraseña se pedirá la contraseña antigua e introducir la nueva contraseña dos veces.
RF2.20	Eliminar cuenta	El usuario podrá eliminar su cuenta.
RF2.21	Internacionalización	El idioma de la aplicación podrá ser modificado a inglés o español en la página principal, en la página de gestión de salas y en la página de ajustes de la cuenta.
RF2.22	Cerrar sesión	El usuario registrado podrá cerrar sesión, de modo que pasará a ser un usuario no registrado y será devuelto a la página de inicio de sesión y registro.

Tabla 5.2. Requisitos usuarios registrados

5.2.1.1.3 Todos los usuarios registrados que participen en una sala (Miembros, moderadores o administradores)

Código	Nombre Requisito	Descripción del Requisito
RF3.1	Listado de participantes de la sala	Se mostrará un listado con los usuarios que participan en la sala con sus permisos.
RF3.2	Listado de participantes conectados en la sala	Se mostrará un listado con los usuarios que participan en la sala que están conectados.
RF3.3	Abrir servicio	Con un solo clic se mostrará el servicio seleccionado en la sala.

RF3.4	Cerrar servicio	Con un solo clic se cerrará el servicio seleccionado en la sala.
RF3.5	Mover servicio	Se permitirá mover servicios en un espacio reservado para ello de la sala.
RF3.6	Redimensionar servicio	Todos servicios abiertos, exceptuando el de radio, podrán ser redimensionados.
RF3.7	Videollamadas	Participar en videollamadas con otros participantes de la sala.
RF3.8	Parar video de la videollamada	Se permitirá parar la transmisión de video a otros usuarios.
RF3.9	Silenciar micrófono	Se permitirá parar la transmisión de audio a otros usuarios.
RF3.10	Silenciar otros usuarios	Se permitirá silenciar a todos los usuarios que participan en la videollamada.
RF3.11	Mensajes de texto	Enviar mensajes de texto a otros participantes de la sala.
RF3.12	Compartir presentación	Compartir presentación con formato HTML o PDF con los usuarios de una sala que estén conectados.
RF3.13	Compartir video	Compartir URL de video de YouTube u otro video con formato MP4, WebM o OGV, de modo que el usuario que lo envía y los usuarios conectados en la sala lo pueden reproducir en el servicio de video compartido.
RF3.14	Controlar video	El video compartido podrá ser pausado, reanudado, adelantado, retrasado, abierto en pantalla completa y su volumen alterado.
RF3.15	Enviar ficheros	Se podrá enviar y descargar ficheros a través del servicio de chat de texto. Los ficheros no pueden tener un tamaño mayor a 10MB.
RF3.16	Feedback chat de texto	Se mostrará a través de una numeración al lado de los mensajes o mediante notificaciones en el caso de ficheros enviados el número de usuarios conectado en la sala que recibieron el mensaje o fichero.
RF3.17	Compartir URL de radio o canción	Compartir URL de una radio o música.

RF3.18	Controlar radio o canción	El audio podrá ser pausado y su volumen alterado.
RF3.19	Crear dibujo colaborativo	Los usuarios que utilicen el servicio de dibujos podrán añadir formas y moverlas por el servicio, hacer dibujos con el ratón y borrar todos los cambios realizados. Estos cambios serán comunicados a todos los usuarios que estén usando el servicio.

Tabla 5.3. Requisitos usuarios que participan en una sala

5.2.1.1.4 Usuarios registrados que sean miembros o moderadores de cierta sala

Código	Nombre Requisito	Descripción del Requisito
RF4.1	Abandonar sala	Se da la posibilidad de salir de las salas. De esta forma la sala dejará de formar parte del listado de salas.

Tabla 5.4. Requisitos miembros y moderadores

5.2.1.1.5 Usuarios registrados que sean administradores o moderadores

Código	Nombre Requisito	Descripción del Requisito
RF5.1	Eliminar solicitudes de unión a salas	Las solicitudes de unión a salas podrán ser eliminadas antes de que el usuario la acepte o ignore.
RF5.2	Cambiar los permisos concedidos en las solicitudes de unión a salas	Se podrán cambiar los permisos concedidos en las solicitudes de unión a salas a miembro o moderador.
RF5.3	Eliminar miembros de salas	Se podrá eliminar miembros de las salas.
RF5.4	Modificar permisos de miembros	Los permisos de los miembros podrán ser aumentados a permisos de moderador.
RF5.5	Modificar datos de la sala	El nombre, descripción y foto de la sala podrán ser modificados. Se validará que el nombre de la sala no está vacío y que la foto tenga un tamaño menor o igual que 8MB.

Tabla 5.5. Requisitos administradores y moderadores

5.2.1.1.6 Usuarios registrados que sean administradores de cierta sala

Código	Nombre Requisito	Descripción del Requisito
RF6.1	Eliminar sala	Existirá una opción en la vista de gestión de salas que permitirá eliminar una sala. Al eliminar la sala todos los usuarios que participan en ella serán eliminados de la sala. Al proceder a eliminar la sala, se mostrará una ventana emergente en el que se preguntará si está seguro de eliminar la sala.
RF6.2	Modificar permisos de moderadores	El administrador de la sala podrá modificar los permisos de los moderadores, de modo que pasen a tener permisos de miembro.
RF6.3	Eliminar moderadores de salas	El administrador de la sala podrá eliminar moderadores de las salas.

Tabla 5.6. Requisitos administradores

5.2.1.2 Requisitos no funcionales

5.2.1.2.1 Requisitos tecnológicos

Código	Nombre Requisito	Descripción del Requisito
RNF2.1	Navegadores	La aplicación web funcionará correctamente en aquellos navegadores de escritorio que soporten WebRTC.

Tabla 5.7. Requisitos tecnológicos

5.2.1.2.2 Requisitos de usabilidad

Código	Nombre Requisito	Descripción del Requisito
RNF3.1	Notificaciones	Se mostrarán notificaciones en la esquina superior derecha de la interfaz.

Tabla 5.8. Requisitos de usabilidad

5.2.1.2.3 Requisitos de seguridad

Código	Nombre Requisito	Descripción del Requisito
RNF4.1	Contraseña	Las contraseñas no se almacenarán en la base de datos, sino que se generará un hash y una salt que será lo que se persiste.
RNF4.2	Protocolo HTTPS	Las peticiones HTTP serán redirigidas a HTTPS.

Tabla 5.9. Requisitos de seguridad

5.2.2 Identificación de Actores del Sistema

Se han identificado dos actores en el sistema, estos son:

- Usuario no registrado: Cualquier actor que interactúa con la aplicación sin haber iniciado sesión. Solo puede acceder a la página de inicio de sesión y registro.
- Usuario registrado: Aquel actor que ha iniciado sesión. Este podrá navegar por todas las vistas de la aplicación exceptuando la de inicio de sesión y registro.

Los usuarios registrados a su vez pueden tener diferentes roles de cara a la gestión de una sala concreta a la que pertenecen. Estos roles son:

- Miembro: Este usuario puede utilizar los servicios de la sala para comunicarse con otros usuarios conectados en esa misma sala.
- Moderador: Además de los permisos del miembro, este usuario puede editar el nombre, descripción y foto de la sala, enviar solicitudes de unión a la sala o eliminarlas, y eliminar a miembros de la sala.
- Administrador: Además de las funcionalidades del moderador, el administrador puede cambiar los permisos de los moderadores y miembros, eliminar moderadores de la sala y eliminar la sala.

5.2.3 Especificación de Casos de Uso

5.2.3.1 Casos de uso usuario no registrado

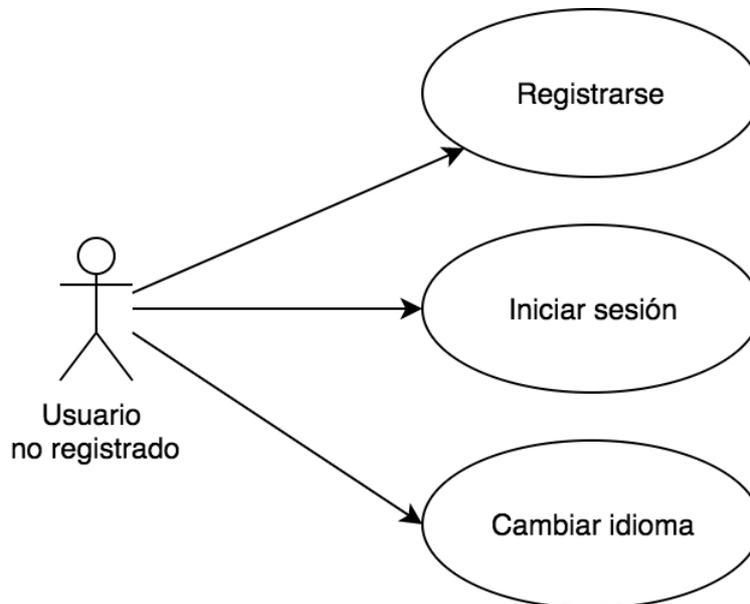


Figura 5.1. Caso de uso de usuario no registrado

Nombre del Caso de Uso
Registrarse
Descripción
<p>El usuario no registrado podrá registrarse introduciendo la siguiente información:</p> <ul style="list-style-type: none"> - Nombre - Apellidos - Nombre de usuario - Email - Contraseña - Foto de perfil <p>De estos datos son obligatorios el nombre, nombre de usuario, email y contraseña. El usuario deberá introducir dos veces la contraseña.</p>

Nombre del Caso de Uso
Iniciar sesión
Descripción
<p>El usuario no registrado podrá iniciar sesión introduciendo su nombre de usuario y contraseña.</p>

Nombre del Caso de Uso
Cambiar idioma
Descripción
El usuario no registrado podrá cambiar el idioma de la aplicación en la página de inicio de sesión y registro pulsando en el botón con la bandera que represente el idioma que desee.

5.2.3.2 Casos de uso usuario registrado

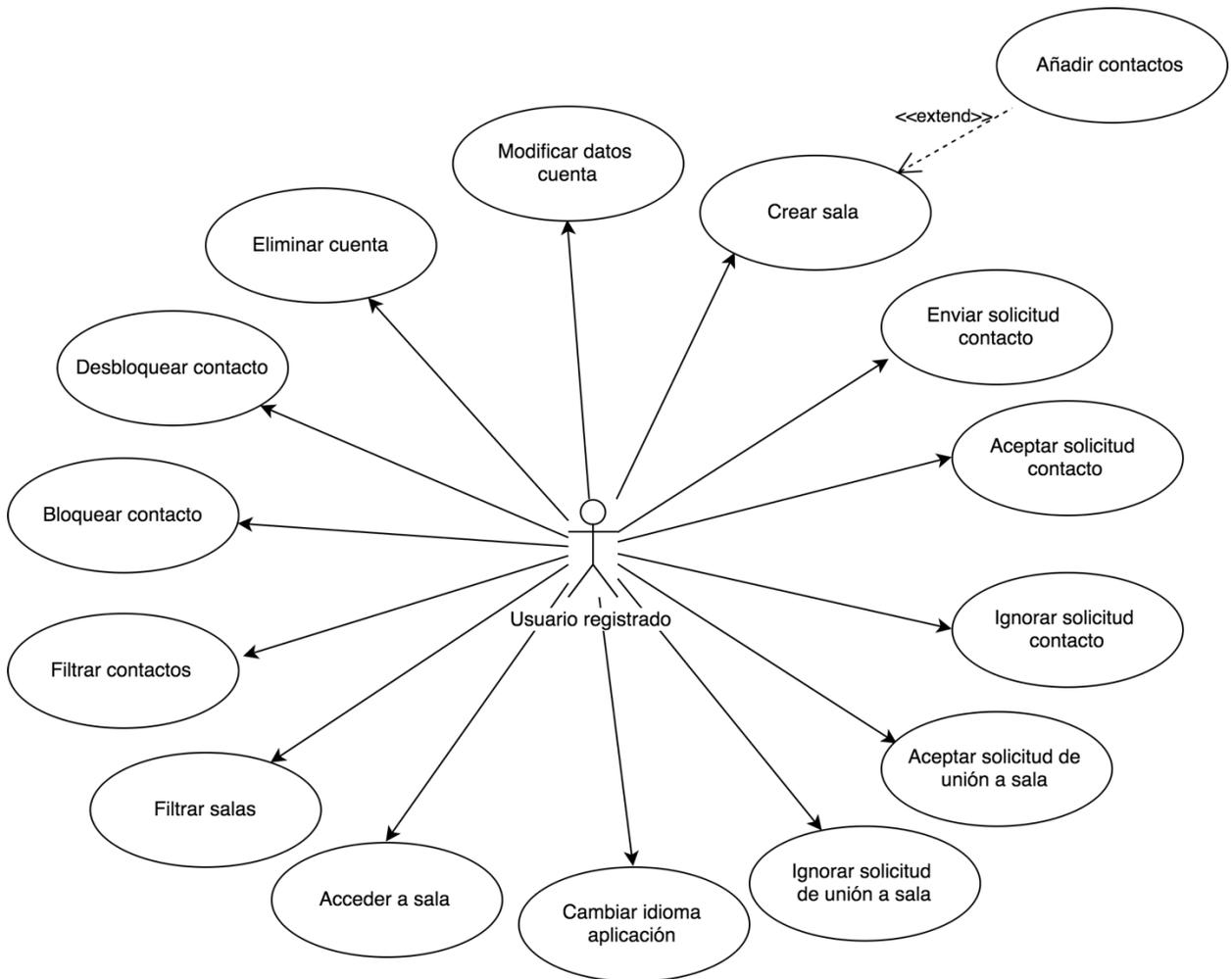


Figura 5.2. Caso de uso de usuario registrado

Nombre del Caso de Uso
Crear sala
Descripción
El usuario registrado podrá crear una sala especificando los siguientes datos:
<ul style="list-style-type: none"> - Nombre - Descripción - Foto de la sala

- Contactos que participarán en la sala y sus permisos

De estos datos solo es obligatorio el nombre.

Nombre del Caso de Uso

Añadir contactos

Descripción

Durante la creación de una sala el usuario registrado podrá especificar los contactos que participarán en la sala y sus permisos utilizando un campo de texto con autocompletar. En este campo de texto especificará un nombre de usuario o parte de él y se mostrará un listado de contactos cuyo nombre de usuario contiene el texto especificado. Se seleccionará un contacto de los mostrados y, pulsando un botón, se añadirá a un listado de contactos a los cuales se enviarán una solicitud de unión una vez creada la sala. En ese listado se puede cambiar los permisos del contacto o eliminarlo.

Nombre del Caso de Uso

Enviar solicitud contacto

Descripción

El usuario registrado podrá enviar una solicitud de contacto junto a un mensaje opcional. Para ello se especificará en un campo de texto un nombre de usuario o parte de él y se mostrará un listado de los contactos cuyo nombre de usuario contiene el texto especificado, de ese listado se seleccionará un contacto. Finalmente se enviará la solicitud de contacto al pulsar un botón.

Nombre del Caso de Uso

Aceptar solicitud contacto

Descripción

El usuario registrado podrá acceder a un listado de solicitudes de contacto recibidas. En ese listado se podrá pulsar un botón para aceptar cada solicitud de forma independiente.

Nombre del Caso de Uso

Ignorar solicitud contacto

Descripción

El usuario registrado podrá acceder a un listado de solicitudes de contacto recibidas. En ese listado se podrá pulsar un botón para ignorar cada solicitud de forma independiente.

Nombre del Caso de Uso
Aceptar solicitud de unión a sala
Descripción
El usuario registrado podrá acceder a un listado de solicitudes de unión a salas recibidas. En ese listado se podrá pulsar un botón para aceptar cada solicitud de forma independiente.

Nombre del Caso de Uso
Ignorar solicitud de unión a sala
Descripción
El usuario registrado podrá acceder a un listado de solicitudes de unión a salas recibidas. En ese listado se podrá pulsar un botón para ignorar cada solicitud de forma independiente.

Nombre del Caso de Uso
Acceder a sala
Descripción
El usuario registrado dispondrá de botones para cada sala en su listado de salas, los cuales pulsará para acceder a las salas.

Nombre del Caso de Uso
Filtrar salas
Descripción
En los listados de salas se proporcionarán unos campos en los que al introducir texto se modificarán automáticamente los contactos que se muestran en el listado en función de si contienen o no el texto introducido en su nombre.

Nombre del Caso de Uso
Filtrar contactos
Descripción
En el listado de contactos se proporcionarán unos campos en los que al introducir texto se modificarán automáticamente los contactos que se muestran en el listado en función de si contienen o no el texto introducido en su nombre de usuario.

Nombre del Caso de Uso
Bloquear contacto
Descripción
El usuario registrado puede bloquear un contacto de dos formas distinta:

- Pulsando en un botón en el listado de contactos, se mostrará una ventana en la que se permitirá bloquear al contacto.
- El usuario podrá acceder a una ventana en la que habrá un campo para introducir un nombre de usuario o parte de él. Al introducir texto en el campo irán apareciendo en un listado contactos cuyo nombre de usuario contiene el texto introducido. Una vez seleccionado un contacto de esa lista, el usuario pulsará un botón y el contacto sería bloqueado.

Nombre del Caso de Uso

Desbloquear contacto

Descripción

En la misma ventana donde el usuario registrado puede bloquear contactos introduciendo su nombre de usuario, se mostrará un listado de usuarios bloqueados que se podrá desbloquear pulsando un botón para cada uno de ellos.

Nombre del Caso de Uso

Eliminar cuenta

Descripción

El usuario registrado pulsará un botón en la página de ajustes de la cuenta para eliminar su cuenta. Una vez pulsado el botón, se mostrará una ventana emergente en la que se confirmará si de verdad se quiere eliminar la cuenta.

Nombre del Caso de Uso

Modificar datos cuenta

Descripción

El usuario registrado podrá modificar la siguiente información de su cuenta en la página de ajustes de cuenta:

- Nombre
- Apellidos
- Email
- Contraseña
- Foto de perfil

Todos los campos deben tener algún valor exceptuando los apellidos.

5.2.3.3 Casos de uso compartidos por miembro, moderador y administrador

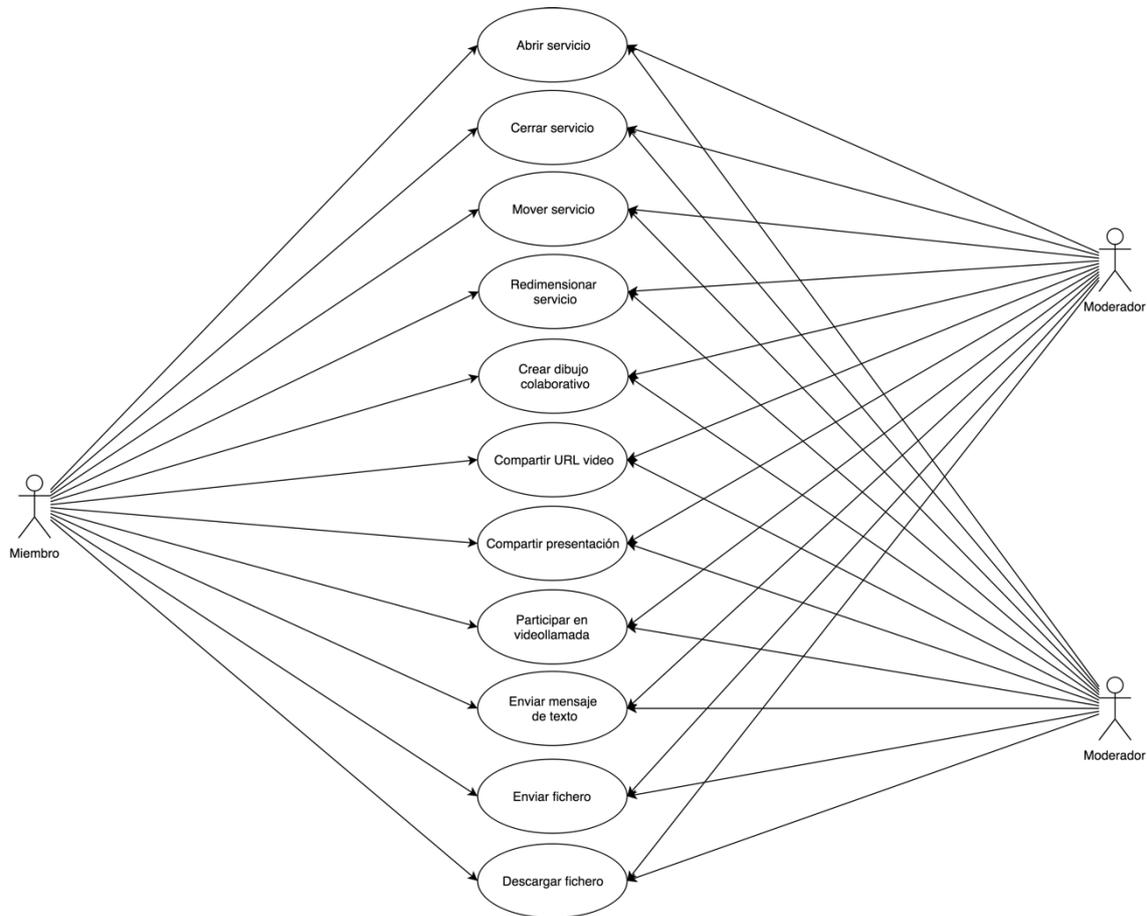


Figura 5.3. Caso de uso compartidos por miembro, moderador y administrador

Nombre del Caso de Uso
Abrir servicio
Descripción
El usuario clicará en el servicio que desea abrir en un listado de servicios disponibles.

Nombre del Caso de Uso
Cerrar servicio
Descripción
El usuario clicará en una “X” posicionada en la esquina superior derecha del servicio abierto para cerrarlo.

Nombre del Caso de Uso
Mover servicio
Descripción
El usuario mantendrá pulsado en cualquier zona de un servicio o en una zona indicada para ello lo que le permitirá mover el servicio mientras mueve el puntero.

Nombre del Caso de Uso
Redimensionar servicio
Descripción
El usuario mantendrá pulsado sobre un icono en la esquina inferior derecha del servicio lo que le permitirá redimensionarlo mientras mueve el puntero. Esto será válido para aquellos servicios que permitan ser redimensionados.

Nombre del Caso de Uso
Crear dibujo colaborativo
Descripción
Una vez abierto el servicio de dibujos el usuario podrá añadir figuras, realizar dibujos a mano alzada, moverlos y redimensionarlos por el lienzo, y borrarlos todo pulsando unos botones. Los cambios realizados sobre el lienzo serán transmitidos a los usuarios que estén utilizando el servicio de dibujos.

Nombre del Caso de Uso
Compartir URL video
Descripción
Una vez abierto el servicio de video compartido, el usuario añadirá una URL de un video alojado en YouTube u otro servicio (siempre que tengan un formato MP4, WebM o OGV) y al pulsar un botón la URL será enviada al reproductor de todos los usuarios conectados y al del propio usuario que la envió.

Nombre del Caso de Uso
Compartir presentación
Descripción
Una vez abierto el servicio de presentaciones, el usuario pulsará un botón que le permitirá subir una presentación en formato HTML o un fichero PDF, el cual será enviado a todos los usuarios conectados.

Nombre del Caso de Uso
Participar en videollamada
Descripción
Al abrir el servicio de videollamada el usuario comenzará la transmisión de video y audio a otros usuarios en la videollamada, y recibirá el video y audio de estos últimos. El usuario podrá controlar la transmisión de video y audio, así como el audio que recibe mediante tres botones.

Nombre del Caso de Uso
Enviar mensaje de texto
Descripción
Una vez abierto el servicio de chat de texto, el usuario podrá mandar mensajes escribiéndolos en un campo de texto y pulsando la tecla Enter o un botón.

Nombre del Caso de Uso
Enviar fichero
Descripción
Una vez abierto el servicio de chat de texto, el usuario podrá pulsar un botón el cual abrirá una ventana emergente donde se podrá seleccionar el archivo que quiere enviar.

Nombre del Caso de Uso
Enviar fichero
Descripción
El usuario podrá descargar los ficheros recibidos a través del chat de texto pulsando sobre ellos.

5.2.3.4 Casos de uso compartidos por miembro y moderador

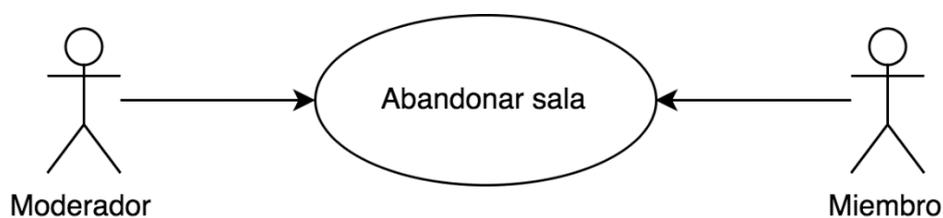


Figura 5.4. Casos de uso compartidos por miembro y moderador

Nombre del Caso de Uso
Abandonar sala
Descripción
El moderador y miembro podrán abandonar una sala pulsando un botón. Una vez pulsado ese botón aparecerá una ventana emergente para asegurar que de verdad quieren abandonar la sala.

5.2.3.5 Casos de uso compartidos por moderador y administrador

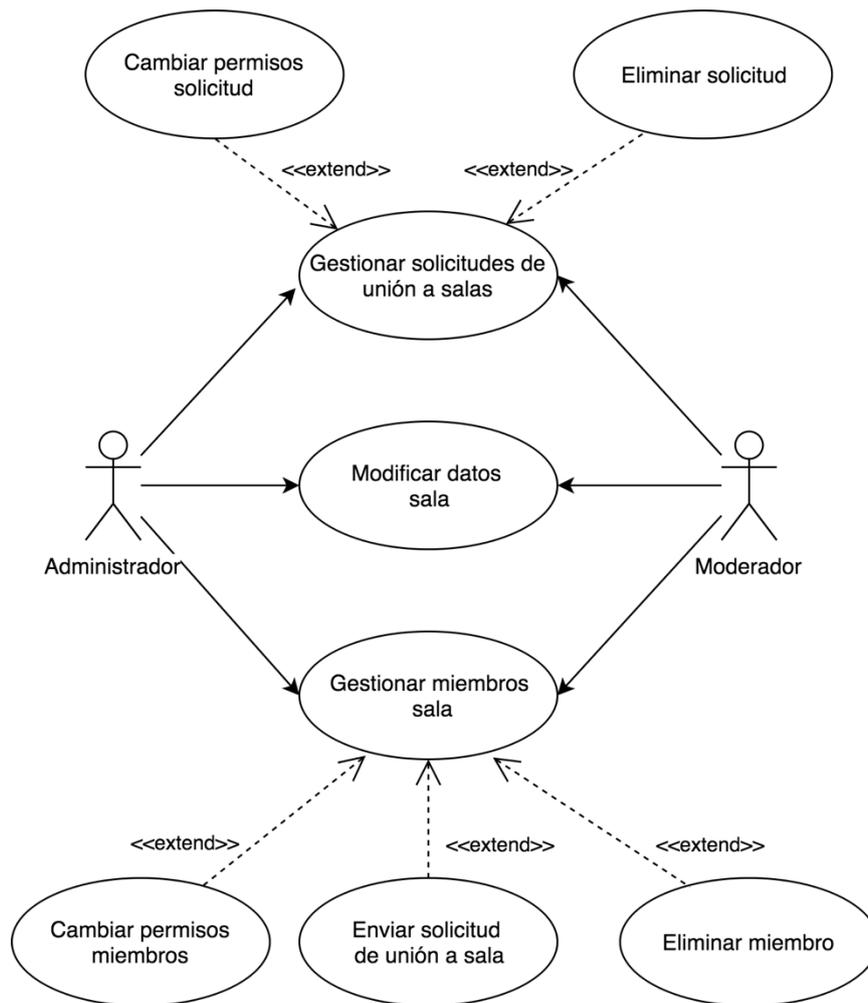


Figura 5.5. Casos de uso compartidos por administrador y moderador

Nombre del Caso de Uso
Gestionar solicitudes de unión a salas
Descripción
El administrador y moderador podrán acceder a un listado de las solicitudes de unión a salas pendientes de resolución para gestionarlas.

Nombre del Caso de Uso

Cambiar permisos solicitud

Descripción

El administrador y moderador podrán cambiar los permisos concedidos especificados en las solicitudes de unión a salas. Estos permisos son “Miembro” o “Moderador”.
--

Nombre del Caso de Uso

Eliminar solicitud

Descripción

El administrador y moderador podrán eliminar las solicitudes de unión a salas enviadas, antes de que el usuario pueda aceptarla o ignorarla.
--

Nombre del Caso de Uso

Modificar datos sala

Descripción

El administrador y moderador podrán modificar los datos de las salas. Estos datos son:
--

- Nombre
- Descripción
- Foto de la sala

Nombre del Caso de Uso

Gestionar miembros salas

Descripción

El administrador y moderador podrán acceder a un listado con los miembros que participan en la sala y gestionarlos.

Nombre del Caso de Uso

Enviar solicitud de unión a sala

Descripción

El administrador y moderador podrán enviar solicitudes de unión a la sala cuando la sala ya está creada. Para ello especificarán en un campo de texto el nombre de usuario y este campo desplegará un listado de usuarios. De este listado se seleccionará el que contacto que se desee y, después de dar a un botón, se le enviará una solicitud de unión a la sala como miembro.
--

Nombre del Caso de Uso

Eliminar miembro

Descripción

El administrador y moderador podrán eliminar miembros de la sala pulsando un botón en el listado con miembros.
--

Nombre del Caso de Uso
Cambiar permisos miembro
Descripción
El administrador y moderador podrán cambiar los permisos de los miembros a “Moderador”.

5.2.3.6 Casos de uso administrador

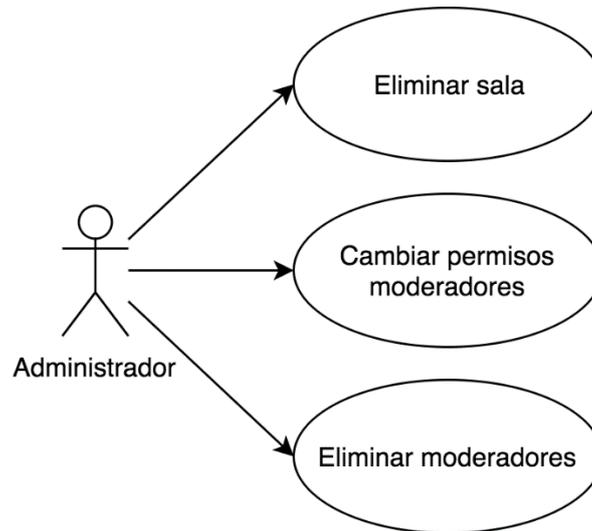


Figura 5.6. Caso de uso administrador

Nombre del Caso de Uso
Eliminar sala
Descripción
El administrador podrá eliminar una sala pulsando un botón. Al pulsar el botón se mostrará una ventana emergente para confirmar que se quiere eliminar la sala.

Nombre del Caso de Uso
Cambiar permisos moderadores
Descripción
El administrador podrá cambiar los permisos de los moderadores a “Miembro”.

Nombre del Caso de Uso
Eliminar moderadores
Descripción
El administrador podrá eliminar moderadores de la sala.

5.3 Identificación de los Subsistemas en la Fase de Análisis

5.3.1 Descripción de los Subsistemas

La aplicación estará dividida en 4 grandes subsistemas, en la parte del servidor nos encontraremos con los subsistemas “Servidor Web”, “Servidor WebSockets” y “Capa de persistencia”, mientras que un cuarto subsistema “Cliente” se localizará en la parte del cliente. A continuación, se explicarán estos subsistemas:

- **Servidor Web:** Todos los scripts de JavaScript relativos al funcionamiento del servidor HTTP.
- **Servidor WebSockets:** Todos los scripts de JavaScript relativos al funcionamiento del servidor de WebSockets.
- **Capa de persistencia:** Estará formado por aquellos controladores, modelos y otros scripts JavaScript necesarios para la gestión y respuesta de peticiones HTTP mediante una API REST, además de para la persistencia de la información en la base de datos Neo4j.
- **Cliente:** Todos los ficheros EJS que formarán la vista de la aplicación. También incluirá los controladores, directivas, servicios, módulos y otros scripts JavaScript que serán alojados en el cliente y necesarios para obtener información del usuario y para mostrarle la información obtenida de la base de datos.

5.3.2 Descripción de los Interfaces entre Subsistemas

La comunicación entre el subsistema “Cliente” y “Capa de persistencia” será posible mediante la utilización de una API REST. Los controladores del subsistema “Cliente” enviarán peticiones HTTP a las que la API REST localizada en el subsistema “Capa de persistencia” responderá.

El subsistema “Cliente” se comunicará mediante peticiones HTTP con el servidor HTTP localizado en el subsistema “Servidor web”. De esta forma se podrá realizar la navegación entre las vistas de forma programática. El subsistema “Cliente” también se comunicará de forma bidireccional con el subsistema “Servidor WebSockets”.

También cabe destacar como se realizará la comunicación entre los distintos componentes en el subsistema “Cliente”. Las comunicaciones entre los ficheros EJS y controladores

serán posibles gracias al objeto \$scope que ofrece AngularJS. Este objeto puede ser inyectado en los controladores y en las vistas proporcionándonos así una sincronización entre los valores en los controladores y los mostrados en las vistas. A este hecho se le conoce como *Two-way Binding*.

5.4 Diagrama de Clases Preliminar del Análisis

Una vez definidos los subsistemas de la aplicación, podemos pasar a realizar un diseño inicial del diagrama de clases.

Debido a que este diagrama se realiza en una fase temprana del desarrollo puede que sea necesario posteriormente la inclusión de nuevas clases, métodos o atributos o la actualización de los existentes debido a factores que difícilmente pueden ser previstos inicialmente.

5.4.1 Diagrama de Clases

Como se puede observar en el siguiente diagrama, existirán cuatro subsistemas independientes entre sí, ya que el subsistema “Cliente” se comunicará con el subsistema “Capa de persistencia” y “Servidor web” mediante peticiones HTTP y con el subsistema “Servidor WebSockets” mediante mensajes de WebSockets.

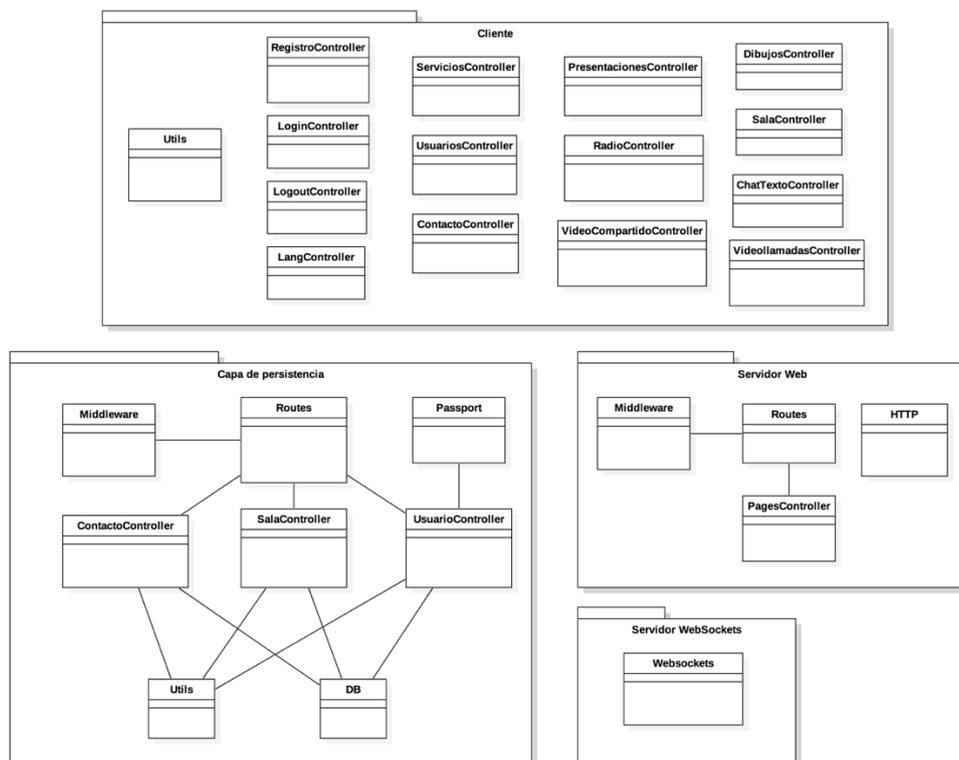


Figura 5.7. Diagrama de clases preliminar

En el subsistema “Cliente” se han identificado varios controladores para gestionar las salas, usuarios y también los servicios. Es muy probable la existencia de funciones compartidas entre todos los controladores y, por ello, se ha definido una clase “Utils” en la que se incluirá todo ese código repetido. Esta clase “Utils” estará relacionada con los controladores del subsistema, en este diagrama no se han dibujado estas relaciones para no complicar su lectura.

El subsistema “Capa de persistencia” dispondrá de una clase “Routes” que será la encargada de redirigir las peticiones realizadas por los clientes al respectivo controlador encargado de solucionarlas. La clase “Middleware” incluirá comprobaciones a realizar antes de que la petición sea enviada al controlador. Estas comprobaciones se realizarán en esta clase y no en los controladores, ya que puede haber comprobaciones que se repitan en distintas peticiones. También existirá una clase “DB” en la que se incluirá toda la información relativa a la conexión con la base de datos. Finalmente, encontramos una clase “Utils” para incluir todo el código compartido por los controladores.

En el subsistema “Servidor Web” se incluyen una clase en las que se definirán la información del servidor HTTP. Esta clase no tiene ninguna relación ya que será utilizada por Express JS para la configuración del servidor. También encontramos una clase “Routes” encarga de recibir las peticiones HTTP referentes a la obtención de las vistas de la aplicación web, y que redirigirá a “PagesController”. Al igual que ocurre en el subsistema “Capa de persistencia”, “Routes” utilizará una clase “Middleware” para realizar las comprobaciones oportunas y decidir así qué página servir al cliente. Estas comprobaciones serán, por ejemplo, comprobar si se envía la cookie de sesión necesaria para acceder a cierta página. “PagesController” será la clase encargada de devolver una respuesta indicando la vista a visualizar.

5.4.2 Descripción de las Clases

5.4.2.1 Cliente

Nombre de la Clase
SalaController
Descripción
Controlador encargado de sincronizar las vistas e información persistida que sea relativa a las salas.
Responsabilidades
Entre sus responsabilidades se encontrará: <ul style="list-style-type: none"> - Permitir la creación, modificación y eliminación de salas - Salir de salas - El envío y gestión de solicitudes de unión a salas - Proporcionar acceso a las salas - Filtrar salas

Atributos Propuestos
contactosSala: Contendría los contactos de una sala.
Métodos Propuestos
<p>crearSala: Utilizado para crear una nueva sala.</p> <p>actualizarSala: Utilizado para modificar los datos de una sala.</p> <p>accederSala: Utilizado por un participante para acceder a una sala.</p> <p>mostrarInfoSala: Muestra la información de una sala en función de los permisos poseídos.</p> <p>enviarSolicitudSala: Utilizado para enviar solicitudes para unirse a la sala.</p> <p>eliminarSolicitudSala: Utilizado para eliminar solicitudes para unirse a una sala enviadas.</p> <p>aceptarSolicitud: Utilizado por un usuario para aceptar una solicitud de unión a una sala.</p> <p>ignorarSolicitud: Utilizado por un usuario para ignorar una solicitud de unión a una sala recibida.</p> <p>cambiarPermisosSolicitud: Modifica los permisos otorgados en una solicitud de unión a una sala.</p> <p>cambiarPermisosParticipantes: Modifica los permisos otorgados en una solicitud de unión a una sala.</p> <p>eliminarUsuario: Elimina un usuario de una sala.</p> <p>salirSala: Utilizado por un usuario para dejar de participar en una sala.</p> <p>eliminarSala: Utilizado por un usuario para eliminar una sala.</p> <p>filtrarSalas: Utilizado para filtrar las salas mostradas en un listado.</p>

Nombre de la Clase
UsuarioController
Descripción
Controlador encargado de la gestión de la cuenta del usuario.
Responsabilidades
Entre sus responsabilidades se encontrará:
<ul style="list-style-type: none"> - Modificar información de la cuenta del usuario
Atributos Propuestos
Métodos Propuestos
<p>actualizarDatos: Actualiza los datos de la cuenta del usuario.</p> <p>actualizarPassword: Actualiza la contraseña del usuario.</p>

Nombre de la Clase
ContactoController
Descripción
Controlador encargado de la gestión de contactos y envío de solicitudes de contacto.
Responsabilidades
Entre sus responsabilidades se encontrará:

<ul style="list-style-type: none"> - Gestionar solicitudes de contacto - Bloquear y desbloquear usuarios - Filtrar usuarios
Atributos Propuestos
contactos: Contiene los contactos del usuario.
Métodos Propuestos
<p>buscarContactos: Obtiene los contactos del usuario.</p> <p>mostrarContacto: Obtiene los datos de un contacto seleccionado.</p> <p>bloquearContacto: Bloquea un contacto del usuario.</p> <p>buscarContactosBloqueados: Obtiene los contactos bloqueados del usuario.</p> <p>desbloquearContacto: Desbloquea un contacto del usuario.</p> <p>filtrarContactos: Filtra los contactos mostrados en el listado del usuario.</p> <p>aceptarSolicitud: Acepta una solicitud de contacto.</p> <p>ignorarSolicitud: Ignora una solicitud de contacto.</p>

Nombre de la Clase	
LoginController	
Descripción	
Controlador encargado del inicio de sesión en la aplicación.	
Responsabilidades	
Realizará una petición a la API de la aplicación para comprobar que las credenciales son correctas. Si es así, deberá almacenar una cookie de sesión en el navegador del cliente.	
Atributos Propuestos	
Métodos Propuestos	
login: Utilizado para iniciar sesión.	

Nombre de la Clase	
LogoutController	
Descripción	
Controlador encargado del cierre de sesión en la aplicación.	
Responsabilidades	
Elimina el token de sesión y redirecciona al usuario a la página de iniciar sesión y registro.	
Atributos Propuestos	
Métodos Propuestos	
logout: Utilizado para cerrar sesión.	

Nombre de la Clase
RegistroController
Descripción
Controlador encargado del registro en la aplicación
Responsabilidades
Mediante una petición a la API REST de la aplicación solicitará el registro de un nuevo usuario.
Atributos Propuestos
Métodos Propuestos
registrar: Utilizado para registrar un usuario.

Nombre de la Clase
langController
Descripción
Controlador encargado de modificar el idioma mostrado.
Responsabilidades
Atributos Propuestos
Métodos Propuestos
cambiarIdioma: Modifica el idioma de la aplicación.

Nombre de la Clase
ServiciosController
Descripción
Controlador encargado de la gestión de los servicios.
Responsabilidades
Este controlador permitirá gestionar los servicios de una sala permitiendo abrirlos, teniendo en cuenta la limitación de 4 servicios, y cerrarlos.
Atributos Propuestos
serviciosAbiertos: Listado en el que se especifican los servicios actualmente abiertos.
Métodos Propuestos
addServicio: Muestra un servicio en la interfaz. cerrarServicio: Elimina un servicio de la interfaz.

Nombre de la Clase
ChatTextoController
Descripción
Controlador encargado de la gestión del servicio de chat de texto.
Responsabilidades

Este controlador es el encargado de llevar acabo las funciones del servicio de chat de texto:
<ul style="list-style-type: none"> - Enviar mensajes y ficheros - Obtener mensajes recibidos - Obtener el número de usuarios que recibieron el mensaje
Atributos Propuestos
mensajes: listado de mensajes recibidos.
Métodos Propuestos
getMensajes: Obtiene mensajes de otros usuarios. enviarMensaje: Envía un mensaje a otros usuarios. enviarFichero: Envía un fichero a otros usuarios. getFeedback: Obtiene el número de usuarios que recibieron el mensaje.

Nombre de la Clase
DibujosController
Descripción
Controlador encargado de la gestión del servicio de dibujos.
Responsabilidades
Este controlador es el encargado de llevar acabo las funciones del servicio de dibujos:
<ul style="list-style-type: none"> - Añadir figuras - Borrar - Cambiar a modo dibujo o selección - Comunicar cambios a otros usuarios
Atributos Propuestos
canvas: referencia al canvas del servicio.
Métodos Propuestos
addFigura: Añade una figura al dibujo y se lo comunica a los demás usuarios. seleccionar: Activa el modo de selección. dibujar: Activa el modo de dibujo. borrar: Borra el dibujo y se lo comunica a los demás usuarios.

Nombre de la Clase
VideoCompartidoController
Descripción
Controlador encargado de la gestión del servicio de video compartido.
Responsabilidades
Este controlador es el encargado de llevar acabo las funciones del servicio de video compartido:
<ul style="list-style-type: none"> - Gestionar la reproducción del video - Cambiar la URL del video compartido

- Comunicar los cambios a otros usuarios
Atributos Propuestos
urlVideo: URL del video en reproducción.
Métodos Propuestos
cambiarVideo: Modifica la URL del video y se lo comunica a los demás usuarios. reproducirVideo: Reproduce o pausa la reproducción del video. setVolumen: Controla el volumen del video.

Nombre de la Clase
RadioController
Descripción
Controlador encargado de la gestión del servicio de radio
Responsabilidades
Este controlador es el encargado de llevar acabo las funciones del servicio de radio: <ul style="list-style-type: none"> - Cambiar la URL de la radio o canción - Comunicar los cambios a otros usuarios
Atributos Propuestos
urlRadio: Almacena la URL de la radio o canción que se reproduce actualmente.
Métodos Propuestos
cambiarEmisora: Cambia la URL de la radio o canción que se está reproduciendo y lo comunica a los otros usuarios.

Nombre de la Clase
PresentacionesController
Descripción
Controlador encargado de la gestión del servicio de presentaciones.
Responsabilidades
Este controlador es el encargado de llevar acabo las funciones del servicio de presentaciones: <ul style="list-style-type: none"> - Cambiar la presentación mostrada - Comunicar los cambios a otros usuarios
Atributos Propuestos
presentación: Almacena los datos de la presentación.
Métodos Propuestos
cambiarPresentacion: Cambia la presentación mostrada y lo comunica a los otros usuarios.

Nombre de la Clase
VideollamadasController
Descripción
Controlador encargado de la gestión del servicio de videollamadas
Responsabilidades
Este controlador es el encargado de llevar a cabo las funciones del servicio de videollamadas: <ul style="list-style-type: none"> - Iniciar la transmisión del video local a otros usuarios conectados - Recibir video remoto - Actualizar los videos en función de los usuarios que se conectan y desconectan - Activar o desactivar el altavoz - Activar o desactivar el micrófono - Activar o desactivar la transmisión del video local
Atributos Propuestos
conexiones: listado de conexiones remotas.
Métodos Propuestos
iniciarVideollamada: Comienza una videollamada. actualizarVideo: Actualiza los videos remotos en el caso de que alguien se conecte o salga de la videollamada. setAltavoz: Activa o desactiva el sonido de los usuarios remotos. setMicrofono: Activa o desactiva el micrófono. setVideo: Activa o desactiva la transmisión del video local.

Nombre de la Clase
Utils
Descripción
Utilizado para reunir todas aquellas funciones compartidas por varias clases.
Responsabilidades
Entre sus responsabilidades encontraremos: <ul style="list-style-type: none"> - Proporcionar métodos para mostrar notificaciones de información, error y éxito.
Atributos Propuestos
Métodos Propuestos
mensajeSuccess: Muestra una notificación de éxito con tiempo mensajeInfo: Muestra una notificación de información con tiempo mensajeError: Muestra una notificación de error con tiempo mensajeSuccessSinTiempo: Muestra una notificación de éxito sin tiempo mensajeInfoSinTiempo: Muestra una notificación de información sin tiempo mensajeErrorSinTiempo: Muestra una notificación de error sin tiempo

5.4.2.2 Capa de persistencia

Nombre de la Clase
ContactoController
Descripción
Controlador encargado de procesar peticiones recibidas referentes a la gestión de contactos.
Responsabilidades
Las responsabilidades de este controlador son las de buscar y persistir información relativa a los contactos en la base de datos. Esto incluye: <ul style="list-style-type: none"> - Buscar los contactos del usuario - Buscar posibles nuevos contactos - Bloquear contactos - Desbloquear contactos - Buscar contactos bloqueados - Buscar solicitudes de contacto - Enviar solicitudes de contacto - Aceptar solicitudes de contacto - Ignorar solicitudes de contacto
Atributos Propuestos
Métodos Propuestos
<p>buscarContactos: realiza una consulta en la que obtiene todos los contactos de un usuario.</p> <p>buscarPosibleContactos: realiza una consulta en la que obtiene para un usuario todos aquellos usuarios que pueden ser sus nuevos contactos. Estos usuarios son aquellos que no lo tiene bloqueado, que no forman parte de sus contactos, entre los que no se tienen solicitudes de contacto pendiente.</p> <p>bloquearContacto: realiza una consulta para crea una relación de “Bloqueado” entre dos nodos “Usuario”. También elimina las relaciones del usuario bloqueado con todas aquellas salas en las que el usuario que lo bloqueó es administrador.</p> <p>desbloquearContacto: realiza una consulta que elimina una relación de “Bloqueado” entre dos usuarios, y crea una relación de “Contacto” entre ellos.</p> <p>buscarBloqueados: realiza una consulta que devuelve todos los usuarios a los que se dirige una relación “Bloqueado” desde un usuario concreto.</p> <p>buscarSolicitudesContacto: realiza una consulta que devuelve todas las relaciones de “SolicitudContacto” entre un usuario concreto y todos los demás.</p> <p>enviarSolicitudContacto: realiza una consulta que crea una relación “SolicitudContacto” entre dos usuarios concretos.</p> <p>aceptarSolicitudContacto: realiza una consulta que elimina una relación “SolicitudContacto” y crea una relación “Contacto” entre dos usuarios concretos.</p> <p>ignorarSolicitudContacto: realiza una consulta que elimina una relación “SolicitudContacto” entre dos usuarios concretos.</p>

Nombre de la Clase
UsuarioController
Descripción
Controlador encargado de procesar peticiones recibidas referentes a las gestiones de un usuario.
Responsabilidades
Las responsabilidades de este controlador son las de buscar y persistir información relativa a un usuario en la base de datos. Esto incluye: <ul style="list-style-type: none"> - Comprobar las credenciales de un usuario - Registrar a un usuario - Validar un nombre de usuario - Buscar información del usuario - Modificar datos del usuario - Modificar contraseña del usuario - Eliminar cuenta de un usuario
Atributos Propuestos
Métodos Propuestos
<p>login: realiza una consulta para verificar que las credenciales enviadas son correctas. Para ello hará uso de la clase Passport.</p> <p>validarUsername: realiza una consulta para comprobar que un nombre de usuario no está en uso.</p> <p>registrar: realiza una consulta para crear un nuevo nodo “Usuario”.</p> <p>perfil: realiza una consulta para obtener los datos del usuario que realiza la petición.</p> <p>datosUsuario: realiza una consulta para obtener los datos de un usuario concreto.</p> <p>modificarPass: realiza una consulta para modificar la contraseña del usuario que realiza la petición.</p> <p>modificarDatos: realiza una consulta para modificar los datos personales del usuario que realiza la petición.</p> <p>eliminarCuenta: realiza una consulta para eliminar el nodo “Usuario” que representa al usuario que realiza la consulta. Además, elimina todas sus relaciones de tipo “Administrador” con salas y todas las relaciones de nodos “Usuario” a esas salas.</p>

Nombre de la Clase
SalaController
Descripción
Controlador encargado de procesar peticiones recibidas referentes a las gestiones de salas.
Responsabilidades
Las responsabilidades de este controlador son las de buscar y persistir información relativa a las salas en la base de datos. Esto incluye: <ul style="list-style-type: none"> - Crear salas - Actualizar los datos de una sala - Eliminar salas

- Enviar solicitudes de unión a salas
- Aceptar solicitudes de unión a salas
- Ignorar solicitudes de unión a salas
- Cambiar permisos otorgados en solicitudes de unión a salas
- Cambiar permisos de participantes de la sala
- Eliminar usuarios de la sala
- Buscar las salas para las que un usuario es administrador, moderador, miembro o ha recibido una solicitud de unión a la sala
- Buscar todos los participantes de una sala
-

Atributos Propuestos

Métodos Propuestos

crearSala: realiza una consulta para crear un nodo “Sala” con los datos pasados.

actualizarDatos: realiza una consulta para actualizar los datos de un usuario.

eliminarSala: realiza una consulta para eliminar un nodo “Sala”.

enviarSolicitudSala: realiza una consulta para crear una relación “Candidato” entre un nodo “Sala” y un nodo “Usuario” con una propiedad “Permisos”.

aceptarSolicitud: realiza una consulta que obtiene la propiedad “Permisos” de una relación “Candidato” entre un nodo “Sala” y un nodo “Usuario”, elimina la relación “Candidato” y crea entre los nodos una relación cuyo tipo depende de la propiedad “Permisos” recuperada.

ignorarSolicitud: realiza una consulta que elimina una relación “Candidato” entre un nodo “Sala” y un nodo “Usuario”.

cambiarPermisosSolicitud: realiza una consulta que modifica la propiedad “Permisos” de una relación “Candidato”.

cambiarPermisos: realiza una consulta que elimina una relación entre un nodo “Usuario” y un nodo “Sala” para crear una nueva con los permisos especificados.

eliminarUsuario: elimina la relación de un nodo “Usuario” con un nodo “Sala”.

buscarSalasParticipa: realiza una consulta en la que se busca los nodos “Sala” a los que un usuario posee una relación del tipo “Miembro”, “Moderador” o “Administrador”.

buscarSalasCandidato: realiza una consulta en la que se busca los nodos “Sala” a los que el usuario que realiza la petición posee una relación del tipo “Candidato”.

buscarSalasAdmin: realiza una consulta en la que se busca los nodos “Sala” a los que el usuario que realiza la petición posee una relación del tipo “Admin”.

buscarSalasModerador: realiza una consulta en la que se busca los nodos “Sala” a los que el usuario que realiza la petición posee una relación del tipo “Moderador”.

buscarSalasMiembro: realiza una consulta en la que se busca los nodos “Sala” a los que el usuario que realiza la petición posee una relación del tipo “Miembro”.

buscarParticipantesSala: realiza una consulta en la que se busca los nodos “Usuario” que tienen una relación del tipo “Admin”, “Moderador” o “Miembro” con una sala concreta.

buscarCandidatos: realiza una consulta en la que se busca los nodos “Usuario” que tienen una relación del tipo “Candidato” con una sala concreta.

Nombre de la Clase
Middleware
Descripción
Proporciona unas funciones que se ejecutan antes de procesar una petición.
Responsabilidades
Entre sus responsabilidades se encontrarán: <ul style="list-style-type: none"> - Comprobar que la petición es realizada por un administrador o moderador - Comprobar que dos usuarios son contactos
Atributos Propuestos
Métodos Propuestos
checkAdminOrModerador: Comprueba si la petición fue realizada por un administrador o moderador. checkAdmin: Comprueba si la petición fue realizada por un administrador. checkSonContactos: Comprueba si dos usuarios son contactos.

Nombre de la Clase
Routes
Descripción
Realiza el enrutado de peticiones
Responsabilidades
Recibe peticiones HTTP y decide a que funciones del middleware y de los controladores la envía.
Atributos Propuestos
express: Proporciona las herramientas necesarias para realizar el enrutado.
Métodos Propuestos

Nombre de la Clase
DB
Descripción
Contiene la información para acceder a la base de datos.
Responsabilidades
Obtener la información para acceder a la base de datos a partir de variables de entorno y almacenarlas en un atributo.
Atributos Propuestos
config: Datos necesarios para acceder a la base de datos.
Métodos Propuestos

Nombre de la Clase
Utils
Descripción
Utilizado para reunir todas aquellas funciones compartidas por varias clases.
Responsabilidades
Una de sus responsabilidades será proporcionar un método que devuelva el nombre de usuario codificado en la cookie de sesión enviada en una petición. Probablemente surjan nuevas responsabilidades mientras se realiza el desarrollo de la aplicación.
Atributos Propuestos
Métodos Propuestos
getUsername: Devuelve el nombre de usuario codificado en la cookie de sesión enviada en una petición.

Nombre de la Clase
Passport
Descripción
Valida si las credenciales son correctas
Responsabilidades
Su responsabilidad es comprobar si las credenciales enviadas para iniciar sesión son correctas.
Atributos Propuestos
passport: módulo para realizar autenticación. localStrategy: módulo que proporciona una estrategia para la autenticación de usuarios utilizando nombre de usuario y contraseña.
Métodos Propuestos

5.4.2.3 Servidor Web

Nombre de la Clase
HTTP
Descripción
Servidor HTTP
Responsabilidades
Incluye el código para la configuración del servidor HTTP.
Atributos Propuestos
http: Módulo que permite la creación de un servidor HTTP. port: Puerto en el que se está ejecutando el servidor de HTTP. servidor: Servidor de HTTP creado.
Métodos Propuestos

Nombre de la Clase
Middleware
Descripción
Proporciona unas funciones que se ejecutan antes de procesar una petición.
Responsabilidades
Entre sus responsabilidades se encontrará comprobar que una petición contiene el token de sesión.
Atributos Propuestos
Métodos Propuestos
checkToken: Comprueba que existe el token de sesión y, en caso de que no exista, redirige al usuario a la página de inicio de sesión y registro. Este método será utilizado cuando un usuario intenta acceder a una página en la que se necesita haber iniciado sesión.
checkSesion: Comprueba que existe el token de sesión y, en caso de que exista, redirige al usuario a la página principal. Este método será utilizado cuando un usuario intenta acceder a la página de inicio de sesión y registro cuando ya había iniciado sesión previamente.

Nombre de la Clase
PagesController
Descripción
Controlador encargado de procesar peticiones recibidas en las que se pide acceso a cierta vista.
Responsabilidades
Responderá a peticiones con la vista a visualizar.
Atributos Propuestos
Métodos Propuestos
index: devuelve al cliente la vista de inicio de sesión y registro.
mainPage: devuelve al cliente la página principal.
chatroom: devuelve al cliente la vista de las salas
manageRooms: devuelve al cliente la vista de gestión de salas
profileSettings: devuelve al cliente la vista de ajustes de la cuenta
about: devuelve al cliente la vista de información de contacto
legal: devuelve al cliente la vista de información legal

Nombre de la Clase
Routes
Descripción
Realiza el enrutado de peticiones
Responsabilidades
Recibe peticiones HTTP y decide a que funciones del middleware y de los controladores la envía.

Atributos Propuestos
<p>pagesController: Referencia a PagesController para llamar a sus métodos cuando se recibe una petición que tiene que procesar.</p> <p>middleware: Referencia a middleware para utilizar sus funciones.</p> <p>express: Proporciona las herramientas necesarias para realizar el enrutado.</p>
Métodos Propuestos

5.4.2.4 Servidor WebSockets

Nombre de la Clase
WebSockets
Descripción
Servidor de WebSockets
Responsabilidades
Incluye el código del servidor de WebSockets utilizado para las comunicaciones de los usuarios en las salas.
Atributos Propuestos
<p>WebSocket: Librería de WebSockets.</p> <p>port: Puerto en el que se está ejecutando el servidor de WebSockets.</p> <p>wss: Servidor de WebSockets.</p> <p>connections: Almacena las conexiones de los usuarios conectados a una sala.</p>
Métodos Propuestos
getUsuarios: Devuelve los usuarios conectados a una sala.

5.5 Análisis de Casos de Uso y Escenarios

En esta sección se detallarán los casos de uso a través de sus escenarios.

5.5.1 Caso de Uso: Registrarse

Registrarse	
Precondiciones	
Poscondiciones	Debe existir un nuevo usuario con un nombre de usuario único y el usuario será validado
Actores	Iniciado por usuario no registrado y terminado por usuario registrado
Descripción	<p>El usuario no registrado:</p> <ol style="list-style-type: none"> 1. Accederá a la pantalla de inicio de sesión y registro 2. Pulsará en el enlace “Registro” 3. Rellenará la información necesaria para realizar el registro 4. El sistema validará los datos introducidos 5. Accederá a la pantalla principal
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: El nombre de usuario ya existe en el sistema, algún dato introducido tiene formato erróneo o no se rellenaron campos obligatorios <ul style="list-style-type: none"> ○ Volver al paso 3 del escenario principal, manteniendo el resto de información en la pantalla
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No se puede registrarse <ul style="list-style-type: none"> ○ Notificar un error pidiendo que se intente de nuevo más tarde

5.5.2 Caso de Uso: Iniciar sesión

Iniciar sesión	
Precondiciones	Es necesario haberse registrado en el sistema
Poscondiciones	El usuario está validado
Actores	Iniciado por usuario no registrado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario accederá a la pantalla de inicio de sesión y registro 2. El usuario rellenará la información necesaria para realizar el inicio de sesión 3. El sistema validará la información introducida

	4. El usuario accederá a la pantalla principal
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: Campos obligatorios no introducidos <ul style="list-style-type: none"> ○ Notificar qué campo falta ○ Volver al paso 2 del escenario principal, manteniendo el resto de información en la pantalla • Escenario Alternativo 2: El nombre de usuario o contraseña es erróneo <ul style="list-style-type: none"> ○ Notificar que el usuario y/o la contraseña es erróneo ○ Volver al paso 2 del escenario principal, manteniendo el resto de información en la pantalla
Excepciones	<ul style="list-style-type: none"> • La base de datos no está disponible: No puede iniciar sesión <ul style="list-style-type: none"> ○ Notificar un error pidiendo que se intente de nuevo más tarde

5.5.3 Caso de Uso: Cambiar idioma

Cambiar idioma	
Poscondiciones	Se ha cambiado el idioma de los textos
Actores	Iniciado y terminado por usuario registrado o no registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario pulsará en uno de los botones con la bandera representativa de un idioma 5. EL sistema adaptará los textos de la interfaz al idioma seleccionado

5.5.4 Caso de Uso: Modificar datos cuenta

Modificar datos cuenta	
Precondiciones	Es necesario haberse validado en el sistema y dirigirse a la página de ajustes de la cuenta a través del menú lateral de la página principal
Poscondiciones	Los datos del usuario son modificados
Actores	Iniciado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario rellenará la información que desee modificar de su cuenta 2. El sistema validará y modificará los datos de su cuenta

Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: Campos obligatorios no introducidos <ul style="list-style-type: none"> ○ Notificar qué campo falta ○ Volver al paso 1 del escenario principal, manteniendo el resto de información en la pantalla • Escenario Alternativo 2: Se modifica la contraseña, pero la contraseña actual proporcionada no es correcta <ul style="list-style-type: none"> ○ Notificar que no fue autorizada la modificación de la contraseña ○ Volver al paso 1 del escenario principal, manteniendo la información en la pantalla • Escenario Alternativo 3: Se introduce algún dato con formato incorrecto <ul style="list-style-type: none"> ○ Notificar cuál es el problema
---	---

5.5.5 Caso de Uso: Crear sala

Crear sala	
Precondiciones	Es necesario haberse validado en el sistema
Poscondiciones	Existe una sala más en el sistema
Actores	Iniciado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario se dirigirá a la opción para crear salas en el menú lateral de la página principal 2. El sistema mostrará un cuadro de diálogo para crear la sala 3. El usuario rellenará la información necesaria para crear la sala 4. El sistema validará la información introducida 5. El sistema cerrará el cuadro de diálogo 6. El sistema mostrará la sala en el listado de salas
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: Campos obligatorios no introducidos <ul style="list-style-type: none"> ○ Notificar que campo falta ○ Volver al paso 3 del escenario principal, manteniendo el resto de información en la pantalla • Escenario Alternativo 2: Los datos introducidos tienen un formato erróneo <ul style="list-style-type: none"> ○ Notificar cuál es el problema

5.5.6 Caso de Uso: Añadir contactos

Añadir contactos	
Precondiciones	El usuario está creando una sala
Poscondiciones	Los contactos aparecen en un listado de la creación de la sala
Actores	Iniciado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario añadirá el nombre de usuario de un contacto en un campo de texto 2. El sistema mostrará un listado de los contactos en función del texto introducido 3. El usuario seleccionará un usuario del listado 4. El sistema validará que puede añadir el usuario a la sala 5. El sistema mostrará el contacto añadido en un listado
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: Se intenta añadir un usuario sin seleccionarlo en el listado <ul style="list-style-type: none"> ○ Notificar el problema ○ Volver al paso 1 del escenario principal • Escenario Alternativo 2: Se intentará añadir más de 3 contactos <ul style="list-style-type: none"> ○ Notificar que no puede haber más de 4 participantes en la sala junto a la persona que la crea

5.5.7 Caso de Uso: Enviar solicitud contacto

Enviar solicitud contacto	
Precondiciones	Es necesario haberse validado en el sistema
Poscondiciones	Se envía una solicitud de contacto al usuario seleccionado
Actores	Iniciado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario se dirigirá a la opción para añadir contactos en el menú lateral de la página principal 2. El sistema mostrará un cuadro de diálogo para enviar solicitudes de contacto 3. El usuario rellenará la información necesaria para enviar la solicitud de contacto 4. El sistema validará la información introducida 5. El sistema notificará de que se envió una solicitud de contacto 6. El sistema cerrará el cuadro de diálogo
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: Nombre de usuario no introducido

	<ul style="list-style-type: none"> ○ No se permitirá continuar con la solicitud ○ Volver al paso 2 del escenario principal, manteniendo el resto de información en la pantalla
--	--

5.5.8 Caso de Uso: Aceptar solicitud contacto

Aceptar solicitud contacto	
Precondiciones	Es necesario haberse validado en el sistema
Poscondiciones	El usuario tendrá un nuevo contacto y una solicitud de contacto menos
Actores	Iniciado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario se dirigirá a la opción de solicitudes de contacto en el menú lateral de la página principal 2. El sistema abrirá un cuadro de diálogo donde mostrará las solicitudes de contacto recibidas y pendientes 3. El usuario pulsará en el botón “Aceptar” la solicitud que se quiera aceptar 4. El sistema eliminará la solicitud del listado 5. El sistema notificará de que se aceptó la solicitud de contacto

5.5.9 Caso de Uso: Ignorar solicitud contacto

Ignorar solicitud contacto	
Precondiciones	Es necesario haberse validado en el sistema
Poscondiciones	El usuario tendrá una solicitud de contacto menos
Actores	Iniciado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario se dirigirá a la opción de solicitudes de contacto en el menú lateral de la página principal 2. El sistema abrirá un cuadro de diálogo donde mostrará las solicitudes de contacto recibidas y pendientes 3. El usuario pulsará en el botón “Ignorar” de la solicitud que se quiera ignorar 4. El sistema eliminará la solicitud del listado

5.5.10 Caso de Uso: Aceptar solicitud de unión a sala

Aceptar solicitud de unión a sala	
Precondiciones	Es necesario haberse validado en el sistema
Poscondiciones	El usuario tendrá acceso a una sala más y una solicitud de unión a sala menos

Actores	Iniciado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario se dirigirá a la opción de solicitudes de salas en el menú lateral de la página principal 2. El sistema abrirá un cuadro de diálogo donde mostrará las solicitudes de unión a salas recibidas y pendientes 3. El usuario pulsará en el botón “Aceptar” de la solicitud que se quiera aceptar 4. El sistema validará que se pueda aceptar la solicitud 5. El sistema eliminará la solicitud del listado 6. El sistema notificará de que se aceptó la solicitud
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: El usuario acepta la solicitud después de que esta fuese cancelada por otro usuario <ul style="list-style-type: none"> ○ Notificar que no se ha aceptado la solicitud ○ La solicitud se elimina del listado

5.5.11 Caso de Uso: Ignorar solicitud de unión a sala

Ignorar solicitud de unión a sala	
Precondiciones	Es necesario haberse validado en el sistema
Poscondiciones	El usuario tendrá una solicitud de unión a sala menos
Actores	Iniciado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario se dirigirá a la opción de solicitudes de salas en el menú lateral de la página principal 2. El sistema abrirá un cuadro de diálogo donde mostrará las solicitudes de unión a salas recibidas y pendientes 3. El usuario pulsará en el botón “Ignorar” de la solicitud que se quiera ignorar 4. El sistema eliminará la solicitud del listado

5.5.12 Caso de Uso: Acceder a sala

Acceder a sala	
Precondiciones	Es necesario haberse validado en el sistema
Poscondiciones	El usuario accede a la pantalla de las salas
Actores	Iniciado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario pulsará el botón para acceder a una determinada sala 2. El sistema comprobará que el usuario todavía tiene acceso al sistema 3. El usuario accederá a la pantalla de las salas

Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: El usuario es eliminado de la sala antes de que intente acceder <ul style="list-style-type: none"> ○ Notificar que el acceso no está autorizado • Escenario Alternativo 2: La sala es eliminada antes de que intente acceder <ul style="list-style-type: none"> ○ Notificar que el acceso no está autorizado
---	--

5.5.13 Caso de Uso: Filtrar salas

Filtrar salas	
Precondiciones	Es necesario haberse validado en el sistema
Poscondiciones	Se muestran las salas filtradas
Actores	Iniciado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario Introducirá un texto en el campo facilitado para filtrar salas 2. El sistema filtrará las salas mostrando únicamente aquellas cuyo nombre contenga el texto introducido

5.5.14 Caso de Uso: Filtrar contactos

Filtrar contactos	
Precondiciones	Es necesario haberse validado en el sistema
Poscondiciones	Se muestran los contactos filtrados
Actores	Iniciado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario introducirá un texto en el campo facilitado para filtrar contactos 2. El sistema filtrará los contactos mostrando únicamente aquellos cuyo nombre de usuario contenga el texto introducido

5.5.15 Caso de Uso: Bloquear contacto

Bloquear contacto	
Precondiciones	Es necesario haberse validado en el sistema
Poscondiciones	El usuario tiene un contacto menos y un usuario bloqueado más, y se elimina al usuario bloqueado de todas las salas donde el usuario es administrador
Actores	Iniciado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario se dirigirá a la opción de bloquear contactos en el menú lateral de la página principal

	<ol style="list-style-type: none"> 2. El sistema abrirá un cuadro de diálogo donde podrá bloquear contactos 3. El usuario introducirá un nombre de usuario 4. El sistema mostrará un listado con los contactos del usuario 5. El usuario seleccionará un contacto a bloquear 6. El sistema eliminará al usuario bloqueado del listado de contacto, lo eliminará de las salas donde el usuario es administrador y mostrará el usuario bloqueado en un listado de contactos bloqueados
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: Un usuario bloquea a otro y, seguidamente, este último intenta bloquear al primero <ul style="list-style-type: none"> ○ Notificar al último usuario de que la operación no fue autorizada

5.5.16 Caso de Uso: Desbloquear contacto

Desbloquear contacto	
Precondiciones	Es necesario haberse validado en el sistema
Poscondiciones	El usuario tiene un contacto más
Actores	Iniciado y terminado por usuario registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario se dirigirá a la opción de bloquear contactos en el menú lateral de la página principal 2. El sistema abrirá un cuadro de diálogo donde se mostrará un listado con los contactos bloqueados 3. El usuario pulsará en el botón correspondiente para desbloquear a un determinado contacto 4. El sistema mostrará un listado con los contactos del usuario 5. El usuario seleccionará un contacto a bloquear 6. El sistema eliminará al usuario bloqueado del listado de contacto, lo eliminará de las salas donde el usuario es administrador y mostrará el usuario bloqueado en un listado de contactos bloqueados
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: Un usuario bloquea a otro y, seguidamente, este último intenta bloquear al primero <ul style="list-style-type: none"> ○ Notificar al último usuario de que la operación no fue autorizada

5.5.17 Caso de Uso: Eliminar cuenta

Eliminar cuenta	
Precondiciones	Es necesario haberse validado en el sistema y dirigirse a la página de ajustes de la cuenta a través del menú lateral de la página principal
Poscondiciones	El usuario sale de la sesión, existe una cuenta menos en el sistema y todas las salas donde el usuario era administrador se eliminaron
Actores	Iniciado por usuario registrado y terminado por usuario no registrado
Descripción	<ol style="list-style-type: none"> 1. El usuario pulsará el botón para eliminar la cuenta 2. El sistema mostrará una ventana emergente para confirmar la acción 3. El usuario confirmará la acción a través de la ventana emergente 4. El sistema elimina el token de sesión del usuario y lo envía a la página para iniciar sesión y registrarse

5.5.18 Caso de Uso: Abrir servicio

Abrir servicio	
Precondiciones	Es necesario haberse validado en el sistema y acceder a una sala
Poscondiciones	El servicio se muestra en la interfaz y puede ser utilizado
Actores	Iniciado y terminado por usuario registrado que sea miembro, moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario abrirá el listado de servicios disponibles 2. El usuario seleccionará un servicio 3. El sistema validará que se puede abrir el servicio 4. El sistema muestra el servicio en la pantalla
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: El servicio ya estaba abierto <ul style="list-style-type: none"> ○ Notificar de que el servicio ya está abierto ○ No abrir otro servicio • Escenario Alternativo 2: Si es necesario avisar de la acción a otros usuarios conectados <ul style="list-style-type: none"> ○ Notificar de la acción a las instancias del sistema de otros usuarios

5.5.19 Caso de Uso: Cerrar servicio

Cerrar servicio	
Precondiciones	Es necesario haber abierto un servicio en una sala
Poscondiciones	El servicio se cierra
Actores	Iniciado y terminado por usuario registrado que sea miembro, moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario pulsa el botón de cerrar servicio 2. El sistema elimina el servicio de la pantalla
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: El servicio ya estaba abierto <ul style="list-style-type: none"> ○ Notificar de que el servicio ya está abierto ○ No abrir otro servicio • Escenario Alternativo 2: Ya hay 4 servicios abiertos <ul style="list-style-type: none"> ○ Notificar al usuario que no se pueden abrir más de cuatro servicios ○ No abrir otro servicio • Escenario Alternativo 3: Si es necesario avisar de la acción a otros usuarios conectados <ul style="list-style-type: none"> ○ Notificar de la acción a las instancias del sistema de otros usuarios

5.5.20 Caso de Uso: Mover servicio

Mover servicio	
Precondiciones	Es necesario haber abierto un servicio en una sala
Poscondiciones	La interfaz del servicio se localiza en otra posición
Actores	Iniciado y terminado por usuario registrado que sea miembro, moderador o administrador de la sala
Descripción	<p>El usuario registrado:</p> <ol style="list-style-type: none"> 1. Mantiene pulsado en una zona facilitada o, en su defecto, en cualquier zona del servicio para moverlo 2. Mueve el puntero por la interfaz 3. Suelta el servicio donde desee

5.5.21 Caso de Uso: Redimensionar servicio

Redimensionar servicio	
Precondiciones	Es necesario haber abierto un servicio en una sala
Poscondiciones	El tamaño del servicio y sus elementos se ha modificado
Actores	Iniciado y terminado por usuario registrado que sea miembro, moderador o administrador de la sala

Descripción	<p>El usuario registrado:</p> <ol style="list-style-type: none"> 1. Mantiene pulsado en una zona facilitada del servicio para redimensionarlo 2. Mueve el puntero por la interfaz 3. Suelta el servicio con el tamaño que desee
--------------------	--

5.5.22 Caso de Uso: Crear dibujo colaborativo

Crear dibujo colaborativo	
Precondiciones	Es necesario haber abierto el servicio de dibujos en una sala
Poscondiciones	El dibujo es visible por los usuarios conectados al servicio
Actores	Iniciado y terminado por usuario registrado que sea miembro, moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario seleccionará la figura u opción que quiere utilizar 2. El sistema realizará la acción seleccionada y la comunicará a los usuarios que estén usando el servicio

5.5.23 Caso de Uso: Compartir URL video

Compartir URL video	
Precondiciones	Es necesario haber abierto el servicio de video compartido en una sala
Poscondiciones	El video es cargado en los servicios de los usuarios conectados a la sala
Actores	Iniciado y terminado por usuario registrado que sea miembro, moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario introduce la URL del video deseado en un campo de texto 2. El sistema comunica la URL a los usuarios conectados 3. El sistema carga el video en el reproductor de video del servicio

5.5.24 Caso de Uso: Compartir presentación

Compartir presentación	
Precondiciones	Es necesario haber abierto el servicio de presentaciones en una sala
Poscondiciones	La presentación es cargada en los servicios de los usuarios conectados a la sala

Actores	Iniciado y terminado por usuario registrado que sea miembro, moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario pulsa un botón para cargar la presentación 2. El sistema abre el cuadro de diálogo de selección de ficheros por defecto del navegador 3. El usuario selecciona una presentación HTML o PDF 4. El sistema muestra la presentación en el visor del servicio 5. El sistema envía la presentación a los usuarios conectados a la sala

5.5.25 Caso de Uso: Participar en videollamada

Participar en videollamada	
Precondiciones	Es necesario haber abierto el servicio de videollamada en una sala
Poscondiciones	Se reproduce el video local y el video y sonido de los usuarios conectados al servicio
Actores	Iniciado y terminado por usuario registrado que sea miembro, moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El navegador mostrará un mensaje en el que se piden permisos para acceder al micrófono y a la cámara. 2. El usuario acepta los permisos
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: El usuario no concede los servicios <ul style="list-style-type: none"> ○ No se ningún video ni sonido • Escenario Alternativo 2: El usuario pulsa en el botón de silenciar micrófono <ul style="list-style-type: none"> ○ Los otros usuarios conectados no recibirán el sonido local • Escenario Alternativo 3: El usuario pulsa en el botón de silenciar altavoces <ul style="list-style-type: none"> ○ No se escuchará el sonido enviado por los usuarios conectados • Escenario Alternativo 4: El usuario pulsa en el botón de ocultar video <ul style="list-style-type: none"> ○ Los otros usuarios conectados no recibirán el video local

5.5.26 Caso de Uso: Enviar mensaje de texto

Enviar mensaje de texto	
Precondiciones	Es necesario haber abierto el servicio de chat de texto en una sala
Poscondiciones	El mensaje fue enviado y se muestra en el chat de los usuarios conectados al servicio
Actores	Iniciado y terminado por usuario registrado que sea miembro, moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario introducirá su mensaje en un campo facilitado para tal efecto 2. El usuario pulsará la tecla Enter o un botón en el servicio para enviar el mensaje 3. El sistema enviará el mensaje a los usuarios conectados 4. El sistema añadirá el mensaje al chat y desplazará la visualización de mensajes hasta este último 5. El sistema añadirá, únicamente en el servicio del emisor del mensaje, un número al lado del mensaje que indica los usuarios que lo recibieron

5.5.27 Caso de Uso: Enviar fichero

Enviar fichero	
Precondiciones	Es necesario haber abierto el servicio de chat de texto en una sala
Poscondiciones	El fichero fue enviado y se puede descargar desde el chat de los usuarios conectados al servicio
Actores	Iniciado y terminado por usuario registrado que sea miembro, moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario pulsará un botón facilitado para añadir un fichero 2. El sistema abrirá el cuadro de diálogo por defecto para seleccionar el fichero a enviar 3. El sistema enviará el fichero a los usuarios conectados 4. El sistema añadirá una imagen para representar el fichero en el chat y desplazará la visualización de mensajes hasta esta última 5. El sistema notificará, únicamente en el servicio del emisor del mensaje, el número de usuarios que recibió el fichero

5.5.28 Caso de Uso: Descargar fichero

Descargar fichero	
Precondiciones	Es necesario haber abierto el servicio de chat de texto en una sala y haber recibido un fichero

Poscondiciones	El fichero es almacenado en el ordenador
Actores	Iniciado y terminado por usuario registrado que sea miembro, moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario pulsará sobre el fichero recibido 2. El sistema abrirá el cuadro de diálogo por defecto del navegador para almacenar el fichero

5.5.29 Caso de Uso: Abandonar sala

Abandonar sala	
Precondiciones	Es necesario haberse validado en el sistema y dirigirse a la página de gestión de salas a través del menú lateral de la página principal
Poscondiciones	El usuario tiene una sala menos en su listado y no puede acceder a la sala que abandonó
Actores	Iniciado y terminado por usuario registrado que sea miembro o moderador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario seleccionará la sala que desea abandonar de entre sus listados de salas donde participa como miembro o como moderador 2. El usuario pulsará un botón para abandonar la sala 3. El sistema mostrará la ventana emergente por defecto del navegador para confirmar la acción 4. El usuario confirma la acción en la ventana emergente 5. El sistema actualiza la página

5.5.30 Caso de Uso: Cambiar permisos solicitud

Cambiar permisos solicitud	
Precondiciones	Es necesario haberse validado en el sistema y dirigirse a la página de gestión de salas a través del menú lateral de la página principal
Poscondiciones	La solicitud enviada concede distintos permisos
Actores	Iniciado y terminado por usuario registrado que sea moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario seleccionará la sala en la cual quiere modificar los permisos de una solicitud de unión a sala 2. El usuario cambiará los permisos usando un comboBox 3. El sistema notifica que los permisos de la solicitud fueron cambiados
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: El usuario acepta la solicitud y se cambian los permisos de la misma poco después <ul style="list-style-type: none"> ○ Notificar que el cambio no fue autorizado

5.5.31 Caso de Uso: Eliminar solicitud

Eliminar solicitud	
Precondiciones	Es necesario haberse validado en el sistema y dirigirse a la página de gestión de salas a través del menú lateral de la página principal
Poscondiciones	La solicitud enviada está eliminada
Actores	Iniciado y terminado por usuario registrado que sea moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario seleccionará la sala en la cual quiere eliminar la solicitud de unión a sala 2. El usuario elimina la solicitud 3. El sistema actualiza el listado de solicitudes enviadas 4. El sistema notifica que la solicitud fue eliminada
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: La solicitud se elimina justo después de ser aceptada por el usuario <ul style="list-style-type: none"> ○ Notificar que la acción no fue autorizada

5.5.32 Caso de Uso: Cambiar permisos miembros

Cambiar permisos miembros	
Precondiciones	Es necesario haberse validado en el sistema y dirigirse a la página de gestión de salas a través del menú lateral de la página principal
Poscondiciones	El miembro seleccionado ahora es moderador en la sala
Actores	Iniciado y terminado por usuario registrado que sea moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario seleccionará la sala en la cual quiere modificar los permisos de un miembro 2. El usuario cambiará los permisos usando un comboBox 3. El sistema notifica que los permisos fueron cambiados
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: Los permisos se modifican poco después de que el miembro abandona la sala <ul style="list-style-type: none"> ○ Notificar que la operación no fue autorizada

5.5.33 Caso de Uso: Enviar solicitud de unión a sala

Enviar solicitud de unión a sala	
Precondiciones	Es necesario haberse validado en el sistema y dirigirse a la página de gestión de salas a través del menú lateral de la página principal
Poscondiciones	Se envió una solicitud de unión a la sala a un usuario

Actores	Iniciado y terminado por usuario registrado que sea moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario seleccionará la sala para la cual quiere enviar una solicitud de unión 2. El usuario introduce el nombre de usuario y lo selecciona de una lista 3. El sistema actualiza el listado de solicitudes enviadas 4. El sistema notifica que la solicitud fue enviada
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: La suma de las solicitudes enviadas y participantes de la sala son 4 personas <ul style="list-style-type: none"> ○ Notifica que no puede haber más de 4 personas en una sala • Escenario Alternativo 2: Ya se envió una solicitud de unión al usuario <ul style="list-style-type: none"> ○ Notifica que ya se envió una solicitud de unión • Escenario Alternativo 3: El usuario ya participa en la sala <ul style="list-style-type: none"> ○ Notifica que el usuario ya participa en la sala

5.5.34 Caso de Uso: Eliminar miembro

Eliminar miembro	
Precondiciones	Es necesario haberse validado en el sistema y dirigirse a la página de gestión de salas a través del menú lateral de la página principal
Poscondiciones	La sala tiene un miembro menos
Actores	Iniciado y terminado por usuario registrado que sea moderador o administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario seleccionará la sala en la cual quiere eliminar al miembro 2. El usuario elimina al miembro 3. El sistema actualiza la página 4. El sistema notifica que el usuario fue eliminado

5.5.35 Caso de Uso: Eliminar sala

Eliminar sala	
Precondiciones	Es necesario haberse validado en el sistema y dirigirse a la página de gestión de salas a través del menú lateral de la página principal
Poscondiciones	Todos los participantes fueron eliminados de la sala y existe una sala menos en el sistema
Actores	Iniciado y terminado por usuario registrado que sea administrador de la sala

Descripción	<ol style="list-style-type: none"> 1. El usuario seleccionará la sala que quiere eliminar 2. El usuario elimina la sala 3. El sistema actualiza la página
--------------------	--

5.5.36 Caso de Uso: Cambiar permisos moderadores

Cambiar permisos moderadores	
Precondiciones	Es necesario haberse validado en el sistema y dirigirse a la página de gestión de salas a través del menú lateral de la página principal
Poscondiciones	El moderador seleccionado ahora es miembro en la sala
Actores	Iniciado y terminado por usuario registrado que sea administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario seleccionará la sala en la cual quiere modificar los permisos de un moderador 2. El usuario cambiará los permisos usando un comboBox 3. El sistema notifica que los permisos fueron cambiados
Variaciones (escenarios secundarios)	<ul style="list-style-type: none"> • Escenario Alternativo 1: Los permisos se modifican poco después de que el moderador abandona la sala <ul style="list-style-type: none"> ○ Notificar que la operación no fue autorizada

5.5.37 Caso de Uso: Eliminar moderadores

Eliminar moderadores	
Precondiciones	Es necesario haberse validado en el sistema y dirigirse a la página de gestión de salas a través del menú lateral de la página principal
Poscondiciones	Existe un moderador menos en la sala
Actores	Iniciado y terminado por usuario registrado que sea administrador de la sala
Descripción	<ol style="list-style-type: none"> 1. El usuario seleccionará la sala en la cual quiere eliminar al moderador 2. El usuario elimina al moderador 3. El sistema actualiza la página 4. El sistema notifica que el usuario fue eliminado

5.6 Análisis de Interfaces de Usuario

Se ha diseñado una interfaz sencilla cuyo principal objetivo es que todas las vistas de la aplicación puedan ser accesibles con un solo clic desde la página principal.

He buscado que cada vista sea simple y que tenga una razón de ser, de modo que se dediquen a facilitar una función concreta. Además, uno de los requisitos no funcionales establecidos es mostrar notificaciones en la parte superior derecha de la pantalla, por lo que se utilizará esta vía para devolver feedback a los usuarios con algunas de sus acciones.

Por último, se busca que esta interfaz sea accesible por aquellas personas con daltonismo. Por ello, una vez que las interfaces sean construidas, se comprobará con la aplicación Sim Daltonism que las tonalidades y combinaciones elegidas son las correctas.

5.6.1 Descripción de la Interfaz

En esta sección se mostrarán las interfaces que se han diseñado para la aplicación. Se ha utilizado “Balsamiq Mockups” para la realización de todas las interfaces, exceptuando la interfaz para las salas. Esta última fue diseñada utilizando HTML, ya que se utilizó para explicar cómo funcionaría la interfaz “modular” a los directores de este proyecto.

5.6.1.1 Vistas principales

5.6.1.1.1 Interfaz de inicio de sesión

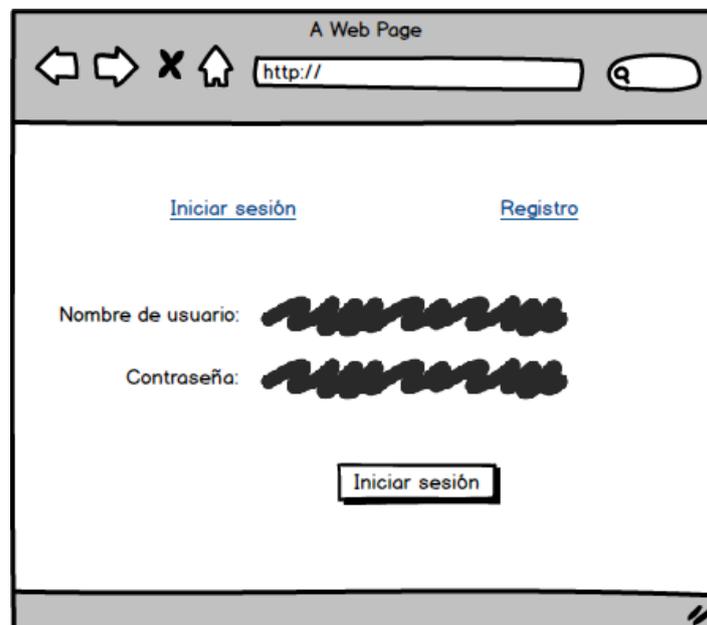


Figura 5.8. Interfaz inicio de sesión

Esta será la interfaz para iniciar sesión en la aplicación. Se proporcionará un nombre de usuario y contraseña y se pulsará el botón “Iniciar sesión”. Encima del formulario encontraremos dos links, uno de ellos nos servirá para movernos a la interfaz de iniciar sesión, que en este caso no funcionaría, y otro a la de registro. Con estos links utilizaremos una animación para cambiar de una interfaz a otra, sin que sea necesario la existencia de una segunda vista.

5.6.1.1.2 Interfaz para registrarse

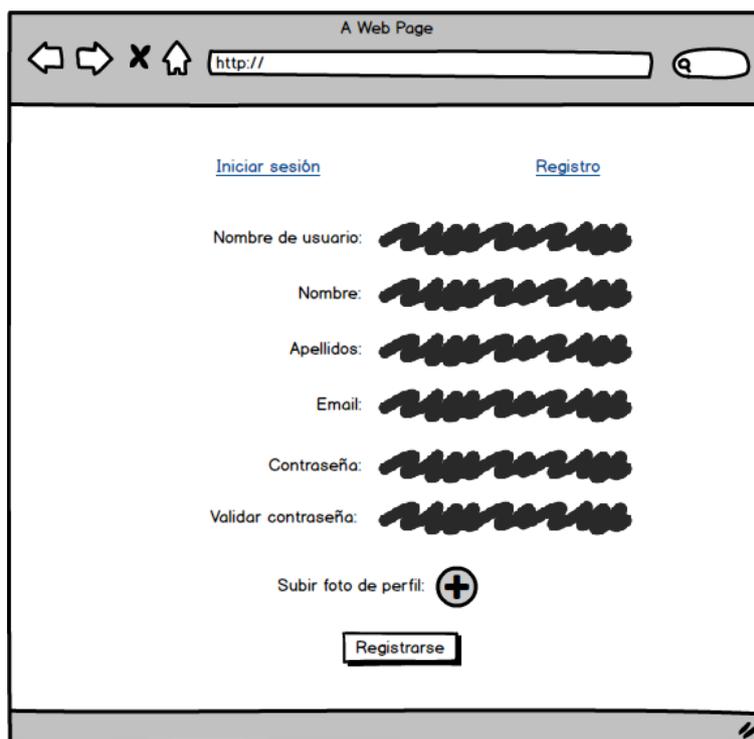


Figura 5.9. Interfaz para registrarse

Para realizar el registro se mostrará un formulario con los campos que se observan en la figura 5.9. Como se explicó en la interfaz de iniciar sesión, existirán dos links encima del formulario que permitirán, mediante una animación, desplazarse entre las interfaces. En este caso el link de “Registro” no funcionaría.

5.6.1.1.3 Interfaz página principal

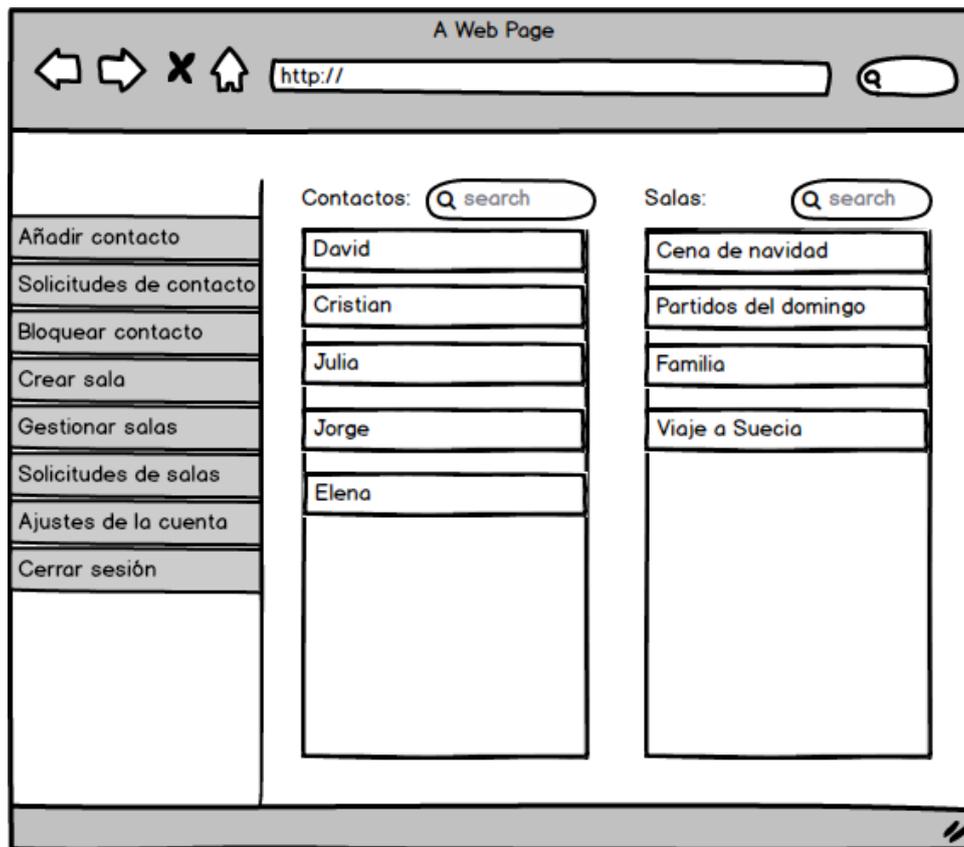


Figura 5.10. Interfaz página principal

La interfaz para la página principal se divide en tres partes. Un menú lateral para acceder a distintas páginas o cuadros de diálogo, un listado con los contactos y un listado con las salas.

Los listados de contactos permitirán pulsar en cada usuario para acceder a un cuadro de dialogo con información sobre este. En el caso de las salas, al pulsar sobre ellas accederemos a la interfaz de la sala. En los dos listados se proporcionará un campo de texto para filtrar los contactos y salas.

5.6.1.1.4 Interfaz para gestionar salas

Como ya se ha explicado anteriormente, un usuario registrado puede desarrollar el rol de miembro, moderador o administrador respecto a una sala. Por lo tanto, es necesario proporcionar diferentes herramientas para gestionar las salas dependiendo de ese rol.

A continuación, se mostrarán las interfaces para la gestión de salas. En la primera figura se muestra la interfaz que ven todos los usuarios al acceder a la gestión de salas. Mientras que las tres siguientes muestran un ejemplo de lo que verían los administradores, moderadores y miembros.

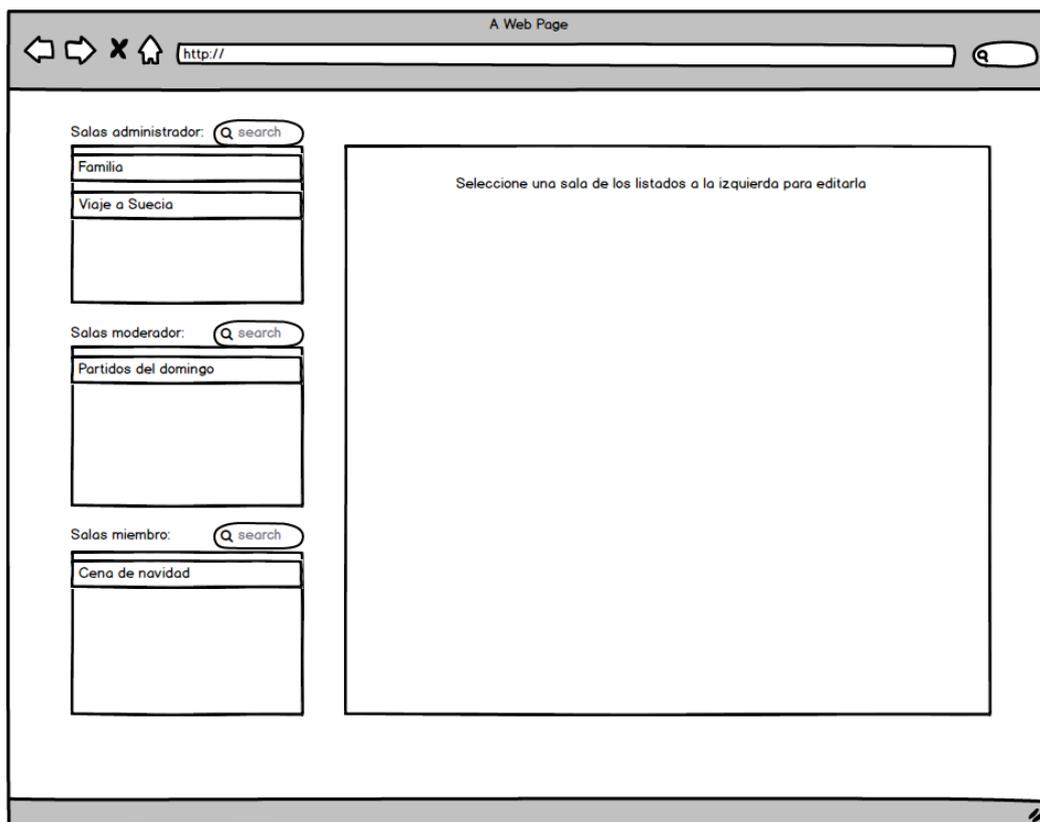


Figura 5.11. Interfaz para gestionar salas

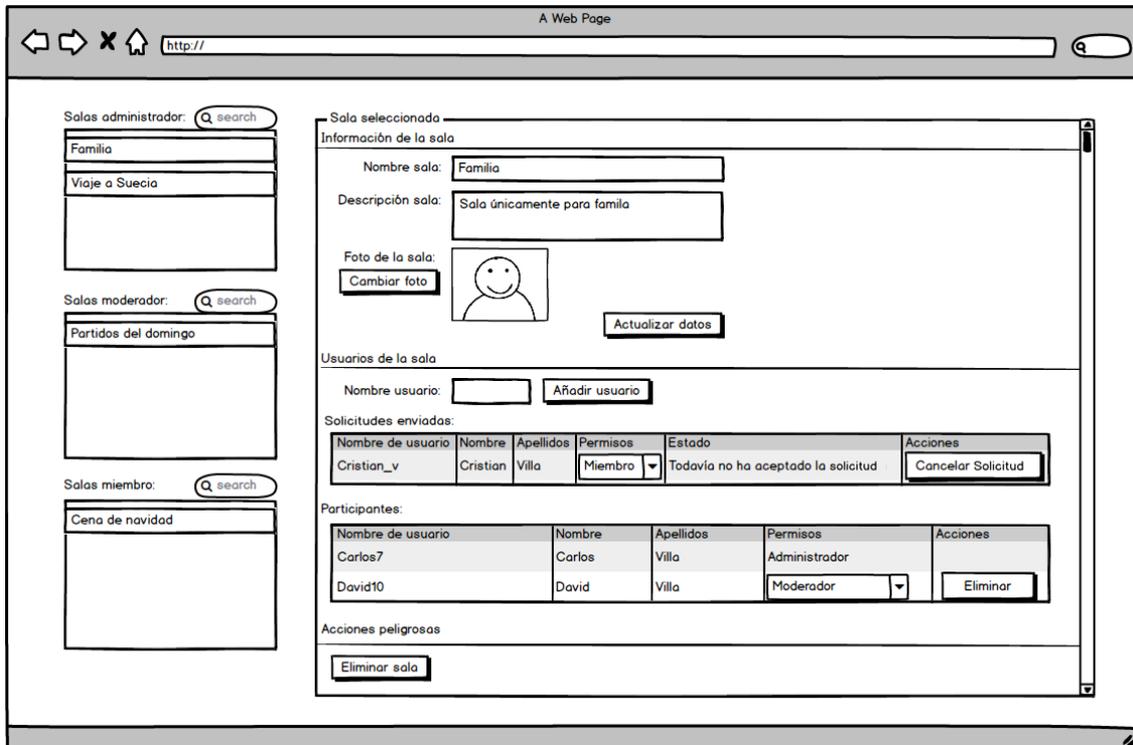


Figura 5.12. Interfaz para gestionar salas como administrador

En la figura 5.12 se muestra la información que puede editar un usuario con permisos de administrador. Como podemos ver, se ha seleccionado la sala “Familia” y se muestra en la parte derecha el estado actual de la sala:

- Primero encontramos un apartado “Información de la sala” donde se puede modificar el nombre, la descripción y la foto de la sala. Estos datos se persistirán al pulsar el botón “Actualizar datos”.
- En el segundo apartado se encuentra las herramientas para gestionar las solicitudes de unión enviadas y también los usuarios que tienen acceso a la sala (Participantes). Mediante el campo de texto, los combo boxes y los botones mostrados podremos:
 - o Añadir usuarios.
 - o Cambiar los permisos especificados en las solicitudes.
 - o Eliminar solicitudes enviadas.
 - o Cambiar los permisos de los usuarios que participan en la sala.
 - o Eliminar usuarios que participan en la sala.
- En el último apartado tendremos acceso a las “Acciones peligrosas”. En el caso del administrador, este dispondrá de un botón para eliminar la sala. Previamente a que se elimine la sala, aparecerá una ventana emergente para confirmar acción. Se usará la ventana con el diseño por defecto del navegador.

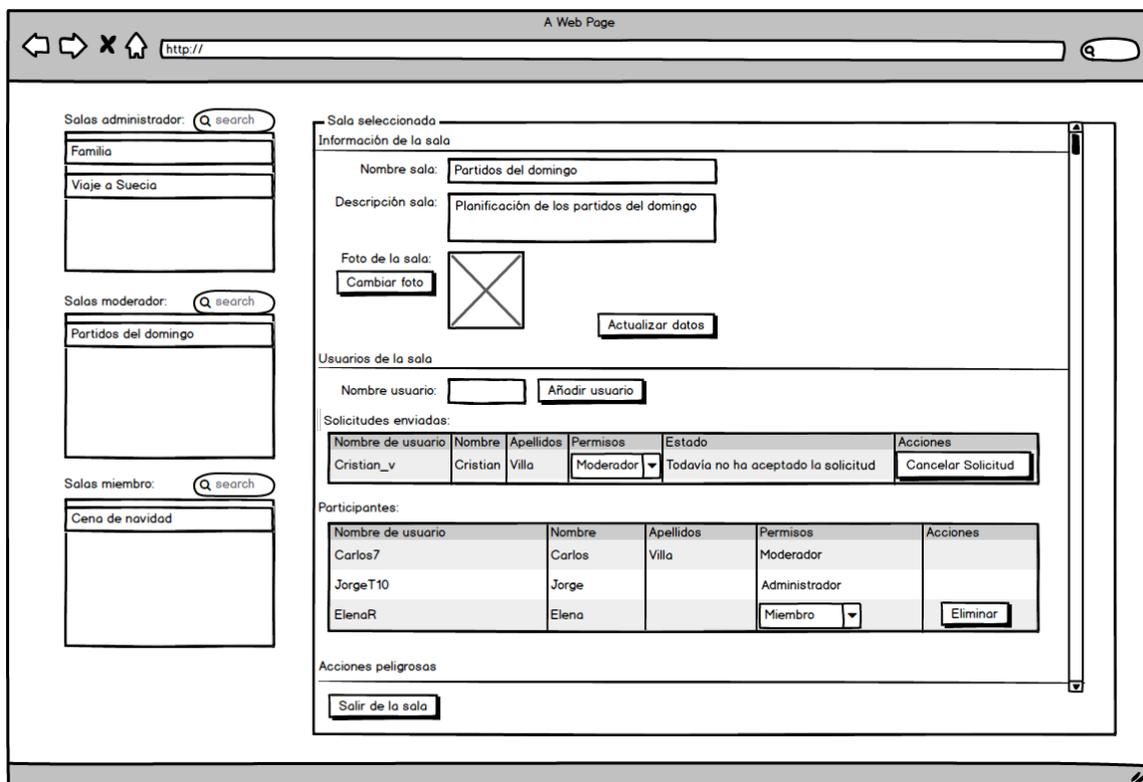


Figura 5.13. Interfaz para gestionar salas como moderador

En la figura 5.13 se muestra las herramientas disponibles para un moderador a la hora de gestionar una sala. La diferencia respecto al administrador es que este no puede editar los permisos de aquellos participantes que sean moderadores ni puede eliminarlos de la sala. Tampoco puede eliminar la sala, pero si puede salir de ella.

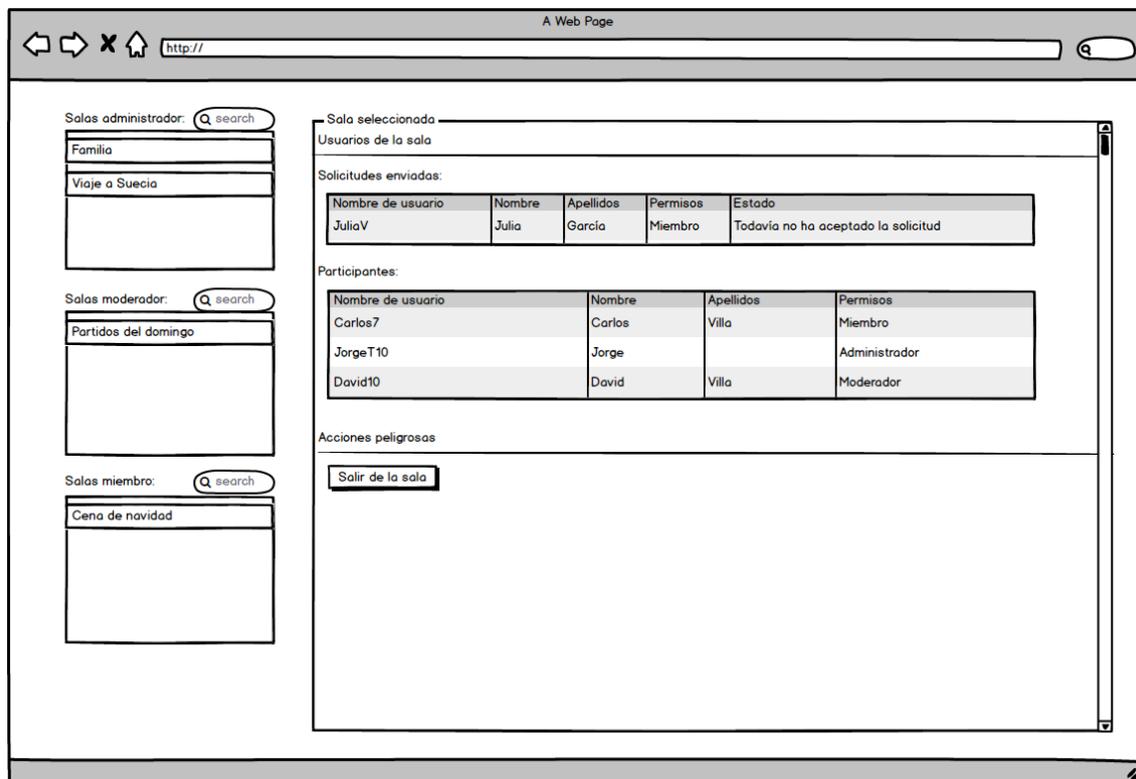


Figura 5.14. Interfaz para gestionar salas como miembro

Finalmente, en la figura 5.14 se muestran las herramientas disponibles por los miembros a la hora de gestionar una sala. Estos únicamente pueden ver las solicitudes enviadas y los participantes y salir de sala.

5.6.1.1.5 Interfaz salas

El objetivo de la interfaz para las salas es ofrecer una interfaz “modular” que permita al usuario seleccionar que servicios quiere ver en pantalla. Antes de comenzar el proyecto se desarrolló el siguiente prototipo utilizando HTML:

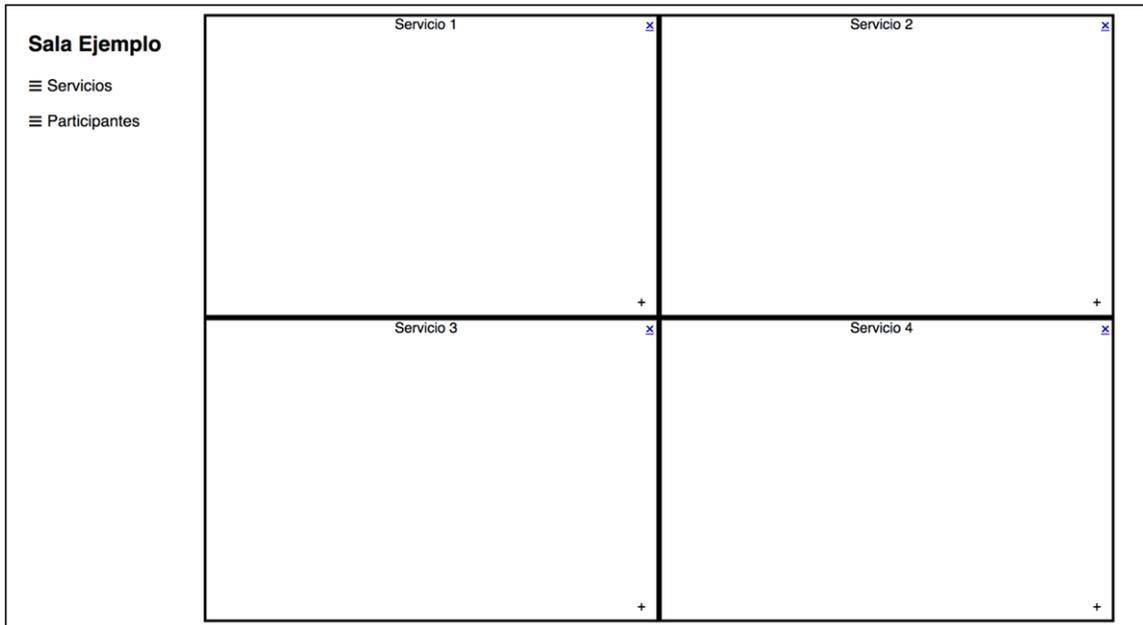


Figura 5.15. Interfaz salas

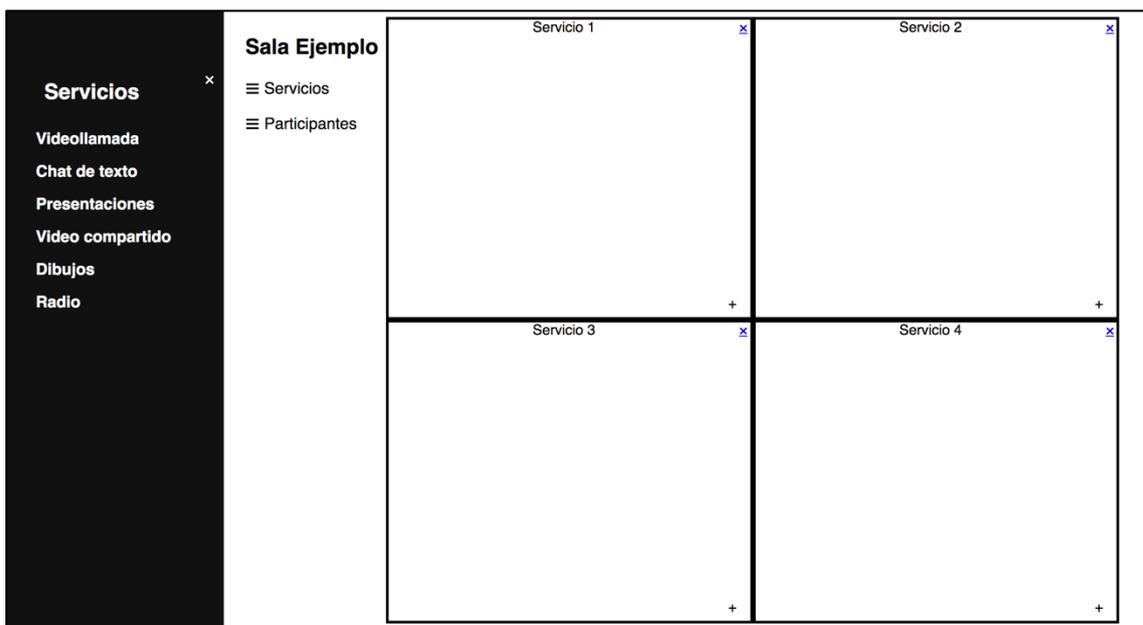


Figura 5.16. Interfaz salas mostrando listado de servicios

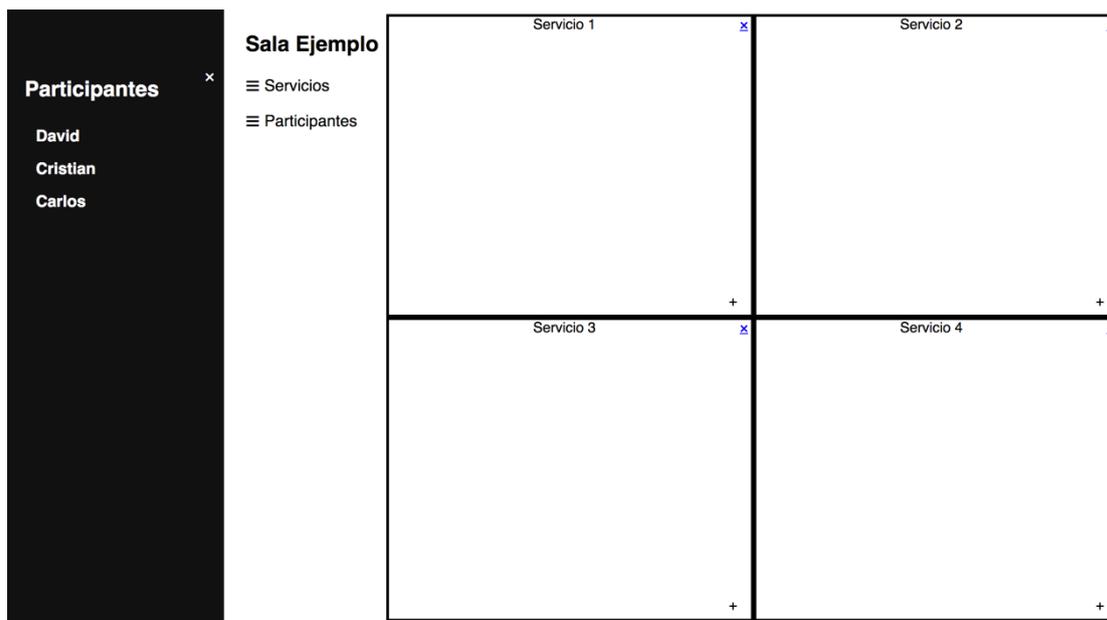


Figura 5.17. Interfaz salas mostrando listado de participantes conectados

Como podemos ver en las figuras 5.15, 5.16 y 5.17, la interfaz constaría de un menú lateral con el cual podemos acceder a los listados de los servicios y de los participantes actualmente conectados en la sala. La mayor parte de la interfaz se dedicaría a una zona en la que se pueden mover los servicios abiertos, cerrarlos y redimensionarlos.

Para abrir un servicio simplemente lo seleccionaríamos en la lista de servicio, en el caso que queramos cerrarlo pulsaríamos en la “X” localizada en la esquina superior derecha de cada servicio. Para redimensionar un servicio pulsaríamos en el símbolo “+” en la esquina inferior derecha y desplazaríamos el puntero.

5.6.1.1.6 Interfaz para ajustes de la cuenta

Figura 5.18. Interfaz para ajustes de la cuenta

En esta interfaz el usuario podrá cambiar información sobre su cuenta, esto es, su nombre, apellidos, email y foto de perfil. También podrá modificar su contraseña y eliminar la cuenta. Al pulsar en el botón “Eliminar cuenta” se mostrará una ventana emergente para confirmar la acción. Esta ventana usará el diseño por defecto del navegador.

5.6.1.1.7 Barras de navegación

En todas las interfaces se utilizará una barra de navegación la cual será la misma para todas ellas. La única diferencia es que dependiendo de la vista en la que nos encontremos se ocultará o mostrará diferente información. Como el idioma o el nombre de usuario.

Figura 5.19. Barra de navegación

5.6.1.2 Cuadros de diálogo

Algunas acciones no requieren un espacio muy amplio para ser realizadas. Por ello se decidió utilizar cuadros de diálogo cuando fuese posible. Esto permitirá simplificar el uso de la aplicación y no complicar la navegabilidad por el sistema.

5.6.1.2.1 Diálogo añadir contacto

El diálogo 'Añadir contacto' presenta un título en la parte superior. Debajo del título, hay un campo de texto etiquetado 'Nombre de usuario:'. A continuación, hay un campo de texto más grande etiquetado 'Mensaje'. En la parte inferior del diálogo, hay dos botones: 'Mandar solicitud de contacto' y 'Cerrar'.

Figura 5.20. Diálogo añadir contacto

Interfaz que permitirá enviar solicitudes de contacto especificando un nombre de usuario y un mensaje.

5.6.1.2.2 Diálogo solicitudes de contactos

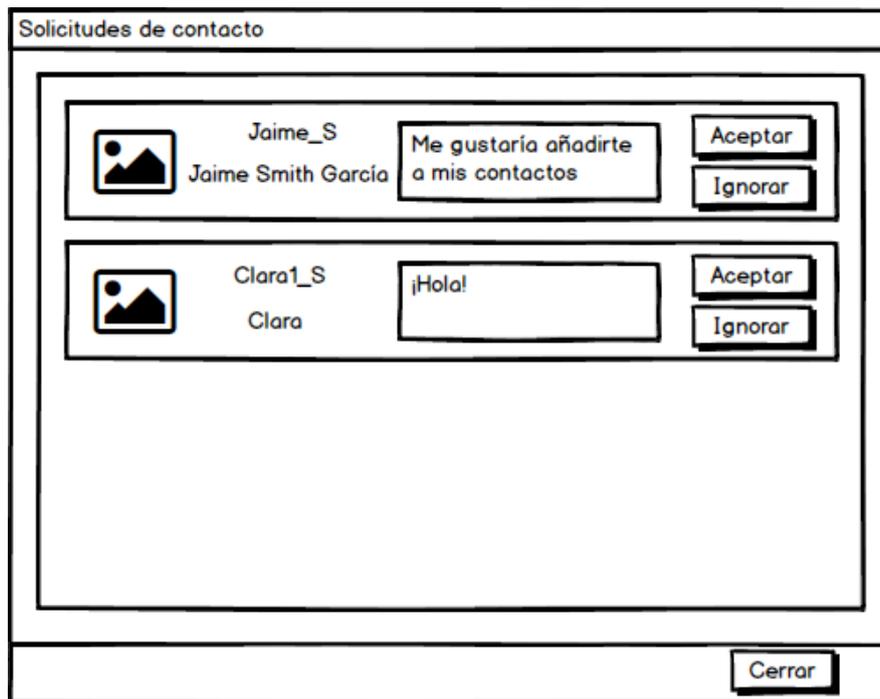


Figura 5.21. Diálogo solicitudes de contactos

Interfaz que permitirá ver las solicitudes de contactos y aceptarlas o ignorarlas.

5.6.1.2.3 Diálogo bloquear contacto

The dialog box is titled "Bloquear contacto". It contains a text input field labeled "Nombre de usuario:". Below this is a button labeled "Bloquear usuario". A table with the following data is displayed:

Nombre de usuario	Nombre	Apellidos	Desbloquear
Claire_2	Claire	Hamilton	Desbloquear
GeorgeH	George	Herrera	Desbloquear

At the bottom right of the dialog is a button labeled "Cerrar".

Figura 5.22. Diálogo bloquear y desbloquear contacto

Interfaz que permitirá bloquear contactos, ver los contactos bloqueados y desbloquearlos.

5.6.1.2.4 Diálogo crear sala

Crear sala

Datos sala

Nombre:

Descripción:

Participantes

Nombre de usuario:

Nombre de usuario	Nombre	Apellidos	Permisos	Eliminar
Cristian_v	Cristian	Villa	Moderador ▼	<input type="button" value="Eliminar"/>
JuliaV	Julia	García	Miembro ▼	<input type="button" value="Eliminar"/>

Foto de la sala

Seleccionar foto para la sala:

Figura 5.23. Diálogo crear sala

Interfaz que permitirá crear salas especificando un nombre, descripción, los participantes y sus permisos y una foto para la sala. De toda esta información solo el nombre será obligatorio.

5.6.1.2.5 Diálogo solicitudes de salas



Figura 5.24. Diálogo solicitudes de salas

Interfaz que permitirá ver las solicitudes de unión a salas recibidas y aceptarlas o ignorarlas.

5.6.1.3 Interfaces para los servicios

5.6.1.3.1 Servicios de videollamada

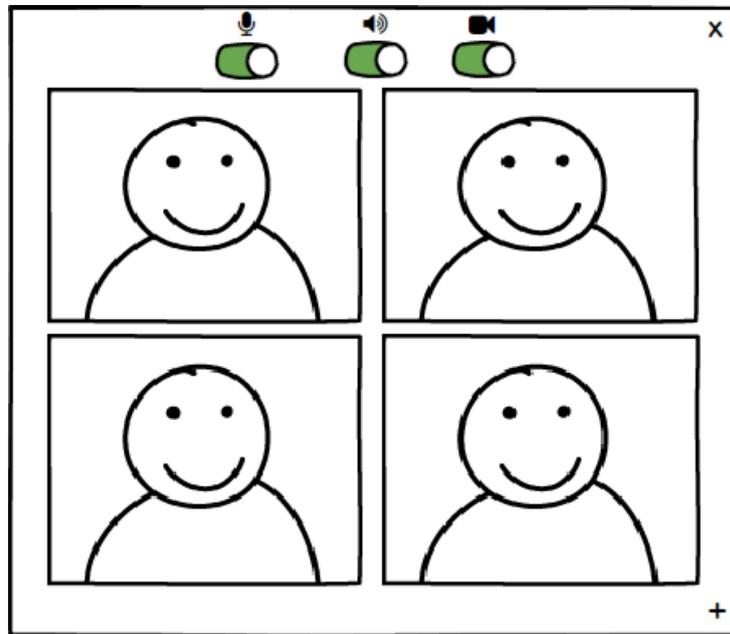


Figura 5.25. Interfaz servicio de videollamada

La interfaz del servicio de videollamada permitirá mostrar el video de hasta 4 usuarios al mismo tiempo incluyendo el del video local. Se facilitarán 3 botones para silenciar el micrófono, los altavoces y pausar el video local.

5.6.1.3.2 Servicio de chat de texto

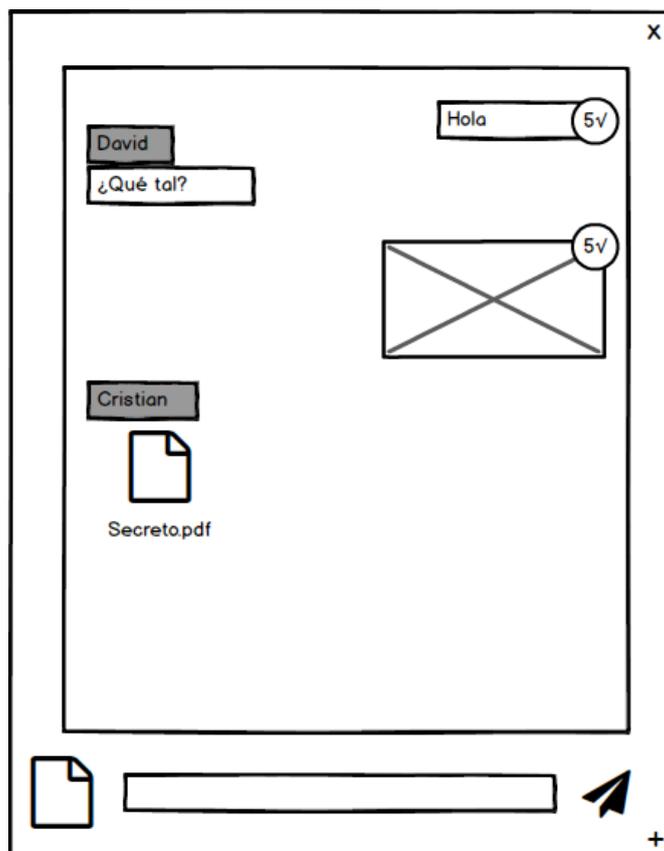


Figura 5.26. Interfaz servicio de chat de texto

La interfaz del servicio de chat de texto proporciona un área de texto para escribir nuestros mensajes y un botón para enviarlos. También existirá un botón para seleccionar ficheros que se enviarán al resto de usuario conectados en la sala. Al lado de los mensajes que enviemos se mostrará un número que indicará el número de usuarios que recibieron el mensaje.

5.6.1.3.3 Servicio de presentaciones

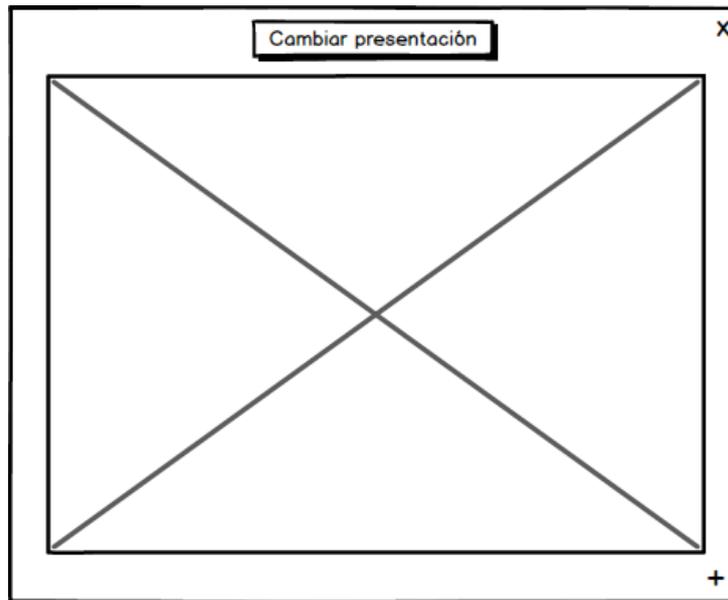


Figura 5.27. Interfaz servicio de presentaciones

La interfaz del servicio de presentaciones incluirá un botón para que seleccionemos nuestras presentaciones HTML o documentos PDF, y las mostrará en un frame.

5.6.1.3.4 Servicio de video compartido

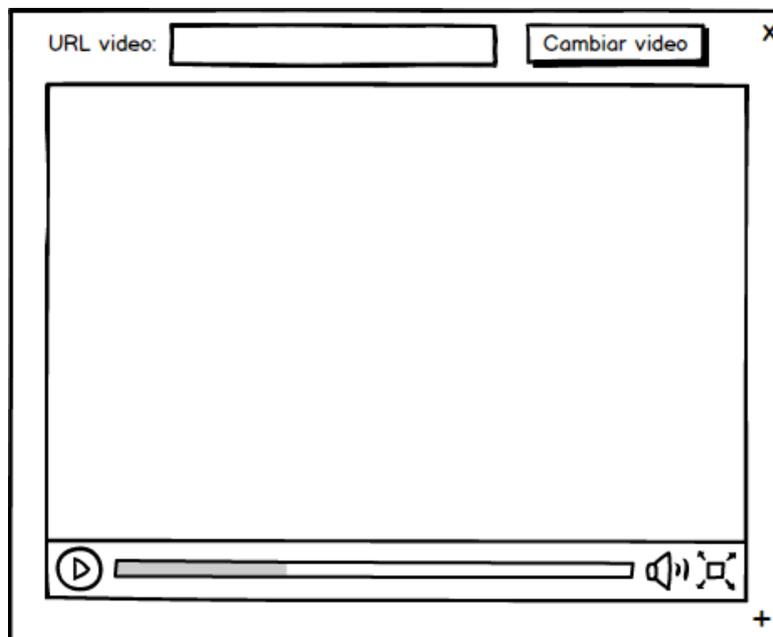


Figura 5.28. Interfaz servicio de video compartido

La interfaz del servicio de video compartido incluirá un campo de texto para que introduzcamos las URLs de los videos, y un botón para cambiar el video. También

incluirá un reproductor con controles de pausa o reanudación, sonido, adelantar o retrasar la reproducción y para mostrar el video en pantalla completa.

5.6.1.3.5 Servicio de dibujos

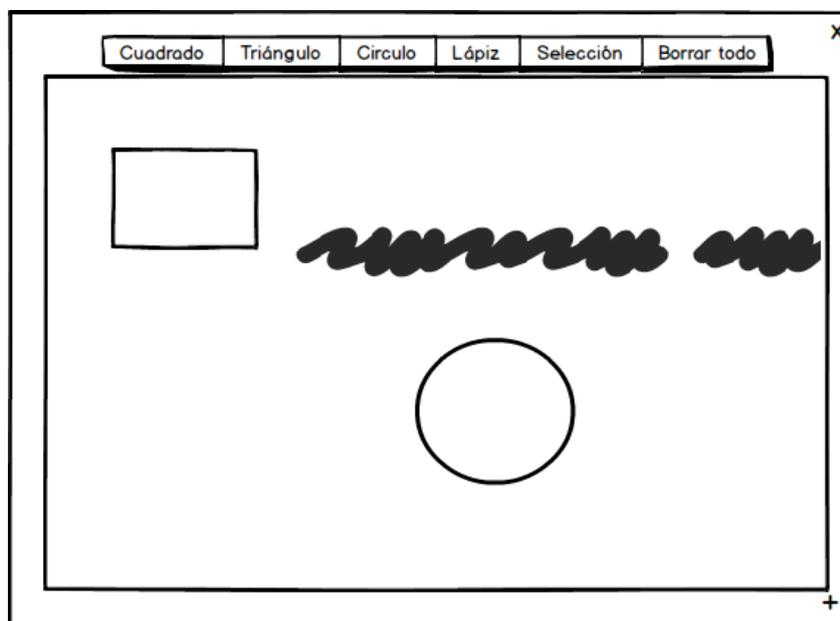


Figura 5.29. Interfaz servicio de dibujos

La interfaz del servicio de dibujos incluirá un canvas para añadir, mover y redimensionar los dibujos, además de una serie de botones para añadir figuras de cuadrados, triángulos y círculos, realizar dibujos a mano alzada, seleccionar los dibujos y borrar todo.

5.6.1.3.6 Servicio de radio

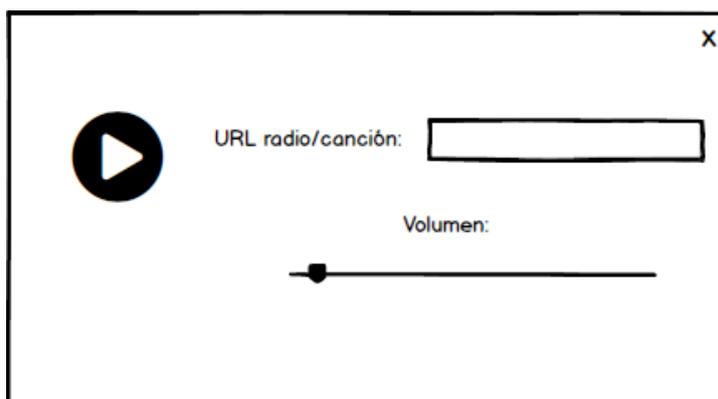


Figura 5.30. Interfaz servicio de radio

La interfaz del servicio de radio un campo de texto para añadir la URL de una radio o canción, un botón de reanudación/pausa y un control de volumen. Será el único servicio que no se podrá redimensionar.

5.6.2 Descripción del Comportamiento de la Interfaz

Siempre que sea posible se realizará una primera validación de los datos en la parte del cliente, de modo que se da una respuesta rápida ante errores. Sin embargo, existen algunas validaciones que solo se podrán hacer en el servidor como, por ejemplo, validar que un nombre de usuario no esté en uso.

Se mostrarán dos tipos de mensajes de error:

- Cuando sea posible, se utilizarán los mensajes de error por defecto del navegador. Por ejemplo, si falta un campo obligatorio o si el formato del email es incorrecto.
- Cuando no sea posible utilizar los mensajes por defecto del navegador, se utilizarán mensajes de error en forma de notificaciones. Estas notificaciones se mostrarán utilizando angular-growl-2 y tendrán la siguiente forma:

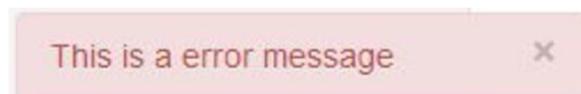


Figura 5.31. Mensaje de error de angular-growl-2

En cuanto a los errores producidos por una navegación errónea por las vistas de la aplicación, se mostrará la siguiente pantalla:

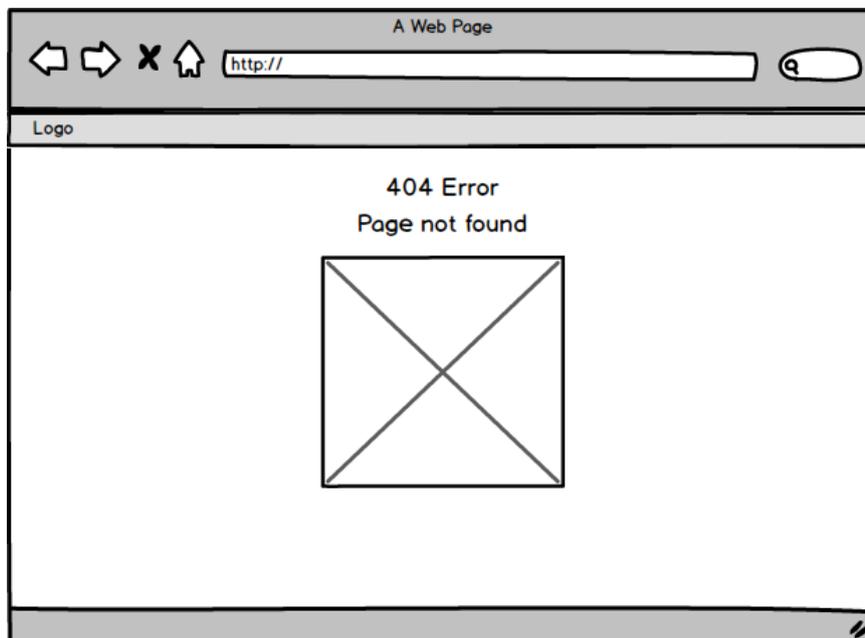


Figura 5.32. Interfaz para errores de navegación

5.6.3 Diagrama de Navegabilidad

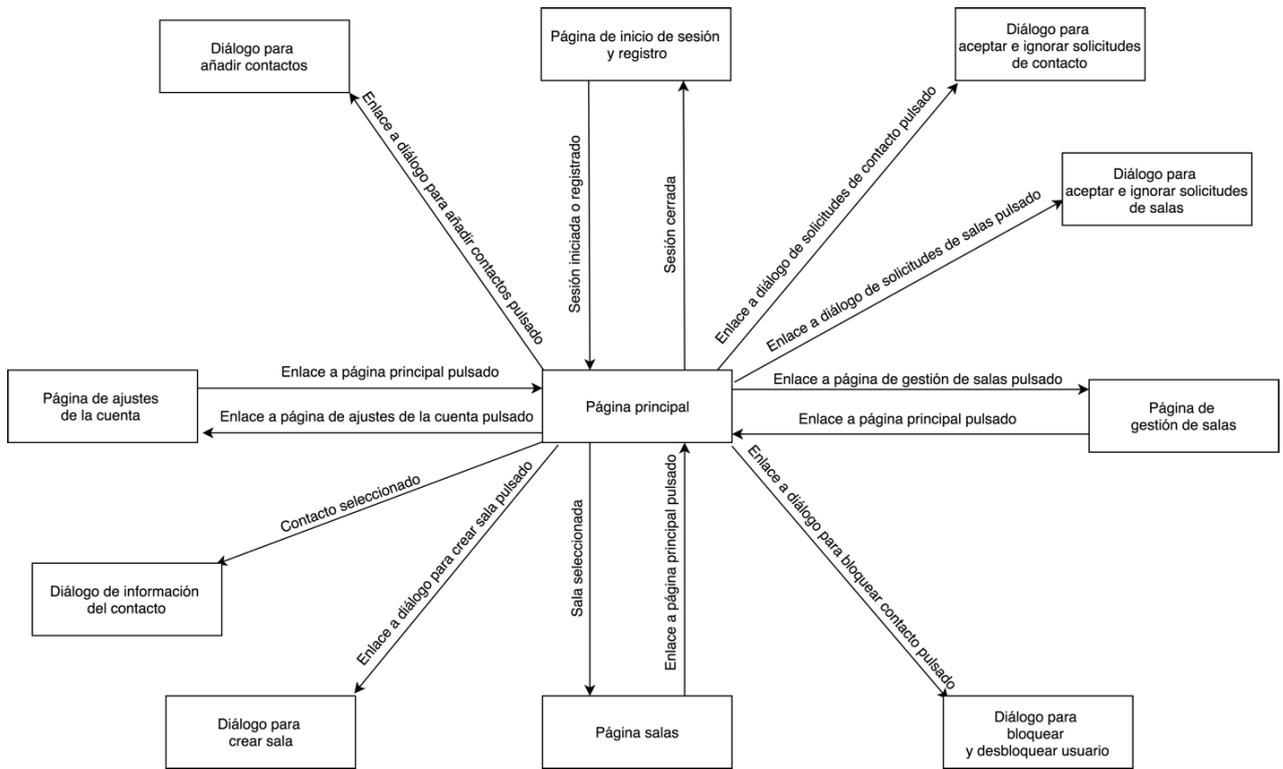


Figura 5.33. Diagrama de navegabilidad

5.7 Especificación del Plan de Pruebas

5.7.1 Pruebas unitarias

Se realizarán pruebas unitarias sobre la API REST para comprobar el correcto funcionamiento de los controladores en la parte del servidor. Se seguirá un desarrollo guiado por pruebas de software (TDD), por lo que primero se desarrollarán las pruebas de la API REST y posteriormente se desarrollará las funciones de la propia API.

Para esto se utilizará Mocha, un framework de testing para Node.js y otros módulos como Should.js, Assert y Supertest, que proporcionan las herramientas para el envío de peticiones y comprobar que el código de respuesta HTTP y valores retornados son los esperados.

Con estos tests se realizan pruebas basadas en los casos de uso encontrados que impliquen la utilización de la API de la aplicación. Aquellas pruebas referentes a la interfaz en las que sea necesario comprobar la aparición de ciertas notificaciones o estén referidas al manejo de los servicios, serán realizadas en las pruebas de integración y del sistema.

Para no tener que ejecutar las pruebas en local cada vez que se realizan cambios, se utilizará el servicio de integración continua Travis CI para desplegar la aplicación y realizar las pruebas cada vez que se envíen nuevos *commits* al repositorio en GitHub. Las pruebas ejecutadas en Travis CI utilizarán una base de datos de alojada en un servidor de GrapheneDB creada exclusivamente para este fin.

5.7.2 Pruebas de integración y del sistema

Estas pruebas tienen como objetivo comprobar que la aplicación en su conjunto funciona sin errores, es decir, que las comunicaciones entre el lado del cliente y el servidor funciona sin problemas.

Para la realización de estas pruebas, se realizarán comprobaciones manuales sobre la aplicación desplegada en distintos navegadores al mismo tiempo. De esta forma se simulará la interacción realizada entre diferentes usuarios.

5.7.3 Pruebas de usabilidad

Para las pruebas de usabilidad se pedirá a personas externas al proyecto que prueben la aplicación y realicen una serie de cuestionarios. No se les aportará más información que los objetivos del software durante sus pruebas para hacer el escenario más realista.

5.7.4 Pruebas de accesibilidad

Las pruebas de accesibilidad tienen como objetivo hacer que la aplicación Web pueda ser correctamente utilizada por el mayor número de personas posibles. En el capítulo de diseño de esta memoria se explicarán las tareas que se llevarán a cabo

5.7.5 Pruebas de rendimiento

Se realizarán pruebas de rendimiento para comprobar el tiempo de respuesta de la aplicación bajo distintos contextos.

5.7.6 Pruebas a partir de casos de uso

<i>Caso de Uso: Registrarse</i>	
Prueba	Resultado Esperado
Registrar usuario con nombre y email válidos y nombre de usuario no utilizado	El sistema posee un usuario más
Registrar usuario con nombre de usuario ya utilizado	El sistema no posee un usuario más y se muestra un mensaje notificándolo

<i>Caso de Uso: Iniciar sesión</i>	
Prueba	Resultado Esperado
Iniciar sesión con nombre de usuario y contraseña correctos	El sistema devuelve un token de sesión y el usuario accede a la página principal de la aplicación
Iniciar sesión con nombre de usuario y contraseña incorrectos	Se muestra un mensaje notificando que el usuario o la contraseña son incorrectos

<i>Caso de Uso: Cambiar idioma</i>	
Prueba	Resultado Esperado
Se pulsa el botón para cambiar idioma a inglés	Los textos mostrados en la aplicación están en inglés.
Se pulsa el botón para cambiar idioma a español	Los textos mostrados en la aplicación están en español.

<i>Caso de Uso: Modificar datos cuenta</i>	
Prueba	Resultado Esperado
Se envía un nombre, apellidos y email válidos.	Se modifican los datos del usuario y se muestra un mensaje notificándolo.
Se envía un nombre, apellidos o email no válidos.	No se modifican los datos del usuario y se muestra un mensaje notificándolo.

<i>Caso de Uso: Crear sala</i>	
Prueba	Resultado Esperado
Un usuario crea una sala con un nombre y descripción válido.	Se añade una nueva sala al sistema donde el usuario es el administrador.
Un usuario crea una sala con un nombre y descripción válido, y especifica participantes con sus permisos.	Se añade una nueva sala al sistema donde el usuario es el administrador y los participantes especificados son moderadores o miembros dependiendo de los permisos dados.
Un usuario crea una sala con un nombre y descripción no válidos.	No se añade una nueva sala al sistema.

<i>Caso de Uso: Enviar solicitud de contacto</i>	
Prueba	Resultado Esperado
Un usuario envía una solicitud de contacto a otro usuario.	Se envía la solicitud de contacto al usuario.
Un usuario envía una solicitud de contacto a otro usuario al que ya envió una solicitud de contacto o que ya es contacto.	No se envía la solicitud de contacto al usuario.

<i>Caso de Uso: Aceptar solicitud contacto</i>	
Prueba	Resultado Esperado
Un usuario acepta una solicitud de contacto.	El usuario tiene un nuevo contacto en su listado de contactos.

Caso de Uso: Ignorar solicitud contacto

Prueba	Resultado Esperado
Un usuario ignora una solicitud de contacto.	El usuario no tiene un nuevo contacto en su listado de contactos.

Caso de Uso: Aceptar solicitud de unión a sala

Prueba	Resultado Esperado
Un usuario acepta una solicitud de unión a sala.	El usuario tiene una nueva sala en su listado de salas.

Caso de Uso: Ignorar solicitud de unión a sala

Prueba	Resultado Esperado
Un usuario ignora una solicitud de unión a sala.	El usuario no tiene una nueva sala en su listado de salas.

Caso de Uso: Acceder a sala

Prueba	Resultado Esperado
Se pulsa en el botón para acceder a una sala.	Se accede a la página de la sala.
Se pulsa en el botón para acceder a una sala en la que nos quitaron el permiso de acceso.	No se accede a la página de la sala.

Caso de Uso: Filtrar salas

Prueba	Resultado Esperado
Se escribe un texto en el campo para filtrar sala.	Se muestran aquellas salas cuyos nombres contienen el texto.

Caso de Uso: Filtrar contactos

Prueba	Resultado Esperado
Se escribe un texto en el campo para filtrar contactos.	Se muestran aquellos contactos cuyos nombres de usuario contienen el texto.

<i>Caso de Uso: Bloquear contacto</i>	
Prueba	Resultado Esperado
Bloquear un contacto.	El contacto desaparece del listado de contactos y es expulsado de todas las salas donde el usuario que lo bloqueó es administrador.
Bloquear un contacto que ya está bloqueado.	No se modifica ningún dato.

<i>Caso de Uso: Desbloquear contacto</i>	
Prueba	Resultado Esperado
Desbloquear un contacto.	El contacto aparece en la lista de contactos.
Desbloquear un contacto que no está bloqueado.	No se modifica ningún dato.

<i>Caso de Uso: Eliminar cuenta</i>	
Prueba	Resultado Esperado
Eliminar cuenta de un usuario.	Se cierra sesión y existe un contacto menos en el sistema.
Eliminar cuenta de un usuario que no existe.	No se modifica ningún dato.

<i>Caso de Uso: Eliminar sala</i>	
Prueba	Resultado Esperado
Eliminar sala.	Los participantes son expulsados y existe una sala menos en el sistema.
Eliminar sala que no existe.	No se modifica ningún dato.

<i>Caso de Uso: Cambiar permisos moderadores</i>	
Prueba	Resultado Esperado
Se modifican los permisos de un moderador.	El moderador pasa a tener permisos de miembro.

Caso de Uso: Eliminar moderadores

Prueba	Resultado Esperado
Eliminar moderador de una sala.	Existe un participante menos en la sala.

Caso de Uso: Cambiar permisos solicitud

Prueba	Resultado Esperado
Cambiar permisos.	En la solicitud se especifican otros permisos.

Caso de Uso: Eliminar solicitud

Prueba	Resultado Esperado
Eliminar solicitud de unión a sala.	Existe una solicitud de unión a la sala menos.

Caso de Uso: Cambiar permisos miembros

Prueba	Resultado Esperado
Cambiar permisos de miembro.	El usuario pasa a tener permisos de moderador.

Caso de Uso: Enviar solicitud de unión a sala

Prueba	Resultado Esperado
Enviar solicitud de unión a sala.	Existe una solicitud de unión más en la sala.
Enviar solicitud de unión a sala cuando las solicitudes de las salas y los participantes suman 4.	No se envía la solicitud y se notifica de ello.

Caso de Uso: Eliminar miembro

Prueba	Resultado Esperado
Eliminar miembro de una sala.	Existe un participante menos en la sala.

<i>Caso de Uso: Abandonar sala</i>	
Prueba	Resultado Esperado
Abandonar sala.	Existe un participante menos en la sala.
Abandonar sala en la que no se participa.	No se modifica ningún dato.

<i>Caso de Uso: Abrir servicio</i>	
Prueba	Resultado Esperado
Abrir servicio.	Aparece el servicio seleccionado en la pantalla.
Abrir servicio que ya fue abierto.	No pasa nada y se muestra un mensaje al usuario.

<i>Caso de Uso: Cerrar servicio</i>	
Prueba	Resultado Esperado
Cerrar servicio.	El servicio desaparece de la pantalla.

<i>Caso de Uso: Mover servicio</i>	
Prueba	Resultado Esperado
Mover servicio.	El servicio cambia su posición.

<i>Caso de Uso: Redimensionar servicio</i>	
Prueba	Resultado Esperado
Redimensionar servicio.	El servicio cambia de tamaño y su contenido se adapta a él.

<i>Caso de Uso: Crear dibujo colaborativo</i>	
Prueba	Resultado Esperado
Seleccionar botón de una figura.	Aparece la figura en el dibujo.
Seleccionar lápiz y clicar y mover con el puntero en el canvas.	Aparece un dibujo con la forma del recorrido del puntero.

<i>Caso de Uso: Compartir URL video</i>	
Prueba	Resultado Esperado
Se comparte URL de video.	Aparece el video en el servicio de video compartido y en el de los usuarios conectados al servicio.
Se comparte URL de video no soportado.	No pasa nada.

<i>Caso de Uso: Compartir presentación</i>	
Prueba	Resultado Esperado
Se comparte presentación.	Aparece la presentación en la pantalla del servicio de presentaciones y en el de los usuarios conectados al servicio.

<i>Caso de Uso: Participar en videollamada</i>	
Prueba	Resultado Esperado
Abrir servicio de videollamada.	Aparece en el servicio los videos de los usuarios conectados a la videollamada y el video local.
Usuario pulsa botón de silenciar micrófono.	Los otros usuarios conectados a la videollamada no pueden escuchar el audio del usuario.
Usuario pulsa botón de silenciar altavoces.	El usuario no puede escuchar a los otros usuarios conectados a la videollamada.
Usuario pulsa botón de pausar transmisión de video.	Los otros usuarios conectados a la videollamada no pueden ver el video del usuario.

<i>Caso de Uso: Enviar mensaje de texto</i>	
Prueba	Resultado Esperado
El usuario envía mensaje de texto.	Aparece el mensaje en el servicio de chat de texto, y los usuarios conectados al servicio pueden ver el mensaje. Además, el usuario ve cuantos participantes recibieron el mensaje.

<i>Caso de Uso: Enviar fichero</i>	
Prueba	Resultado Esperado
El usuario envía una imagen.	Aparece la imagen en el servicio de chat de texto y los usuarios conectados al servicio pueden verla. Además, el usuario ve cuantos participantes recibieron el mensaje.

El usuario envía un documento.	Aparece una imagen de un documento en el servicio de chat de texto y los usuarios conectados al servicio la ven. Además, el usuario ve cuantos participantes recibieron el mensaje.
--------------------------------	---

<i>Caso de Uso: Descargar fichero</i>	
Prueba	Resultado Esperado
El usuario pulsa en el fichero para descargarlo.	Aparece la ventana de diálogo por defecto del navegador que permite descargar el fichero.

Capítulo 6. Diseño del Sistema

Basándome en los subsistemas definidos en el anterior capítulo, se han modelado los siguientes diagramas y diseños a utilizar durante la implementación del proyecto.

6.1 Arquitectura del Sistema

En esta sección se muestran los diagramas de paquetes, componentes y de despliegue.

6.1.1 Diagramas de Paquetes

Se dividirá la explicación del diagrama de paquetes en dos puntos, primero se mostrará el diagrama referente a los elementos de la parte del cliente y, por último, el diagrama para la parte del servidor.

6.1.1.1 Cliente

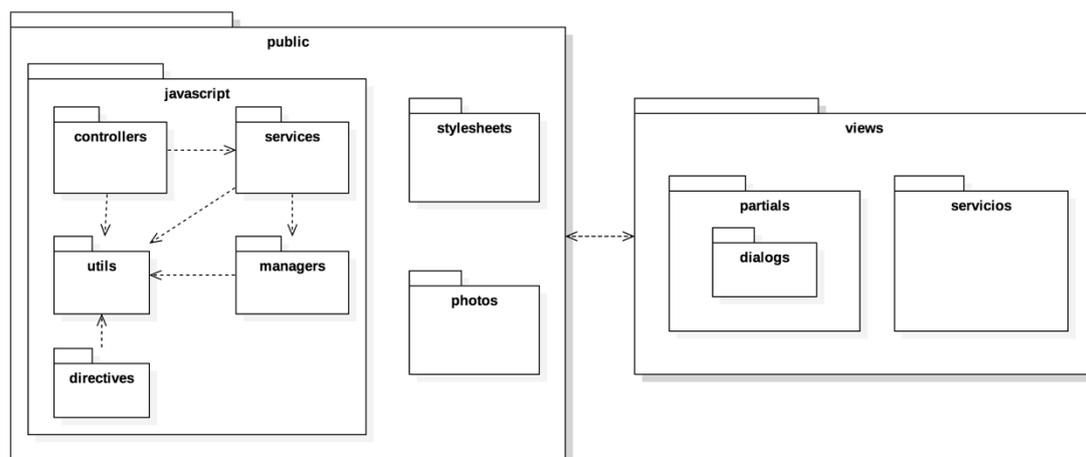


Figura 6.1. Diagrama de paquetes del cliente

En este diagrama de paquetes se representan todos los elementos que serán utilizados para el correcto funcionamiento de la parte del cliente. Encontramos dos paquetes principales “views” y “public” que, además de contener ficheros, estarán subdivididos a su vez en otros paquetes.

6.1.1.1.1 Paquete views

Este paquete incluye los ficheros EJS que forman las vistas de la aplicación. Las vistas incluidas son:

- **index.ejs:** Página de inicio de sesión y registro.
- **mainPage.ejs:** Página principal que será mostrada una vez se inicia sesión.
- **chatroom.ejs:** Página que será mostrada cuando se accede a una sala.
- **manageRooms.ejs:** Página para la gestión de salas.
- **profileSettings.ejs:** Página en la que se accederá a los ajustes de la cuenta.
- **about.ejs:** Página que contendrá información de contacto.
- **error.ejs:** Página que se mostrará ante un error en la navegación entre las vistas.
- **legal.ejs:** Página con información legal.

Este paquete a su vez incluye los paquetes “partials” y “servicios” que contienen EJS utilizados en las vistas principales.

Los elementos de este paquete utilizan los controladores, directivas, hojas de estilo y fotos localizados en el paquete “public”.

6.1.1.1.1.1 *Paquete partials*

Este paquete incluye ficheros EJS utilizados por los EJS principales de la aplicación. Existe un EJS para la cabecera de las páginas y otro que contiene enlaces a scripts, hojas de estilo y controladores del paquete “public”.

6.1.1.1.1.1.1 *Paquete dialogs*

Contendrá los EJS de los cuadros de diálogo mostrados en la aplicación. Por ejemplo, el cuadro de diálogo para crear salas o el utilizado para aceptar solicitudes de contacto.

6.1.1.1.1.2 *Paquete servicios*

Incluirá ficheros EJS para la visualización de los servicios de las salas.

6.1.1.1.2 **Paquete public**

En este paquete se almacenarán todos aquellos scripts, hojas de estilo y fotos necesarios para la correcta visualización de las vistas.

6.1.1.1.2.1 *Paquete javascript*

Este paquete incluirá diversos scripts de JavaScript que serán explicados en los siguientes puntos.

6.1.1.1.2.1.1 *Paquete controllers*

Como ya se explicó con anterioridad, en este proyecto se utilizará el framework MVC AngularJS para el desarrollo de la parte del cliente. Por ello, en este paquete se localizan los controladores que son accedidos desde los ficheros EJS.

Será necesario un número mayor de controladores al planificado en la fase de análisis debido a la necesidad de ejecutar ciertas funciones automáticamente cuando se abre un

cuadro de diálogo o cuando el usuario se dirige a cierta página. Estas funciones tendrán como principal objetivo recuperar información de la base de datos que se mostrará en las vistas, por ello no es posible, por ejemplo, utilizar únicamente un controlador para todas las gestiones de las salas. En el diagrama de clases se mostrarán todos los controladores que se necesitarán.

6.1.1.1.2.1.2 Paquete managers

Debido a las diversas funciones que tienen que realizar los controladores, todos aquellos métodos encargados del correcto funcionamiento de los servicios y sus comunicaciones con otros usuarios serán incluidos en unos ficheros llamados “Managers”.

6.1.1.1.2.1.3 Paquete services

En este paquete se incluye el servicio “WebSocketsService”. Su objetivo es gestionar los mensajes llegados desde el servidor al cliente mediante el servidor de WebSockets y los managers descritos en el anterior paquete.

6.1.1.1.2.1.4 Paquete utils

En este paquete se incluirá un script “Utils”, que será utilizado por los demás elementos del paquete “javascript”.

6.1.1.1.2.1.5 Paquete directives

Algunas validaciones u otras funciones serán realizadas gracias a las directivas localizadas en este paquete. Estas directivas serán utilizadas en los ficheros EJS y serán:

- **usernameValidation:** Directiva para validar que un nombre de usuario está disponible.
- **passwordValidation:** Directiva para validar en el registro que la contraseña y la contraseña de confirmación son iguales.
- **confirmAction:** Directiva para mostrar las ventanas emergentes por defecto de los navegadores.
- **tooltip:** Directiva para mostrar tooltips.

6.1.1.1.2.2 Paquete photos

Todas las fotografías de la aplicación como el logo o banderas se almacenarán en este paquete.

Es importante dejar claro que las fotos de las salas y usuarios no serán almacenadas en este paquete, sino que se utiliza la plataforma Cloudinary para alojar esas fotos.

6.1.1.1.2.3 Paquete stylesheets

Los ficheros CSS usados para el diseño gráfico de las vistas se localizarán en este paquete. Existen un CSS para cada vista.

6.1.1.2 Servidor

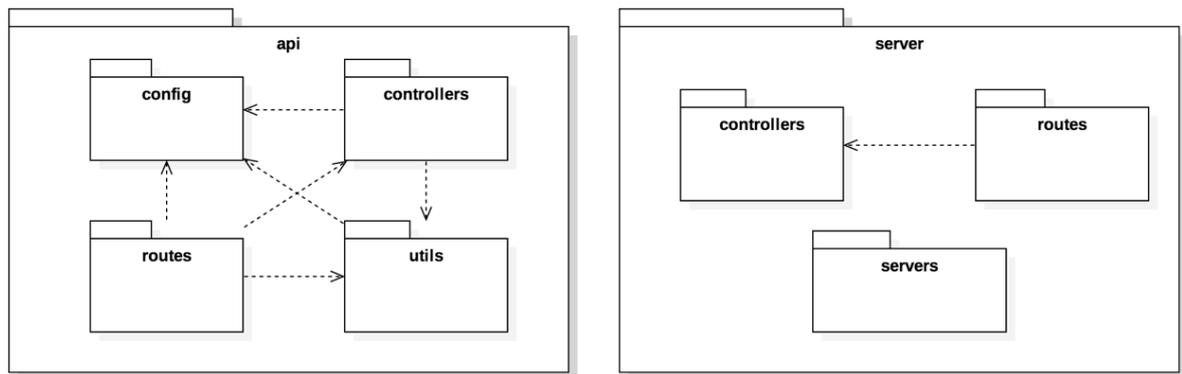


Figura 6.2. Diagrama de paquetes del servidor

En este diagrama de paquetes se representan todos los elementos que será utilizados para el correcto funcionamiento de la parte del servidor. Encontramos dos paquetes principales “api” y “server” que, tal y como ocurrió en el diagrama de cliente, contendrán ficheros y estarán subdivididos a su vez en otros paquetes.

6.1.1.2.1 Paquete api

6.1.1.2.1.1 Paquete config

La clase para con la información de acceso a la base de datos *DB* se encontrará en este paquete.

6.1.1.2.1.2 Paquete controllers

Tras un pequeño estudio, se ha decidido que el número de controladores definidos en el análisis es el correcto, por lo que estos tres controladores serán almacenados en este paquete.

6.1.1.2.1.3 Paquete routes

Incluirá las clases *Routes* y *Middleware* definidas en el análisis.

6.1.1.2.1.4 Paquete utils

Este paquete incluirá la clase de utilidades *Utils* y la de la gestión de las validaciones al sistema *Passport*.

6.1.1.2.2 Paquete server

6.1.1.2.2.1 Paquete controllers

Incluye el controlador *PagesController*.

6.1.1.2.2 Paquete routes

Al igual que ocurre en el paquete *api*, se almacenará las clases *Routes* y *Middleware*.

6.1.1.2.3 Paquete servers

Este paquete contendrá las clases *HTTP* y *Websockets* definidas en el análisis. Gracias a Express será posible ejecutar el servidor HTTP y el servidor de WebSockets a partir de esas clases.

6.1.2 Diagramas de Componentes

En este apartado se mostrarán dos diagramas de componentes, uno que representa la parte del cliente y otro para la parte del servidor.

6.1.2.1 Diagrama de Componentes del Cliente

Como se ha explicado con anterioridad en esta memoria, se utilizará el framework MVC AngularJS para el desarrollo de este proyecto. Esto implica la aparición de componentes como controladores y servicios que serán agrupados en un módulo llamado *copernicus*, y también de directivas y vistas.

A continuación, se muestra el diagrama de componentes que constituye la parte del cliente de esta aplicación. Posteriormente, se mostrará en el diagrama de despliegue como esta parte del cliente interactúa con el servidor HTTP y el de WebSockets.

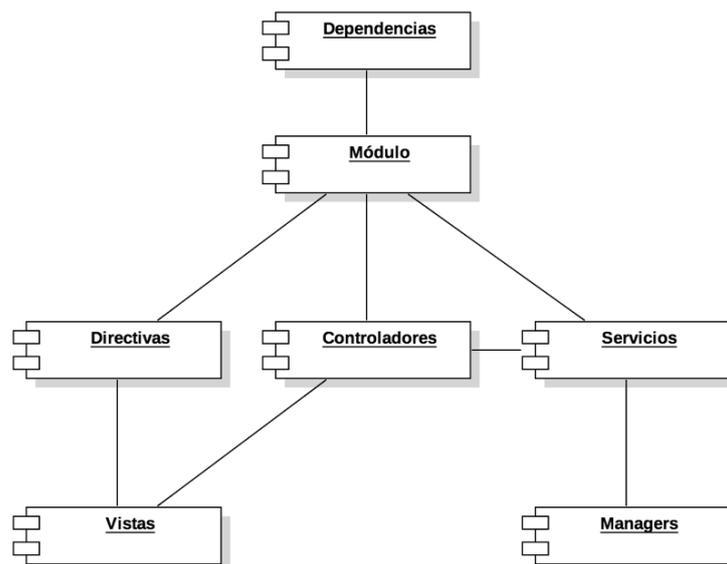


Figura 6.3. Diagrama de componentes del cliente

6.1.2.2 Dependencias

Para el desarrollo de esta aplicación Web, será necesario la utilización de módulos desarrollados por personas ajenas al proyecto.

6.1.2.3 Módulo

Se desarrollará un módulo llamado *copernicus*, el cual será el encargado de agrupar todos los controladores, componentes y directivas desarrollados. También contendrá los textos en inglés y español utilizados en la aplicación.

6.1.2.4 Directivas

Serán necesarias ciertas directivas que se incluirán en los elementos de las vistas. Estas se utilizarán, por ejemplo, para validar un nombre de usuario o comprobar que la contraseña y contraseña de confirmación coinciden.

6.1.2.5 Controladores

Los controladores serán utilizados para obtener y mostrar información utilizando las vistas. La comunicación entre estos componentes será posible gracias al objeto *\$scope* que es proporcionado por AngularJS. De este modo, en los controladores se definirán aquellas funciones que serán almacenadas en el *\$scope* y que podrán ser accedidas desde las vistas.

Otro objetivo de los controladores es comunicarse con la API REST de la aplicación para obtener y almacenar información en la base de datos.

6.1.2.6 Managers

Uno de los objetivos de este proyecto es proveer a sus usuarios con comunicaciones en tiempo real. Estas comunicaciones serán posible con lo que se ha bautizado “Servicios” y que ya se han citado en varias ocasiones al largo de esta memoria. Los managers serán los encargados de gestionar estos servicios y permitir así las comunicaciones entre los usuarios.

En la fase de análisis no se había planteado la existencia de este componente, sin embargo, los controladores planeados contendrían mucho código y además de que dificultarían las comunicaciones con el servidor de WebSockets. Por ello, se ha decidido que los controladores cederán todas las responsabilidades relativas a la gestión de los servicios de las salas a los managers.

6.1.2.7 Vistas

Son los ficheros EJS que proveen la interfaz de esta aplicación. Como ya se ha explicado en los componentes anteriores, las vistas utilizarán las directivas para realizar ciertas acciones y, gracias al objeto *\$scope*, se comunicarán con los controladores.

6.1.2.8 Servicios

Solo existirá uno de estos servicios de AngularJS, el cual se encargará de la gestión de los mensajes de WebSockets recibidos y de redirigirlo al manager correspondiente. Es una pieza clave para permitir las comunicaciones en tiempo real entre los usuarios.

6.1.2.9 Diagrama de Componentes del Servidor

En el diagrama mostrado en esta sección se representa la estructura que se seguirá en la parte del servidor.

Se quiere aclarar que el componente *Express.js* representa el framework de Node.js con el mismo nombre. Se muestra para explicar que el servidor HTTP es utilizado para servir las vistas al usuario y por la API REST a través de Express.

El servidor de WebSockets no será utilizado en esas tareas, si no que será creado a partir del servidor de HTTP y se encargará de gestionar los mensajes relativos a los servicios de las salas.

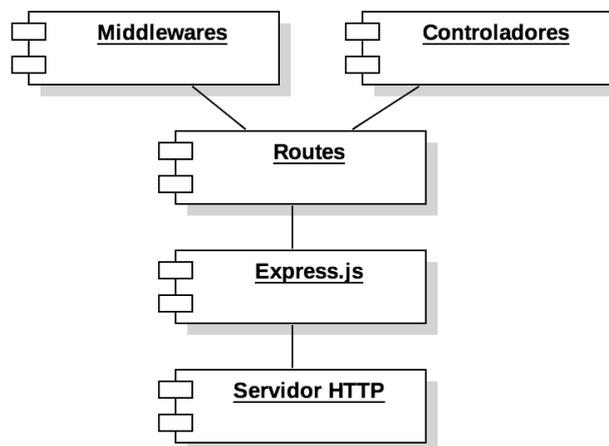


Figura 6.4. Diagrama de componentes del servidor

6.1.3 Diagramas de Despliegue

En este apartado se mostrará el diagrama de despliegue en el que se representan el hardware y componentes a utilizar.

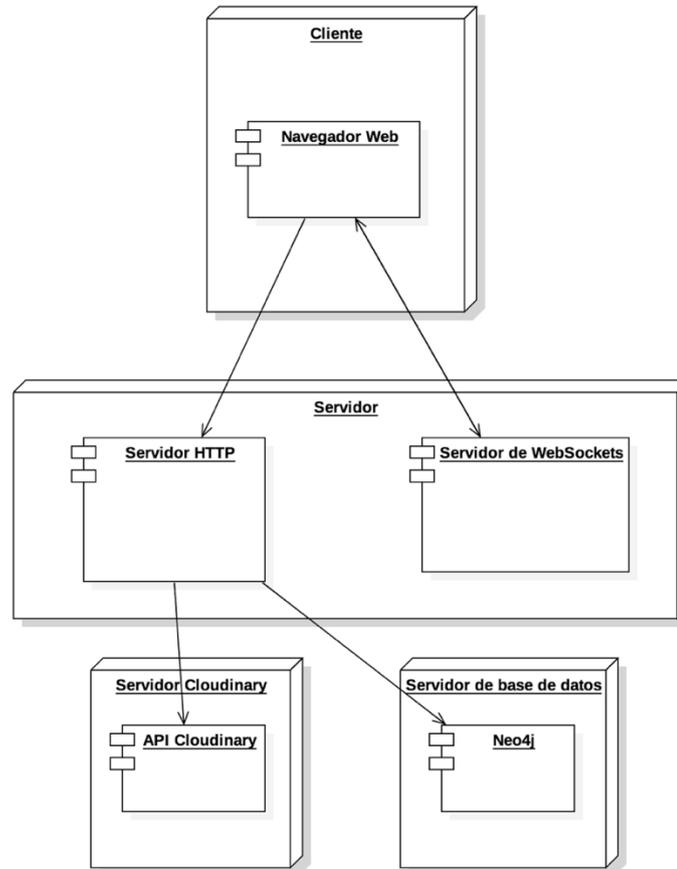


Figura 6.5. Diagrama de despliegue

6.1.3.1 Cliente

En esta máquina se alojará la parte del cliente de la aplicación Web desarrollada.

Se utilizará un navegador web para que sus componentes puedan interactuar con un servidor HTTP que le servirá las vistas de la aplicación y acceso a la API REST. También se comunicará con un servidor de WebSockets que permitirá las comunicaciones en tiempo real de los usuarios en las salas.

6.1.3.2 Servidor

En esta máquina se alojará la parte del servidor de la aplicación. Aquí encontraremos un servidor HTTP encargado de recibir peticiones de páginas o peticiones para una API REST, y un servidor de WebSockets. Como ya se ha comentado, gracias a este servidor de WebSockets serán posibles las comunicaciones bidireccionales entre clientes y servidor permitiendo las comunicaciones en tiempo real entre los usuarios de las salas.

6.1.3.3 *Servidor Cloudinary*

A través de la API de Cloudinary será posible alojar y recuperar de esta plataforma las fotografías de perfil de los usuarios y las salas.

6.1.3.4 *Servidor base de datos*

En esta máquina estará alojado la base de datos Neo4j. Esta será accedida por los usuarios a través de la API REST de la aplicación, la cual accederá a ella utilizando el driver Seraph.

6.2 Diseño de Clases

Como se explicó en el diagrama de clases preliminar, era posible que fuera necesaria una actualización del diagrama debido a problemas o mejoras encontradas que no se habían considerado en el análisis. A continuación, se explicarán los cambios más destacados.

El principal cambio es la necesidad de un número mayor de controladores. Esto es debido, en gran parte, al número de páginas y diálogos para los cuales se requiere obtener cierta información la cual será visualizada una vez que las vistas son mostradas. De esta manera, se pueden dedicar cada controlador a realizar ciertas gestiones, y, cuando una vista es accedida, se ejecutarán instrucciones en los controladores que recuperarán y almacenarán los datos en variables a las que las vistas accederán. Un ejemplo es la recuperación de los contactos o las salas en las que el usuario participa.

Otro cambio destacable es la creación de unas clases denominadas “Managers” que se encargarán de la gestión de los servicios de las salas.

6.2.1 Diagrama de Clases

Para mostrar el diagrama de clases, primero se mostrará las clases pertenecientes a cada paquete definido en el apartado [6.1.1 Diagramas de Paquetes](#). Hay que tener en cuenta que paquetes como *views*, *stylesheets* o *photos* no serán incluidos porque solo incluirán ficheros EJS, CSS e imágenes.

Finalmente, se mostrarán las relaciones entre las clases en los distintos paquetes.

6.2.1.1 Paquete cliente

6.2.1.1.1 Paquete public

6.2.1.1.1.1 Paquete javascript

En este paquete estarán alojadas todas aquellas clases JavaScript utilizadas en el lado del servidor. Esto incluye los controladores, servicios, managers, clases de utilidades y directivas.

6.2.1.1.1.1.1 Paquete controllers

El hecho de disponer del objeto *\$scope* proporcionado por el framework MVC AngularJS, hace que muchos métodos no necesiten parámetros. Esto se debe a que los valores introducidos en las vistas serán almacenados en el *\$scope* el cual será inyectado en los controladores.

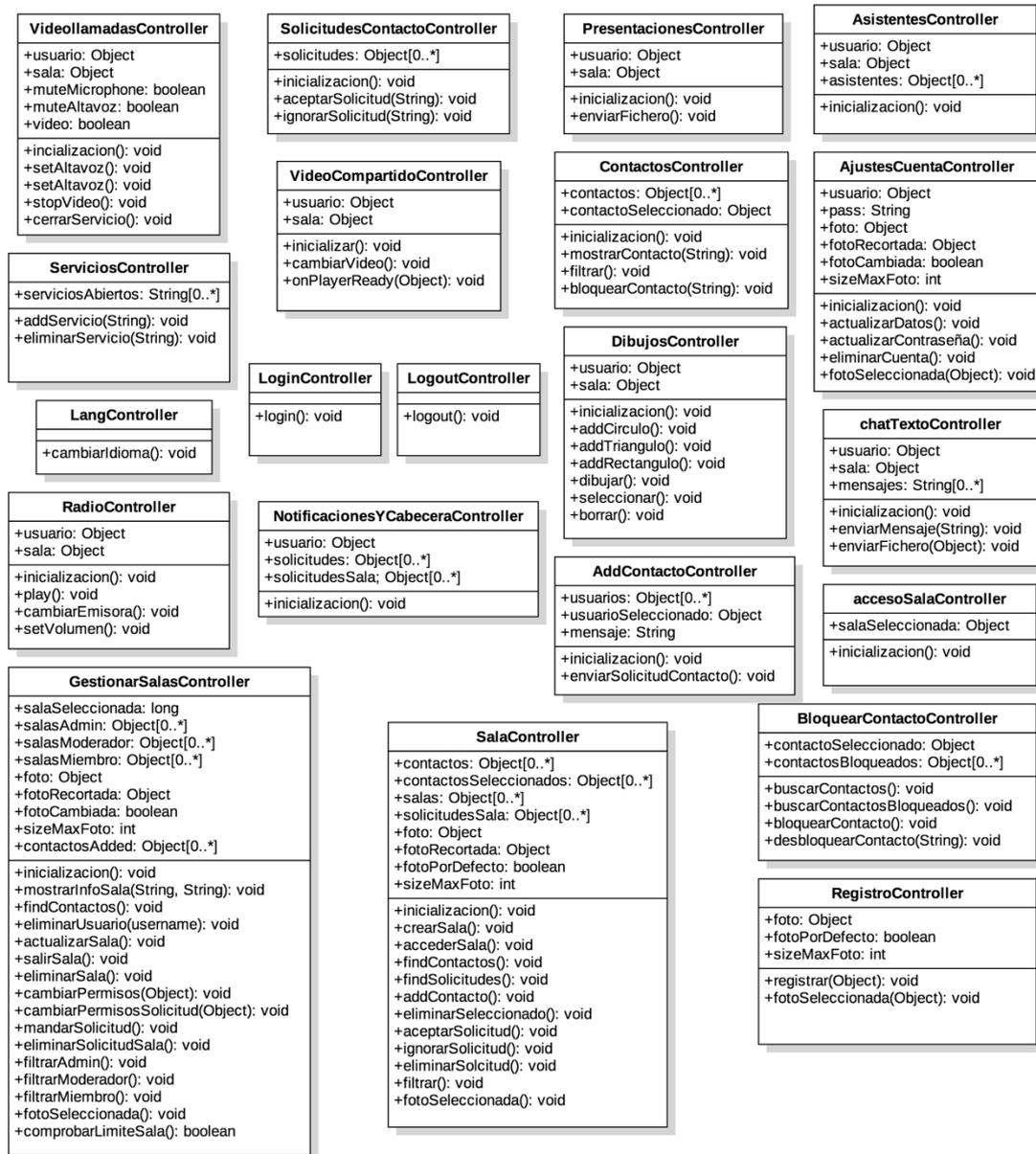


Figura 6.6. Clases controllers (Paquete public)

6.2.1.1.1.2 Paquete services

Este paquete incluirá una clase encargada de la gestión de los mensajes de WebSockets recibidos. Tendrá una instancia de cada manager, como se verá en el diagrama de clases, de modo que pueda llamar a sus métodos en función del mensaje recibido.

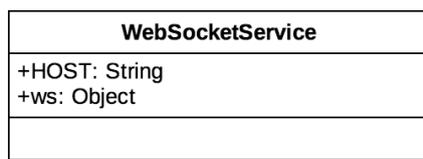


Figura 6.7. Clases services

6.2.1.1.1.3 Paquete utils

Este paquete incluirá una clase de utilidades.

Utils
+IsJsonString(String): boolean +checkDatabaseError(Object): boolean +checkIdSalaExists(long): boolean +checkParticipante(long): boolean +mensajeSuccess(String): void +mensajeInfo(String): void +mensajeError(String): void +mensajeSuccessSinTiempo(String): void +mensajeInfoSinTiempo(String): void +mensajeErrorSinTiempo(String): void

Figura 6.8. Clases utils

6.2.1.1.1.4 Paquete managers

El objetivo de los managers es recibir y enviar mensajes a través del servidor de WebSockets. De modo que actúan sobre los servicios de las salas en función de los mensajes recibidos, además de con *AsistentesManager*. Una clase utilizada para llevar un control en tiempo real de los usuarios que se conectan y desconectan de una sala.

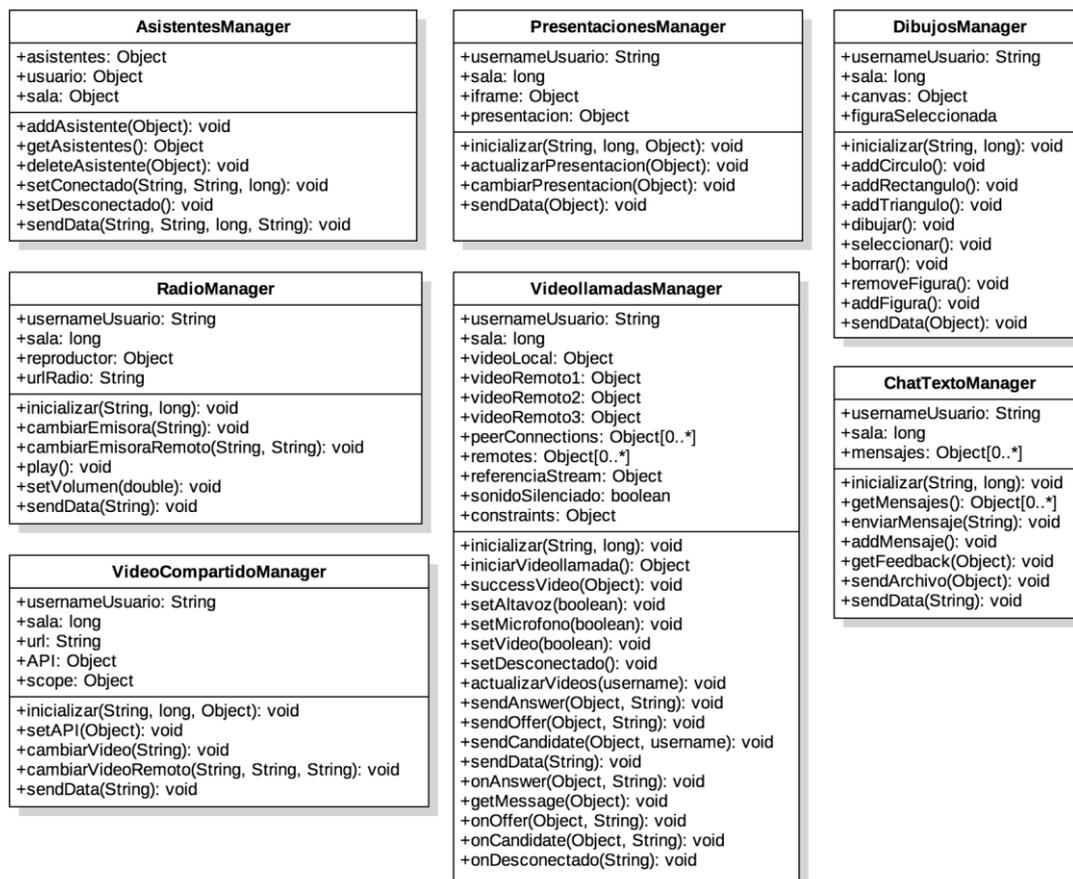


Figura 6.9. Clases managers

6.2.1.1.1.5 Paquete directives

En el paquete *directives* se encuentran las directivas de AngularJS que se incluyen en las etiquetas HTML para realizar ciertas acciones como validaciones. Estas no poseen atributos ni métodos, por lo que no se representarán como clases. Pero si se puede concretar que se necesitarán, al menos, las directivas para:

- Comprobar que la contraseña y la contraseña de comprobación coinciden.
- Comprobar que un nombre de usuario está disponible.
- Mostrar las ventanas emergentes por defecto del navegador.

6.2.1.3 Paquete servidor

6.2.1.3.1 Paquete api

6.2.1.3.1.1 Paquete config

En este paquete se encontrará una clase llamada *DB*, la cual contendrá un atributo *config* que almacenará la dirección y credenciales de la base de datos.

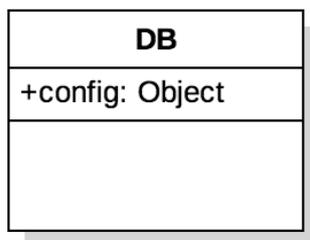


Figura 6.11. Clases config

6.2.1.3.1.2 Paquete controllers

El paquete *controllers* contendrá tres clases encargadas de recibir peticiones HTTP y realizar las validaciones y consultas oportunas contra la base de datos Neo4j. Dos de estas clases también se encargarán de conectarse con los servidores de Cloudinary para la recuperación y persistencia de fotos.

Cada método recibirá un objeto de solicitud y un objeto de respuesta.

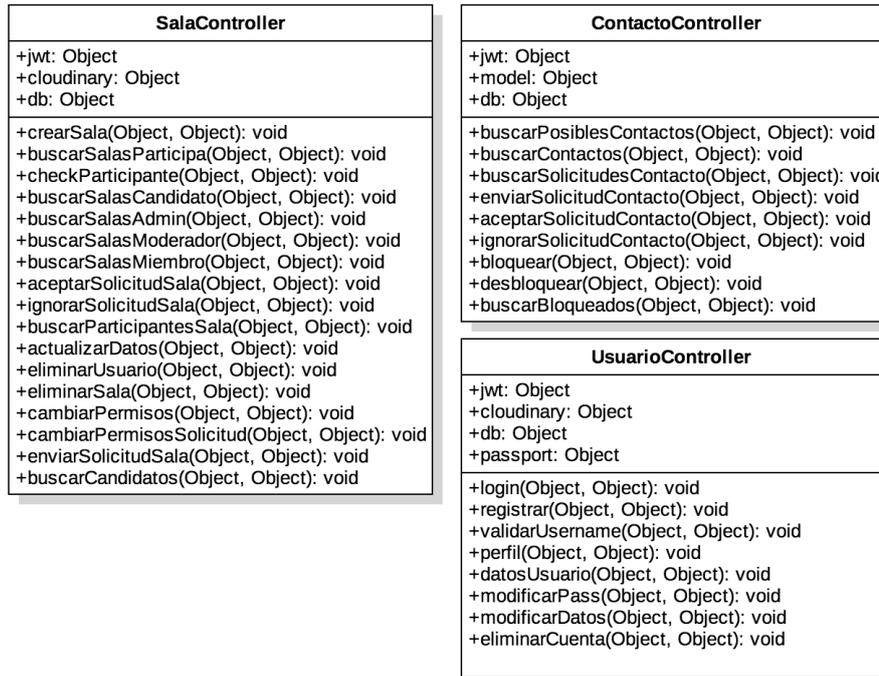


Figura 6.12. Clases controllers (Paquete api)

6.2.1.3.1.3 Paquete routes

En este paquete se encontrará una clase *Routes* encargada de gestionar las peticiones a la API REST y redirigirlas al correspondiente controlador, y una clase *Middleware*, cuyos métodos serán invocados por la clase *Routes* antes de que los controladores reciban las peticiones. De este modo se realizarán las comprobaciones oportunas para decidir si las peticiones deberían continuar hasta el controlador.

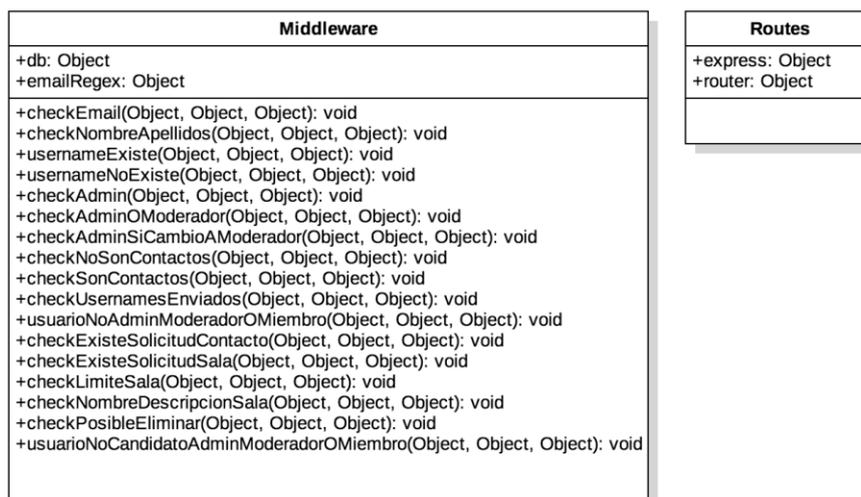


Figura 6.13. Clases routes (Paquete api)

6.2.1.3.1.4 Paquete utils

En este paquete se encuentra una clase de utilidades que será utilizada por algunas clases del paquete api. También se encontrará Passport donde se utilizará el módulo con el mismo nombre para validar las credenciales de los usuarios.

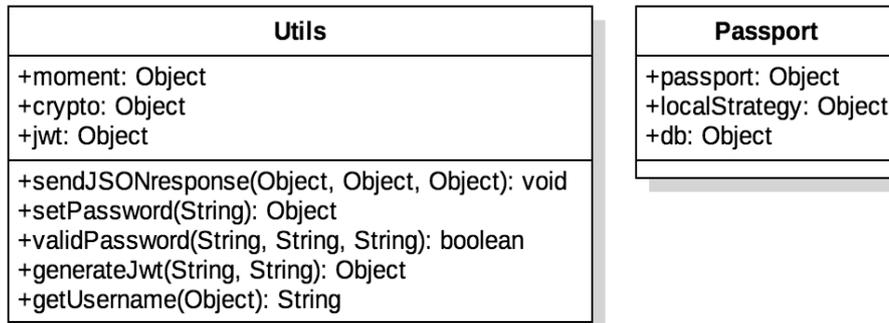


Figura 6.14. Clases utils

6.2.1.4.1 Paquete server

6.2.1.4.1.1 Paquete controllers

El paquete controllers contendrá una única clase encargada de recibir peticiones HTTP, y devolver la vista correspondiente. Cada método recibirá un objeto de solicitud y un objeto de respuesta.

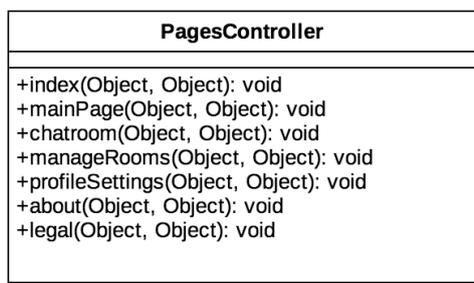


Figura 6.16. Clases controllers (Paquete server)

6.2.1.4.1.2 Paquete routes

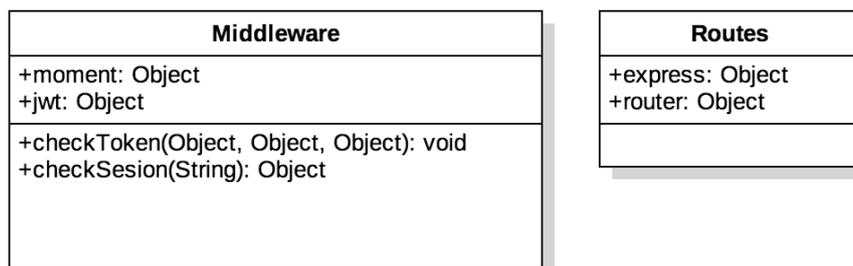


Figura 6.17. Clases routes (Paquete server)

6.2.1.4.1.3 Paquete servers

Aquí se encontrarán dos clases utilizadas para la ejecución de los servidores HTTP y de WebSockets.

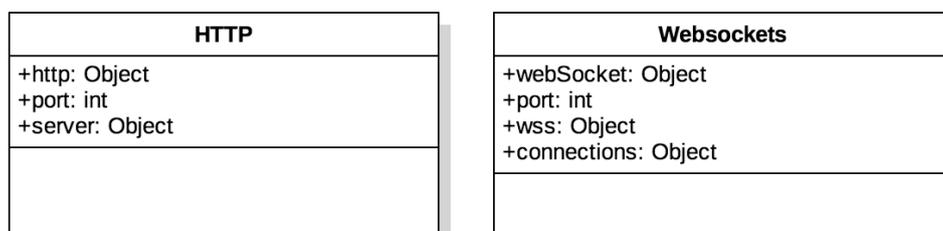


Figura 6.18. Clases servers

6.2.1.5 Diagrama de clases paquete server

Cabe destacar del siguiente diagrama que las clases Websockets y HTTP serán utilizadas por Express para la ejecución de los servidores HTTP y de WebSockets.

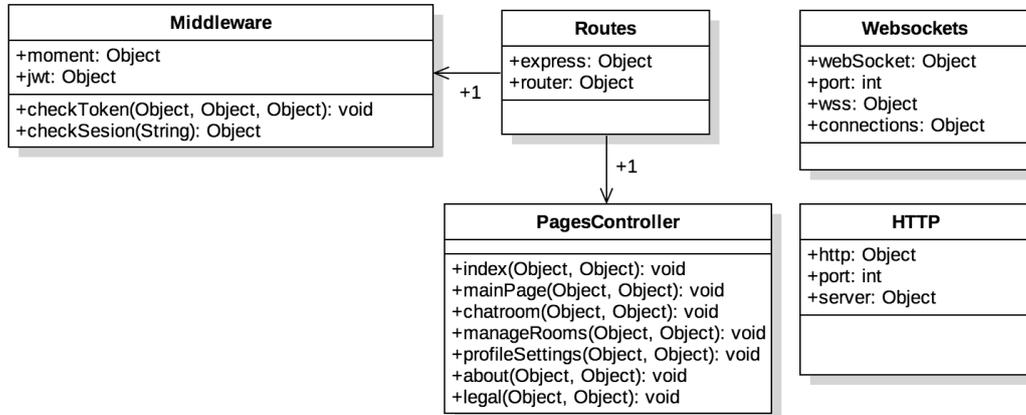


Figura 6.19 Diagrama de clases server

6.3 Diagramas de Interacción

Para la realización de los siguientes diagramas de interacción, se utilizarán algunos de los casos de uso más notables para así facilitar la posterior implementación. La mayoría de casos de uso que no se muestran, seguirán un modelo similar al de los diagramas que se describirán en los siguientes puntos.

6.3.1 Crear sala

Se mostrarán aquellos pasos necesarios para que un usuario cree una sala. Previamente a la creación, el usuario indicará dos contactos a los que enviar solicitudes de unión, pero decidirá eliminar una de estas. Finalmente creará una sala para la que se enviará una solicitud de unión.

Con este diagrama se pretende enseñar como el usuario se comunicará con los controladores y estos, a su vez, se comunicarán con la parte del servidor para llevar a cabo las acciones necesarias y obtener una respuesta. Cabe destacar que, como esta acción involucra a clases de la parte del cliente y servidor, se muestra para cada clase el paquete en el que se localiza para evitar confusiones.

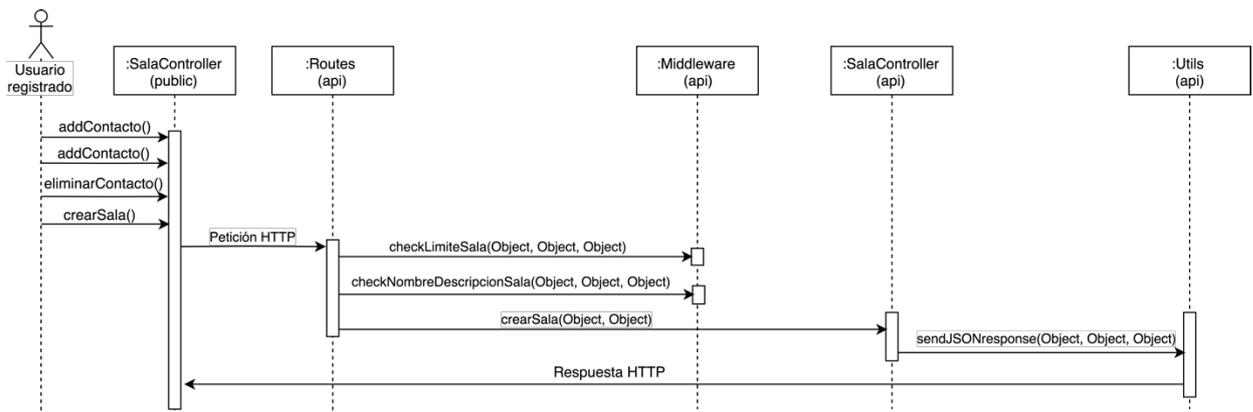


Figura 6.20. Diagrama de Interacción crear sala

Una vez que el usuario ha introducido el nombre y descripción de la sala, indicará en un campo con autocompletar el usuario al que quiere enviar una solicitud de unión. Posteriormente, pulsará un botón que invocará al método `addContacto` de `SalaController`. Esta acción la realizará dos veces lo que conllevará que se muestren dos contactos en una tabla.

El usuario puede cambiar de idea y eliminar a uno de los contactos al que se enviará una solicitud de unión. Para ello pulsará el botón correspondiente que invocará al método `eliminarContacto` y lo eliminará de la tabla.

Una vez están introducidos los datos de la sala y el usuario al que se enviará la solicitud de unión, se pulsará un botón para crear la sala lo que llamará a *crearSala*. Esto provocará que *SalaController* envíe una petición HTTP a la API REST. Esta petición será gestionada por *Routes*, el cual invocará dos métodos de *Middleware* para comprobar la validez de la petición.

Una vez terminadas las comprobaciones del *Middleware*, se invocará a *crearSala* de *SalaController*, el cual añadirá la sala y la solicitud a la base de datos. Finalmente se invocará al método *sendJSONresponse*, que se encargará de devolver una respuesta a *SalaController* localizado en la parte del cliente.

6.3.2 Mostrar moderadores y miembros de una sala y eliminarlos

Con este diagrama se muestra como un administrador podrá eliminar a moderadores o miembros de una sala.

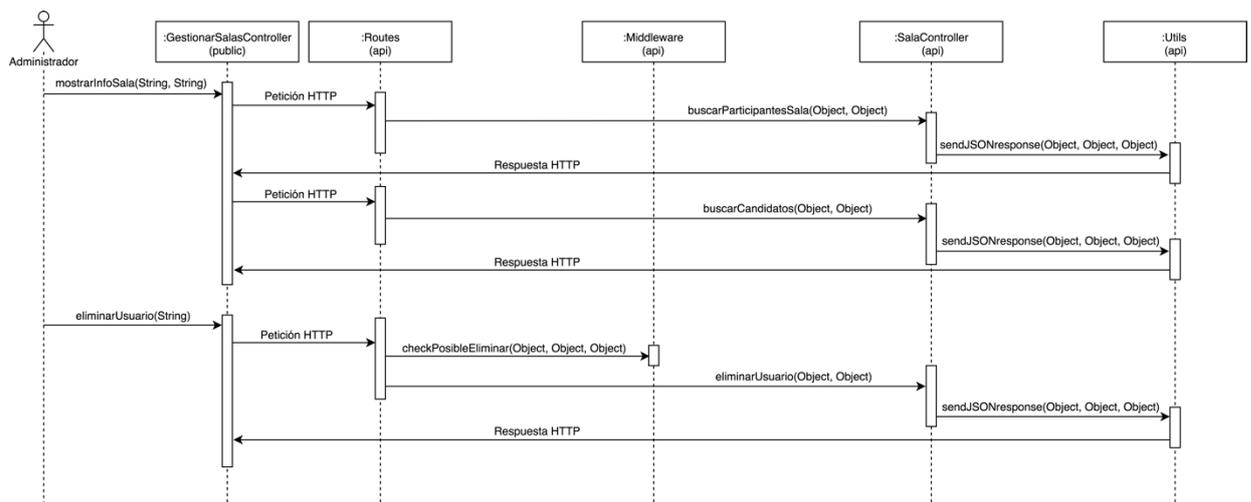


Figura 6.21. Diagrama de Interacción mostrar moderadores y miembros de una sala y eliminarlos

Primero de todo, el usuario seleccionará una sala, en la que es administrador, para visualizar su información. Para ello pulsará un botón que llamará a *mostrarInfoSala*, lo que provocará que se realicen dos peticiones contra la API REST. Una de estas peticiones es para recuperar las personas que actualmente participan en la sala, y la otra es para recuperar los candidatos a unirse a ella, es decir, a los que se envió una solicitud de unión.

Una vez se puede visualizar la información de la sala, se dispondrá de botones para eliminar a los moderadores y participantes. El usuario pulsará un botón para eliminar al usuario que desee de la sala, lo que invocará al método *eliminarUsuario*. Con este método se enviará una petición HTTP con la información necesaria para eliminar al usuario de esa sala. *Routes* recibirá la petición y llamará a un método de *Middleware* para comprobar

si es posible eliminar a ese usuario. Una vez realizadas las comprobaciones, el método *eliminarUsuario* eliminará la relación entre el usuario y la sala, y finalmente se devolverá una respuesta HTTP a través del método *sendJSONresponse* proporcionado por *Utils*.

6.3.3 Abrir chat de texto y mandar mensaje

El objetivo de este diagrama es mostrar como un usuario podrá abrir el servicio de chat de texto, enviar un mensaje a través de él y cerrar el servicio.

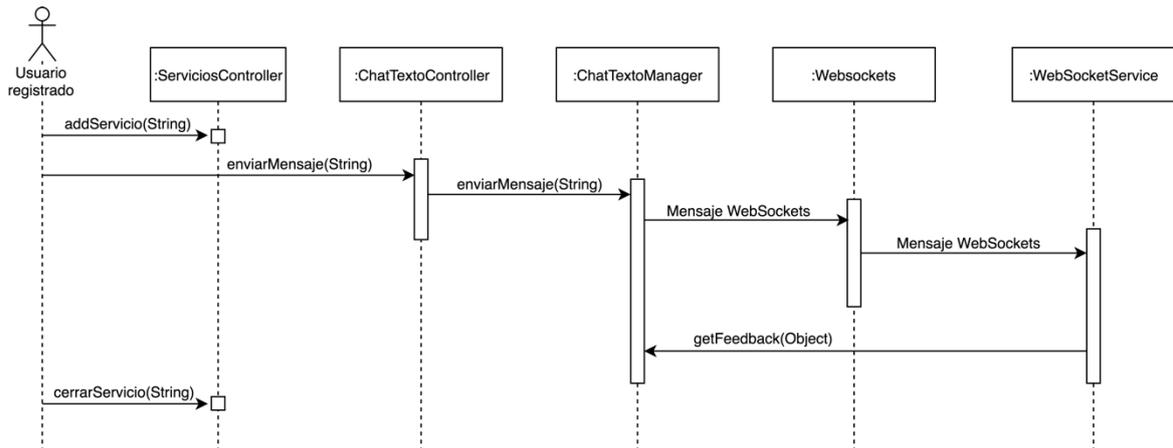


Figura 6.22. Diagrama de Interacción chat de texto

Los diferentes servicios seguirán un esquema muy similar al expuesto anteriormente. Los usuarios se comunicarán con los controladores, los cuales realizan las acciones necesarias contra los managers. Posteriormente, los managers enviarán un mensaje al servidor de WebSockets el cual se encargará de distribuir a todos los usuarios conectados un mensaje a través de WebSocketService. Finalmente, WebSocketService a partir del mensaje recibido realizará la acción que corresponda sobre el manager, el cuál actualizará el servicio.

En el caso del servicio de chat de texto, esto implicará enviar un mensaje de texto desde ChatTextoController hasta WebSockets, el cual distribuirá el mensaje a las instancias de WebSocketService de cada usuario conectado. Para el usuario que envió el mensaje, WebSocketService invocará al método *getFeedback* al cual se le pasará un mensaje que contendrá el número de usuarios que recibieron el mensaje de texto enviado por el usuario.

6.4 Diagramas de Actividades

Se mostrará un diagrama de actividades para la creación de salas debido al número de posibles caminos a tomar durante la creación de una.

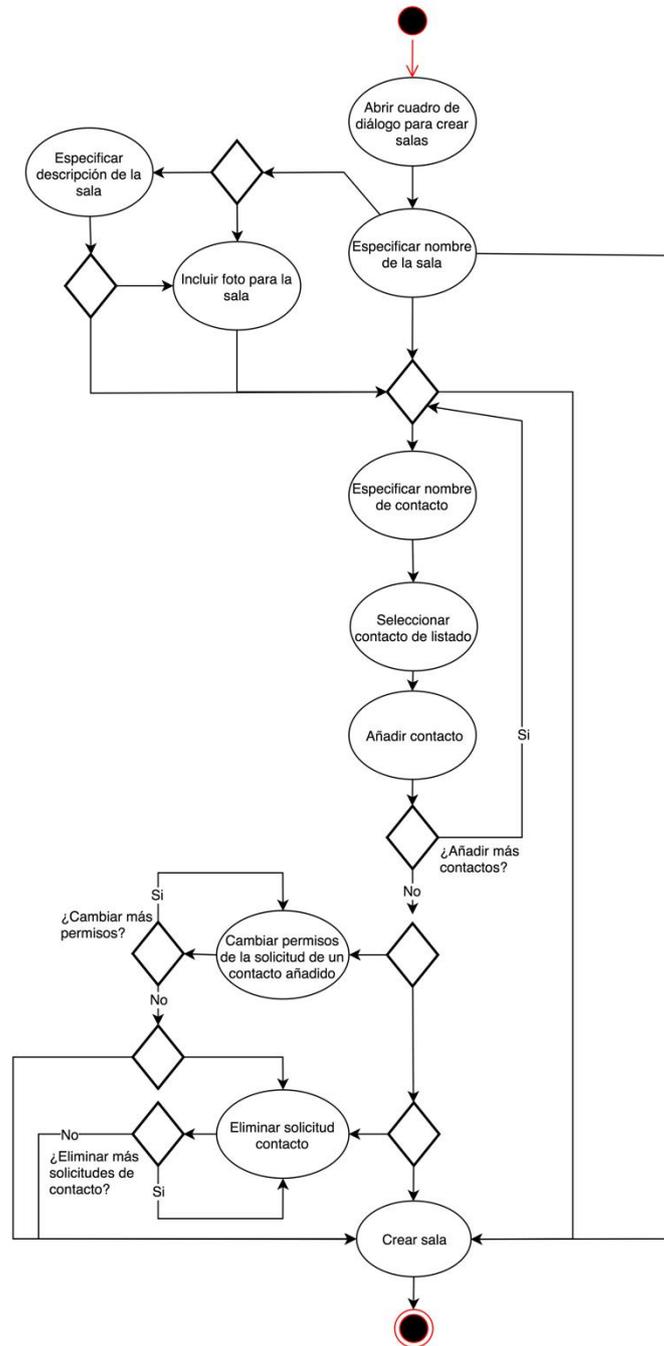


Figura 6.23. Diagrama de Actividad crear sala

Como se puede ver en el diagrama, únicamente el nombre de la sala es obligatorio. Cabe destacar que, si se añaden usuarios para enviarles solicitudes de unión a la sala, solo podremos añadir hasta un máximo de tres contactos.

6.5 Diseño de la Base de Datos

6.5.1 Descripción del SGBD Usado

Como se ha explicado con anterioridad, se ha utilizado la base de datos orientada a grafos Neo4j debido a la facilidad de plasmar los actores y relaciones de este sistema de comunicaciones. Para ello simplemente se utilizarán nodos para representar los usuarios y salas, y relaciones con una única dirección para indicar solicitudes, contactos y permisos sobre salas.

Se han identificado dos tipos diferentes de etiquetas: “Usuario” y “Sala”. Un nodo “Usuario” representa a un usuario registrado y contiene almacenado en sus propiedades el nombre, apellidos, nombre de usuario, email, hash y salt (obtenidos de la contraseña proporcionada) y una dirección URL a la imagen de perfil almacenada en Cloudinary. En el caso de una sala necesitamos un ID para identificar la sala, un nombre, una descripción y, al igual que el usuario, una dirección URL a la imagen de la sala.

En cuanto a relaciones entre nodos encontramos 6 diferentes:

- **Contacto:** Es una relación entre dos nodos “Usuario” e indica que dos usuarios son contactos entre sí. Cypher no permite la existencia de relaciones bidireccionales. Sin embargo, pueden realizarse consultas sin especificar la dirección, por lo que con una relación unidireccional podemos indicar que un usuario es contacto de otro y viceversa.
- **SolicitudContacto:** Es una relación entre dos nodos “Usuario” e indica que un usuario mandó una solicitud de contacto a otro. En este caso sí que nos es útil las relaciones unidireccionales ya que nos ayuda a indicar que usuario mandó la solicitud a quien. Esta relación contiene una propiedad “mensaje” en la que se almacena el mensaje enviado en la solicitud.
- **Bloqueado:** Es una relación entre dos nodos “Usuario” e indica que un usuario bloqueó a otro. Al igual que en la relación “SolicitudContacto” las relaciones unidireccionales son útiles para indicar quien bloqueó a quien.
- **Candidato:** Es una relación entre un nodo “Usuario” y un nodo “Sala” e indica que el usuario es candidato para unirse a la sala, es decir, representa una solicitud de unión a la sala recibida por el usuario. Esta relación contiene una propiedad “permisos” en la que se indica los permisos otorgados al usuario con esa solicitud de unión a la sala.
- **Admin:** Es una relación entre un nodo “Usuario” y un nodo “Sala” e indica que el usuario es el administrador de la sala.

- **Moderador:** Es una relación entre un nodo “Usuario” y un nodo “Sala” e indica que el usuario es el moderador en la sala.
- **Miembro:** Es una relación entre un nodo “Usuario” y un nodo “Sala” e indica que el usuario es el miembro en la sala.
- En el apartado 6.5.3 de esta memoria se muestra un diagrama de la base de datos en las que se puede observar la utilización de los nodos y las relaciones.

6.5.2 Integración del SGBD en Nuestro Sistema

Para conseguir que el sistema se comuniquen con la base de datos Neo4j se utilizan los siguientes elementos:

- **.env:** Archivo utilizado para almacenar variables de entorno. En este archivo se almacena el nombre y contraseña de la base de datos y solo existe si se está ejecutando en local. Heroku y Travis CI proporcionan herramientas para almacenar de forma segura estas variables en sus servidores. En estas últimas plataformas también es necesario almacenar la URL de la base de datos, al contrario que cuando se ejecuta en local ya que se utilizará una base de datos que está ejecutándose en la propia máquina.
- **db:** En este módulo se define un objeto “config” que almacena la URL de la base de datos, su usuario y contraseña.
- **Seraph:** Driver utilizado para poder realizar las conexiones con la base de datos Neo4j.

Como se comentó, se utiliza Heroku y Travis CI para el despliegue de la aplicación en producción y para la ejecución de las pruebas cada vez que la aplicación se despliega en Heroku. Estos diferentes objetivos implican la necesidad de disponer de dos bases de datos en servidores remotos. Por ello se han creado dos bases de datos en GrapheneDB.

6.6 Diseño de la Interfaz

En este apartado se mostrarán las interfaces definitivas de la aplicación ya implementada.

Se han seguido las interfaces definidas en la fase de análisis y se les ha aplicado aquellas mejoras detectadas durante las pruebas de usabilidad.

6.6.1 Interfaz de inicio de sesión y registro

La primera página a la que pueden acceder los usuarios es a la de iniciar sesión y registrarse en la aplicación. Esta página incluye los dos formularios de inicio de sesión y registro de modo que se permite alternar la visualización de una a otra gracias a jQuery. De esta forma no es necesario disponer de dos páginas independientes para proporcionar estos servicios a los usuarios.

The screenshot shows the Copernicus application's login and registration page. At the top, there is a navigation bar with the logo 'COPERNICUS' on the left and links for 'Página Principal', 'Sobre nosotros', and 'Aviso legal' in the center. On the right side of the navigation bar are the flags of Spain and the United Kingdom. Below the navigation bar, the main heading reads 'Bienvenido a Copernicus'. Underneath this heading, there are two tabs: 'Iniciar Sesión' (highlighted in green) and 'Registro'. The 'Iniciar Sesión' tab is active, showing a form with two input fields: 'Nombre de usuario' and 'Contraseña'. Below these fields is a blue button labeled 'Iniciar sesión'.

Figura 6.25. Interfaz de inicio de sesión

COPERNICUS Página Principal Sobre nosotros Aviso legal  

Bienvenido a Copernicus

[Iniciar Sesión](#) **Registro**

Nombre

Apellidos

Nombre de usuario

Correo electrónico

Contraseña

Confirmar contraseña

Selecciona una foto de perfil

Figura 6.26. Interfaz registro

6.6.2 Interfaz de la página principal

En la siguiente imagen se muestra la página de principal de la aplicación. En ella los usuarios pueden visualizar sus listados de contactos y salas, y acceder a través de un menú lateral a distintas vistas y cuadros de diálogo de la aplicación. En el menú lateral también se muestran notificaciones con el número de solicitudes de contacto y de salas que hemos recibido y están pendientes.

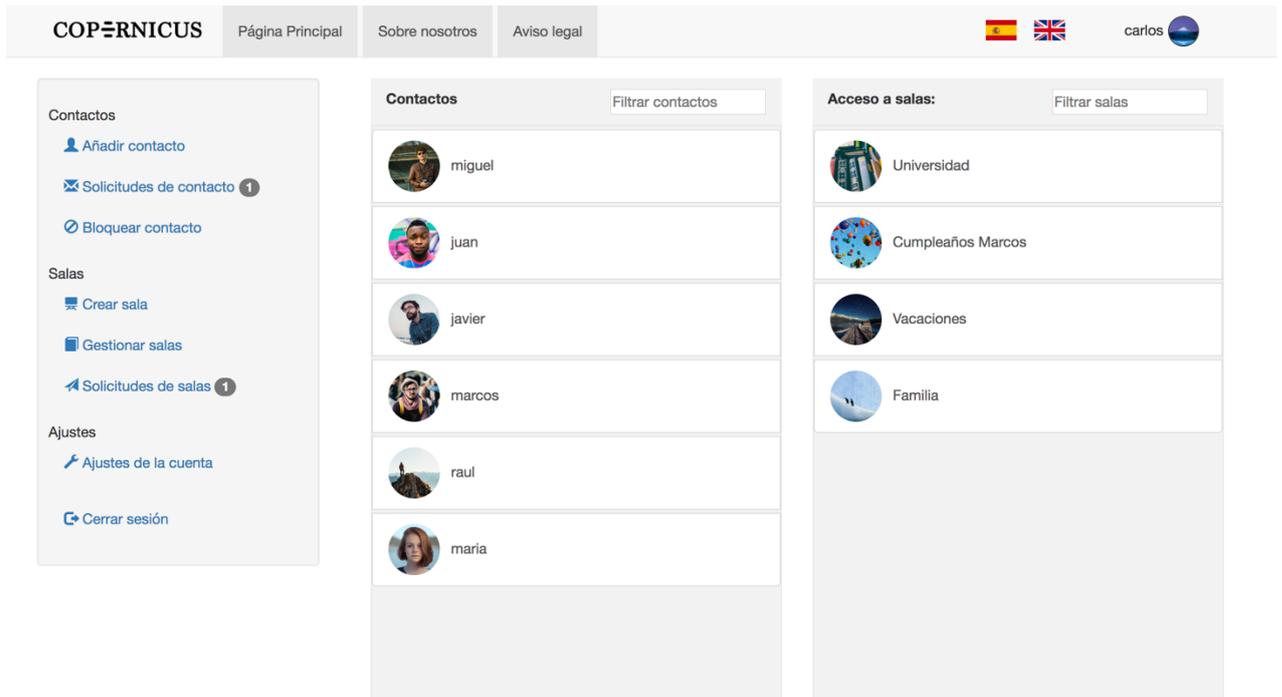


Figura 6.27. Interfaz página principal

Desde el menú principal se puede acceder a varios cuadros de diálogo. Por ello también se mostrará en esta sección todos los cuadros de diálogo a los que se pueden acceder desde esa vista.

El primer cuadro de diálogo que nos encontramos es el mostrado al acceder a “Añadir contacto”. En este diálogo se podrá especificar un usuario y un mensaje para enviar una solicitud de contacto.

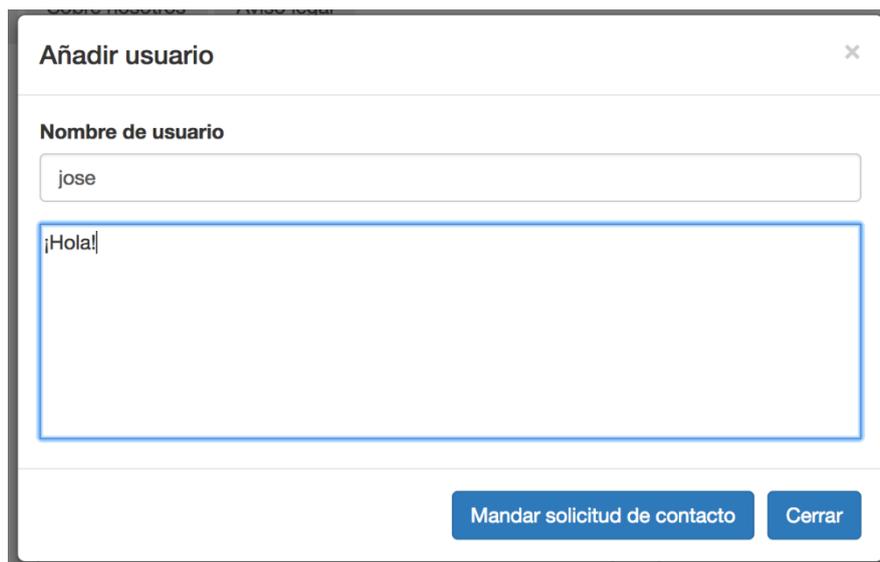


Figura 6.28. Diálogo añadir usuario

Para todos aquellos campos en los que sea necesario especificar un nombre de usuario, se mostrará un listado de usuarios del cual tenemos que seleccionar a la persona que deseemos. A continuación, se muestra el aspecto de este listado.

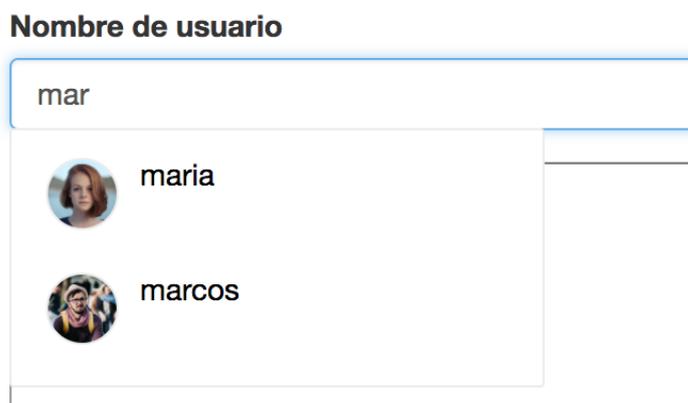


Figura 6.29. Listado de usuarios

El siguiente cuadro de diálogo a mostrar siguiendo el orden del menú lateral, es el de solicitudes de contacto. En este diálogo se podrá aceptar o ignorar las solicitudes de contacto recibidas.

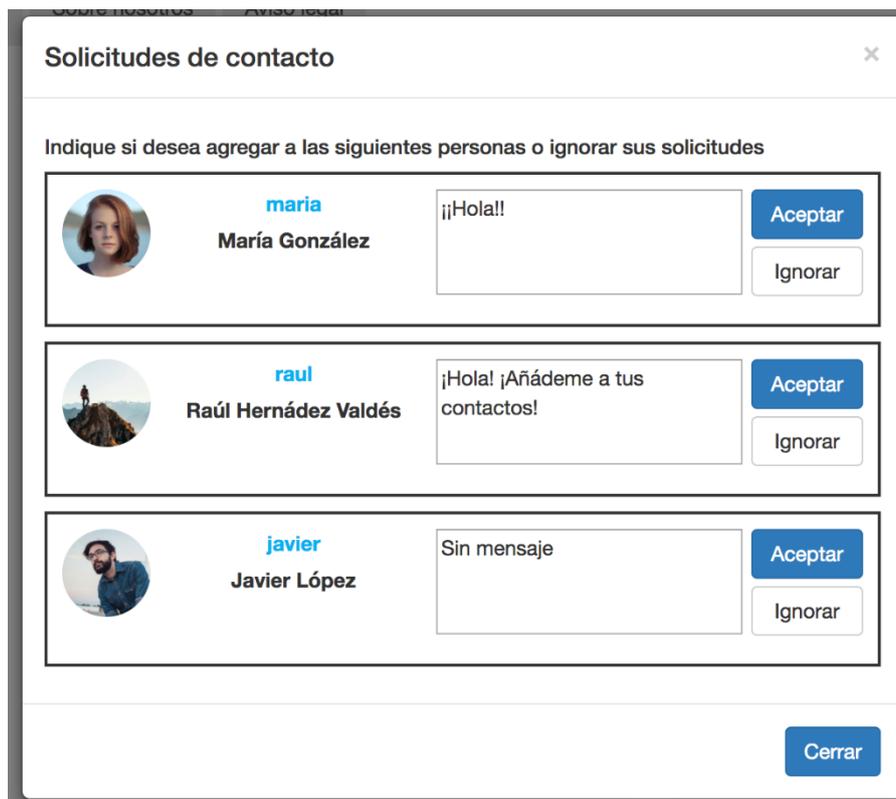


Figura 6.30. Diálogo solicitudes de contacto

Como se puede observar en la foto anterior, se muestra la foto, el nombre de usuario (en azul) y los nombres y apellidos del usuario (solo el nombre si no especificó apellidos). También se puede leer el mensaje recibido junto a la solicitud.

Utilizando el enlace “Bloquear contacto”, podemos acceder a un cuadro de diálogo en el para bloquear y desbloquear usuarios.

Bloquear contacto [X]

Nombre de usuario

Nombre de usuario [input] [Bloquear contacto]

Usuarios bloqueados

Nombre de usuario	Nombre	Apellidos	Desbloquear
miguel	Miguel		[Desbloquear]
javier	Javier	López	[Desbloquear]

[Cerrar]

Figura 6.31. Diálogo bloquear y desbloquear contactos

Para la creación de salas, el usuario se dirigirá al enlace “Crear sala”. Una vez hecho esto, se mostrará el siguiente cuadro de diálogo.

Crear sala
✕

Datos de la sala

Nombre

Descripción

Enviar solicitudes de unión a la sala

Nombre de usuario

Nombre de usuario	Nombre	Apellidos	Permisos	Eliminar
maria	María	González	Moderador ▾	<input type="button" value="Eliminar"/>
raul	Raúl	Hernández Valdés	Miembro ▾	<input type="button" value="Eliminar"/>

Foto de la sala

Seleccionar archivo balloons-2074347_1280.jpg



Figura 6.32. Diálogo crear sala

Al igual que las solicitudes de contacto, para las solicitudes de salas también se dispone de un cuadro diálogo.

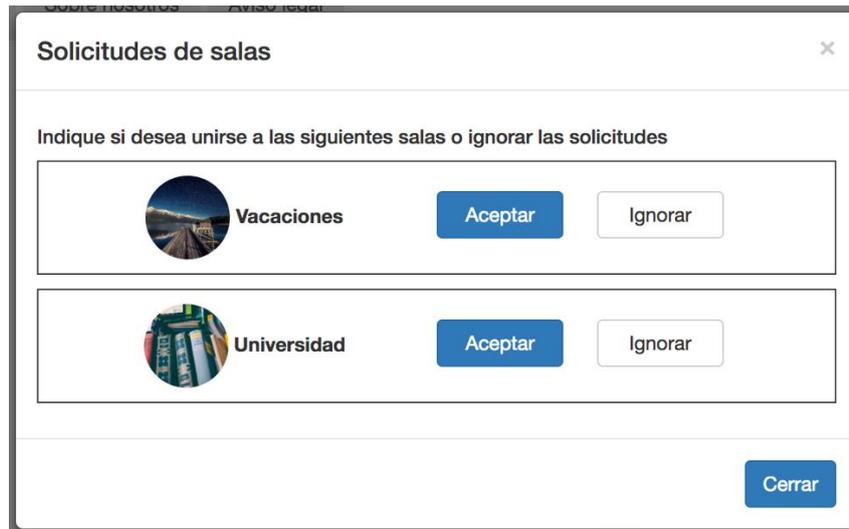


Figura 6.33. Diálogo solicitudes de salas

Por último, desde la página principal también se puede acceder a un cuadro de diálogo para ver la información de nuestros contactos.

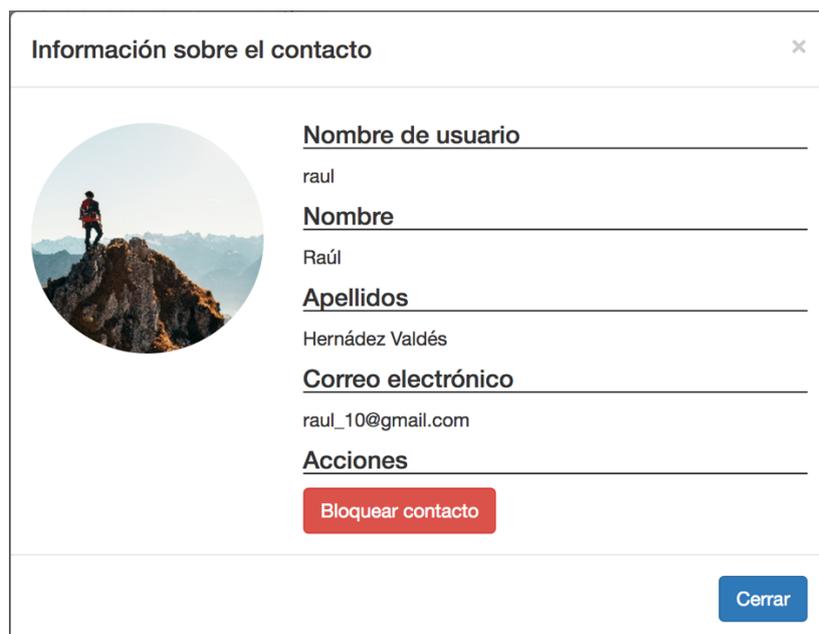


Figura 6.34. Diálogo datos de contacto

Como se puede observar en la captura, desde este último cuadro de diálogo también se da la posibilidad de bloquear al contacto.

6.6.3 Interfaz de las salas

En este apartado se hablará de la interfaz para las salas. Para ello, primero se mostrará la estructura de la sala una vez es accedida y después se enseñará las interfaces de los servicios.

En la siguiente captura se muestra la interfaz de cualquier sala una vez es accedida.

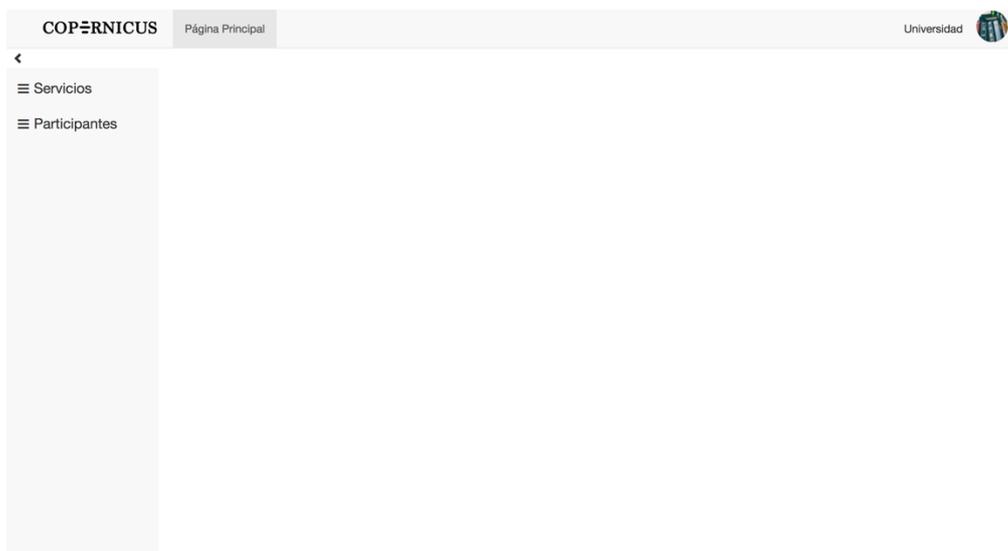


Figura 6.35. Interfaz sala

Como podemos ver, la interfaz consta de un menú lateral, una barra de navegación y una zona en blanco para mostrar, redimensionar y mover los servicios. También existe una flecha encima del menú lateral que permite ocultar el menú y dejar así más espacio para el uso de los servicios.

Si pulsamos en las opciones del menú lateral se muestran otros menús para abrir servicios y para ver los participantes conectados en la sala. A continuación, se pueden ver estos menús.

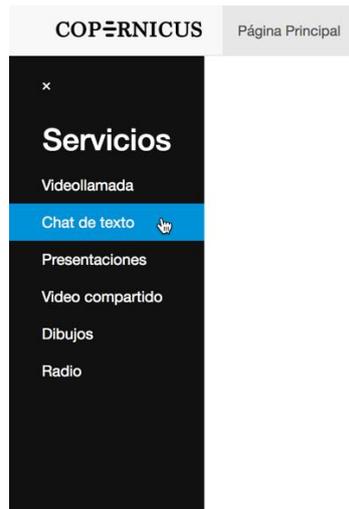


Figura 6.36. Servicios sala

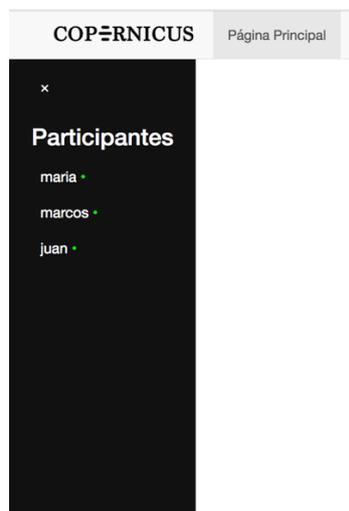


Figura 6.37. Participantes conectados en la sala

Una vez presentada la interfaz de la sala, se hablará de los servicios que dispone.

6.6.3.1 Introducción servicios

Todos los servicios disponen de una zona para moverlos, una “X” para cerrarlos y, exceptuando la radio, pueden ser redimensionados utilizando unas flechas que aparecen al pasar el ratón por la zona inferior derecha.



Figura 6.38. Icono para redimensionar servicios

En los siguientes apartados se estudiarán las interfaces de los 6 servicios disponibles en las salas.

6.6.3.2 Servicio de videollamada

La interfaz del servicio de videollamadas puede mostrar un máximo de 4 videos simultáneos y proporciona tres controles para parar o reanudar el micrófono, altavoz o el video que es transmitido a otros usuarios.

Una vez abierto el servicio de videollamadas, se mostrará el video local en él. En la siguiente imagen se puede ver un ejemplo.

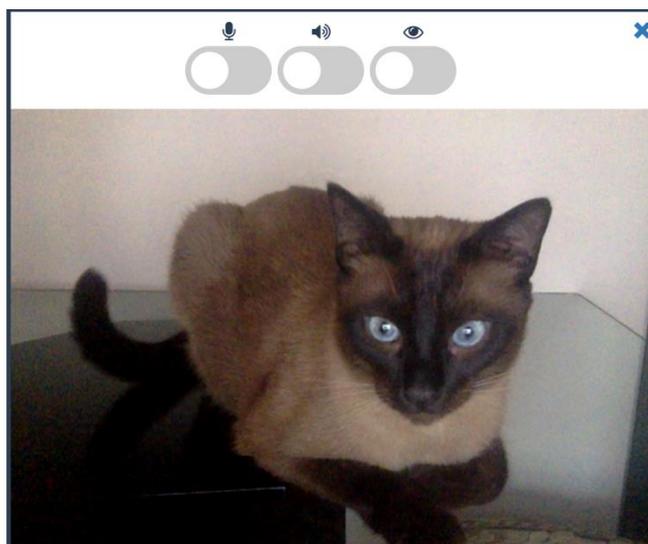


Figura 6.39. Interfaz videollamada

Cuando se conecten nuevos contactos, se mostrarán nuevos videos y se redimensionarán todos para poder verlos correctamente. En la siguiente captura se muestra un ejemplo con tres usuarios conectados.

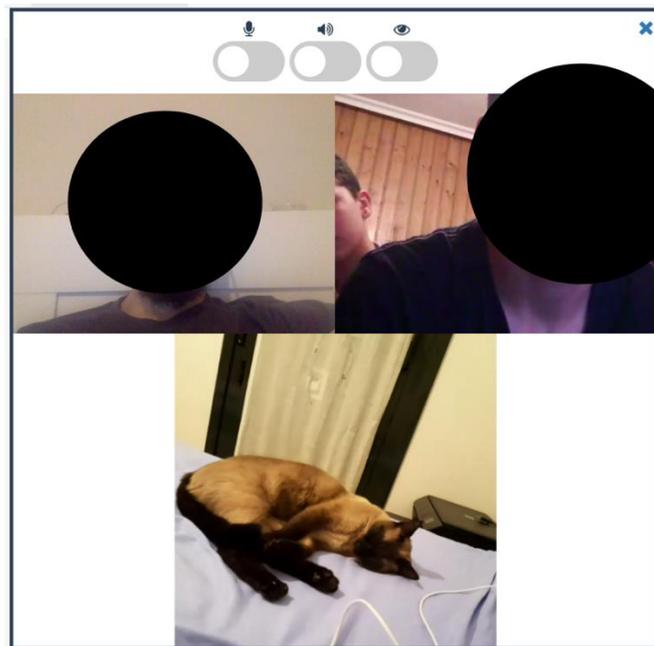


Figura 6.40. Interfaz videollamada con tres usuarios

6.6.3.3 Servicio de chat de texto

El servicio de chat de texto presenta una zona en la que se visualizarán los mensajes, un campo para introducir mensajes y dos botones para mandar mensajes y ficheros.



Figura 6.41. Interfaz chat de texto

6.6.3.4 Servicio de radio

En la interfaz del servicio de radio se muestra un campo de texto para introducir la URL de la emisora o canción que queremos compartir. También dispone de un botón para pausar y reanudar la reproducción, y un regulador para el volumen.

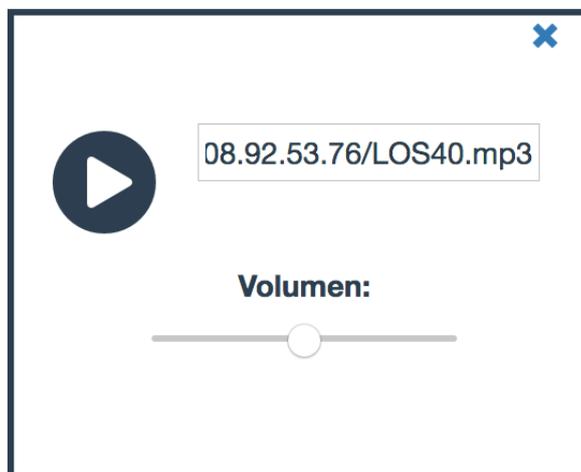


Figura 6.42. Interfaz radio

6.6.3.5 Servicio de presentaciones

La interfaz del servicio de presentaciones contiene una amplia zona para mostrar las presentaciones HTML o PDF, y un botón para modificarlas. Cabe destacar que los anteriores servicios podían ser movidos pulsando en cualquier zona del mismo y desplazando el ratón. Sin embargo, este servicio y los siguientes que se mostrarán facilitan una zona en la parte superior en la que se puede pulsar para moverlo. Esto es así para evitar que esta funcionalidad afecte al correcto funcionamiento de otras zonas del servicio.

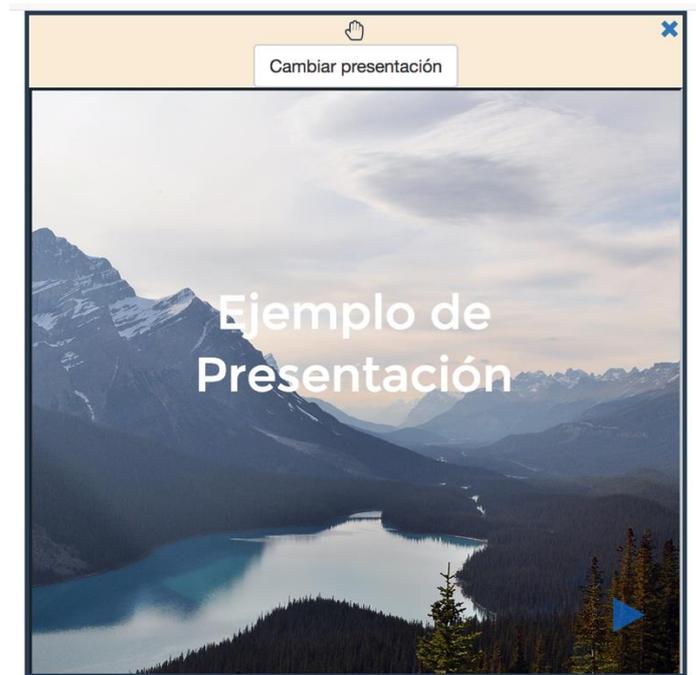


Figura 6.43. Interfaz presentaciones

6.6.3.6 Servicio de video compartido

El servicio de video compartido ofrece una zona para la reproducción de videos y un campo de texto y botón para introducir la URL de un video a reproducir.

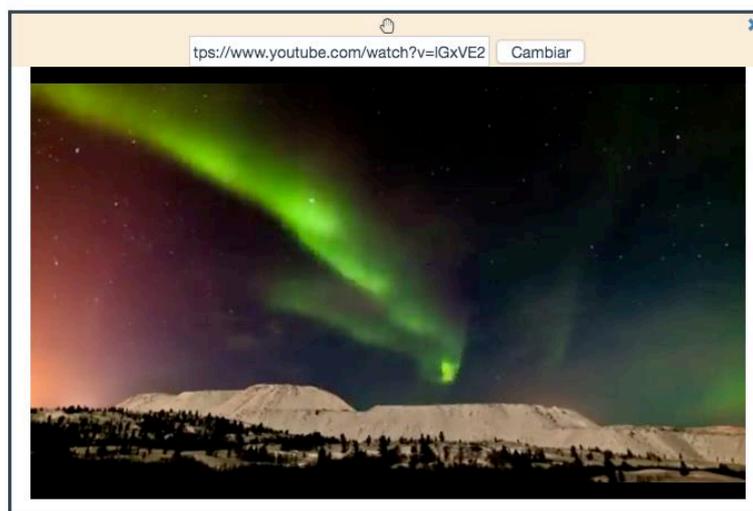


Figura 6.44. Interfaz video compartido

6.6.3.7 Servicio de dibujos

En este servicio podemos dibujar utilizando los botones localizados en la parte superior. Estos botones permiten añadir círculos, cuadrados, triángulos, dibujar con el ratón, mover los dibujos y borrarlo todo.

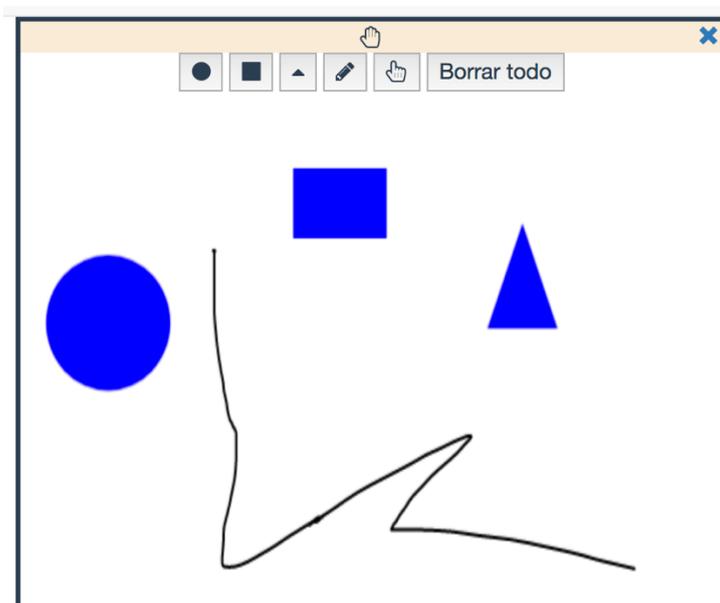


Figura 6.45. Interfaz dibujos

6.6.3.8 Uso conjunto de servicios

Para finalizar con las interfaces de la sala, se quiere mostrar dos ejemplos de cómo se pueden utilizar las salas. En las siguientes capturas se puede apreciar el uso de distintos servicios y distintas distribuciones.

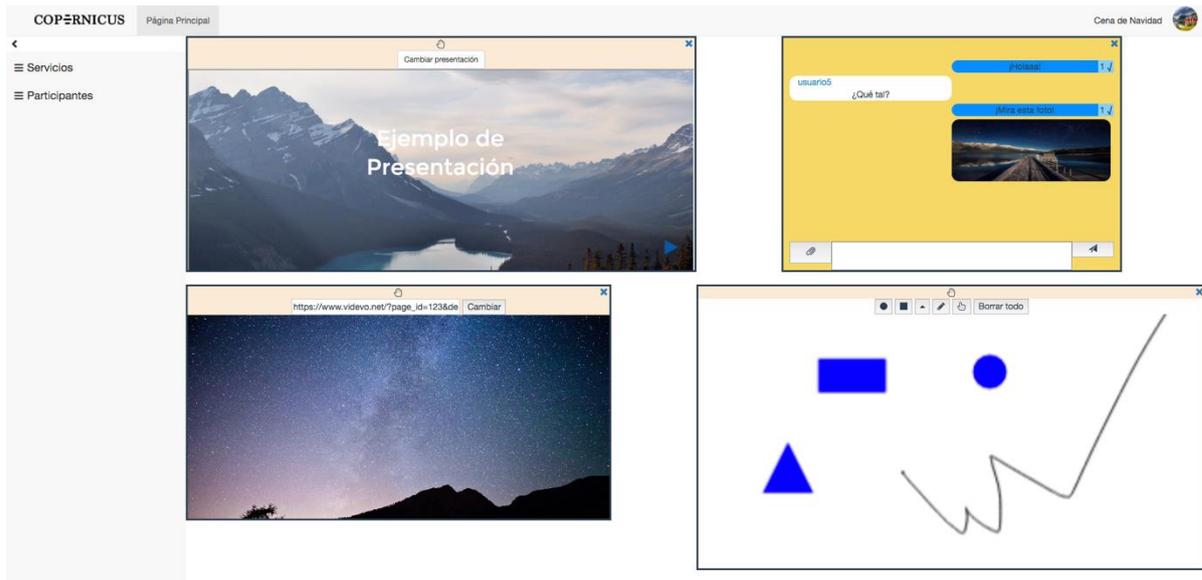


Figura 6.46. Sala con 4 servicios abiertos

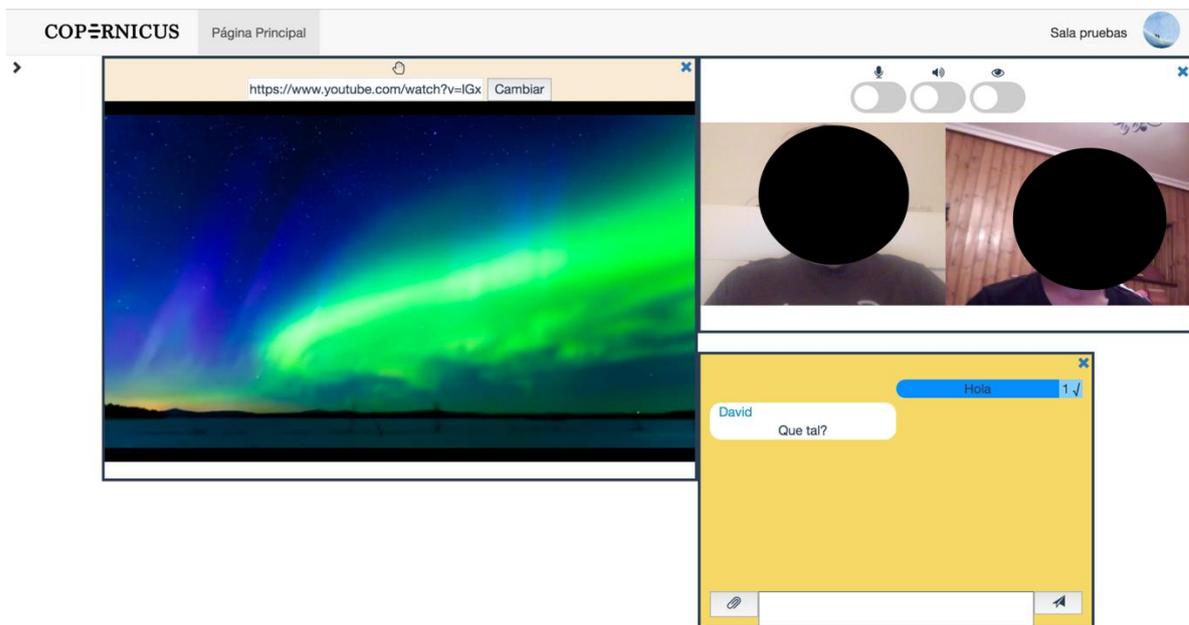


Figura 6.47. Sala con 3 servicios abiertos48

6.6.4 Interfaz para la gestión de salas

En la siguiente captura se muestra la estructura de la interfaz para la gestión de salas. En ella podemos seleccionar la sala que queremos gestionar de los listados de la izquierda, de modo que se muestra su información en la parte de la derecha. Dependiendo de nuestros permisos sobre la sala se muestra unas u otras opciones.

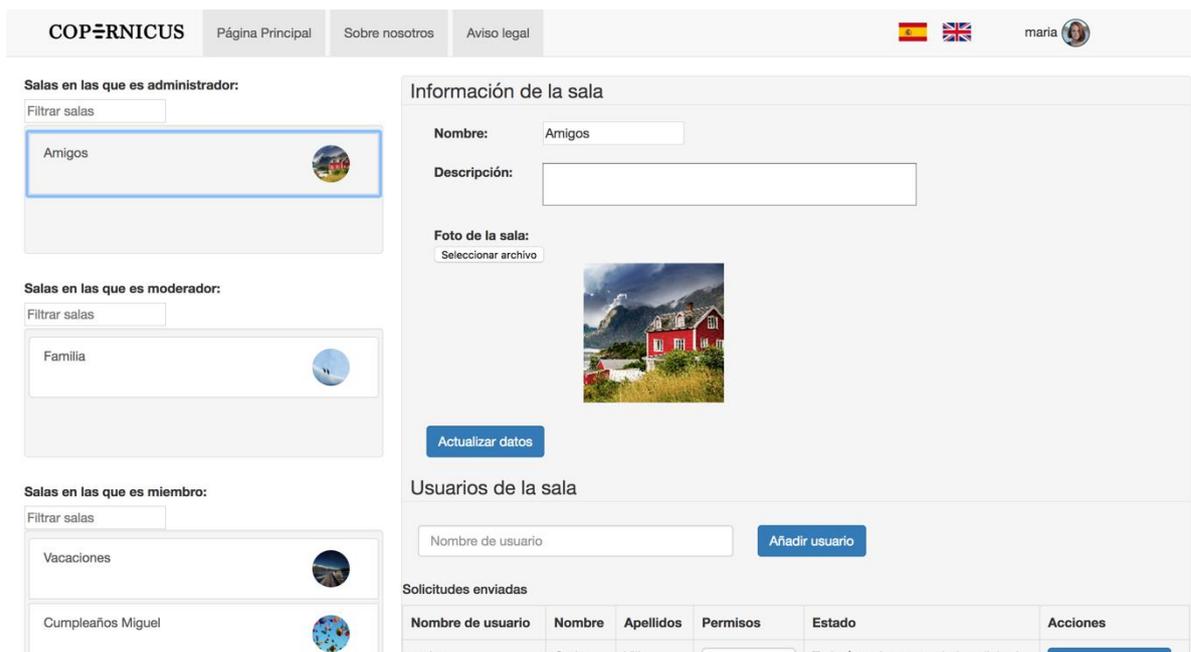


Figura 6.49. Interfaz gestión salas

La zona en la que se muestra la información de la sala está dividida en tres partes: Información de la sala, Usuarios de la sala y Acciones peligrosas. Estas zonas permiten modificar los datos de la sala, gestionar las solicitudes enviadas y participantes y eliminar o salir de la sala.

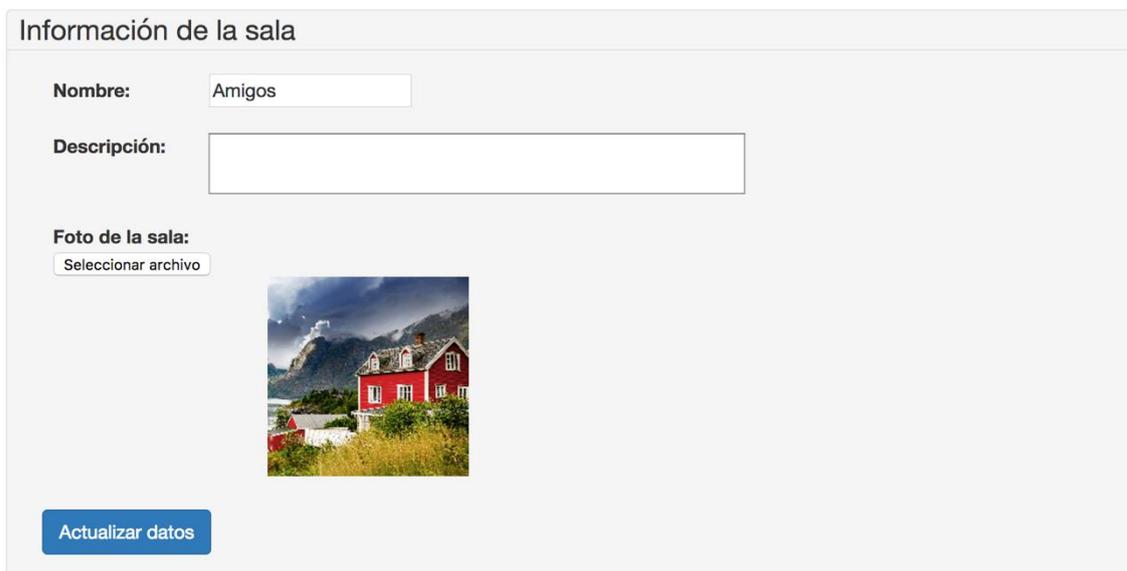


Figura 6.50. Interfaz información de la sala

Usuarios de la sala

Nombre de usuario

Solicitudes enviadas

Nombre de usuario	Nombre	Apellidos	Permisos	Estado	Acciones
carlos	Carlos	Villa	Miembro	Todavía no ha aceptado la solicitud	<input type="button" value="Eliminar solicitud"/>

Participantes

Nombre de usuario	Nombre	Apellidos	Permisos	Acciones
marcos	Marcos	Rodríguez	Moderador	<input type="button" value="Eliminar"/>
maria	María	González	Administrador	<input type="button" value="Eliminar"/>

Figura 6.51. Interfaz usuarios de la sala

Acciones peligrosas

Figura 6.52. Interfaz acciones peligrosas

Estas tres interfaces que se acaban de mostrar son para un usuario administrador, y difieren dependiendo de los permisos del usuario, siendo cada más restrictivas si nuestros permisos son menores. En el caso de un moderador, la interfaz es prácticamente igual salvo que se cambia el botón para eliminar la sala por un botón para salir de ella. También se inhabilita las opciones para eliminar o modificar los permisos de participantes que también sean moderadores. En cuanto a los miembros, estos únicamente ven los usuarios de la sala sin que puedan realizar ninguna modificación.

6.6.5 Interfaz de ajustes de la cuenta

Figura 6.53. Interfaz ajustes de la cuenta

6.6.6 Interfaz con información sobre el creador

Esta es una interfaz simple que tiene el objetivo de mostrar a los usuarios información sobre con quien ponerse en contacto ante cualquier duda o comentario acerca de la aplicación.

Figura 6.54. Interfaz con información del creador

6.6.7 Interfaz aviso legal

Al igual que la anterior interfaz, el único objetivo de esta pantalla es mostrar información. En esta interfaz en concreto se muestra información legal.

COPERNICUS [Página Principal](#) [Sobre nosotros](#) [Aviso legal](#)  

PROTECCIÓN DE DATOS

De conformidad con lo establecido en el Art. 5 de la Ley Orgánica 15/1999 de diciembre de Protección de Datos de Carácter Personal, por el que se regula el derecho de información en la recogida de datos le informamos de los siguientes extremos:

Los datos de carácter personal que nos ha suministrado en esta y otras comunicaciones mantenidas con usted serán objeto de tratamiento en los ficheros responsabilidad de Carlos Villa Blanco.

La finalidad del tratamiento es la de gestionar de forma adecuada la prestación del servicio que nos ha requerido. Asimismo estos datos no serán cedidos a terceros, salvo las cesiones legalmente permitidas.

Su negativa a suministrar los datos obligatorios solicitados implica la imposibilidad prestarle el servicio.

Asimismo, le informamos de la posibilidad de ejercitar los correspondiente derechos de acceso, rectificación, cancelación y oposición de conformidad con lo establecido en la Ley 15/1999 ante Carlos Villa Blanco como responsable del fichero. Los derechos mencionados los puede ejercitar a través de los siguientes medios: carlosvillablanca@gmail.com

Figura 6.55. Interfaz aviso legal

6.6.8 Interfaz de error en la navegación

Esta será la interfaz mostrada en el caso de que se produzca un error en la navegación por la Web.

COPERNICUS

Error 404
Page not found



Figura 6.56. Interfaz error en la navegación

6.7 Especificación Técnica del Plan de Pruebas

La realización de las pruebas será realizada en una maquina con sistema operativo macOS Sierra. En esta máquina se ejecutará el cliente y servidor los cuales se conectan al localhost, y también la base de datos Neo4j.

En el caso de las pruebas unitarias, estas también se llevarán a cabo en el sistema de integración continua Travis CI, de modo que en sus servidores se ejecutará el servidor de la aplicación, y este se conectará con una base de datos alojada en GrapheneDB.

6.7.1 Pruebas Unitarias

Como se ha explicado en la fase de análisis, estas pruebas serán las encargadas de comprobar el perfecto funcionamiento de las diferentes peticiones a la API REST implementada. Se utilizará Mocha, un framework de testing para Node.js que permitirá dividir las pruebas según los diferentes casos de uso.

Las pruebas unitarias serán definidas a partir de aquellas propuestas en el apartado [5.7.6 Pruebas a partir de casos de uso](#). Aun así, también se añadirá alguna prueba adicional.

Hay que tener en cuenta que las pruebas definidas en el análisis incluyen aquellas que solo pueden ser realizadas desde el cliente como, por ejemplo, abrir, cerrar o manipular servicios. Por ello, en este apartado solo se especificarán aquellas que se puedan realizar mediante peticiones a la API REST.

<i>Registrarse</i>	
Prueba	Resultado Esperado
Comprobar que un nombre de usuario no utilizado está disponible.	Se recibe una respuesta HTTP con código 204.
Comprobar que un nombre de usuario utilizado no está disponible.	Se recibe una respuesta HTTP con código 400.
Registrar usuario con nombre y email válidos y nombre de usuario no utilizado.	El sistema posee un usuario más y se recibe un token de sesión. Se recibe una respuesta HTTP con código 200.
Registrar usuario con nombre de usuario ya utilizado.	El sistema no posee un usuario más. Se recibe una respuesta HTTP con código 400.

<i>Iniciar sesión</i>	
Prueba	Resultado Esperado
Iniciar sesión con nombre de usuario y contraseña correctos.	El sistema devuelve un token de sesión. Se recibe una respuesta HTTP con código 200.
Iniciar sesión con nombre de usuario y contraseña incorrectos.	El sistema no devuelve un token de sesión. Se recibe una respuesta HTTP con código 403.

<i>Enviar solicitud de contacto</i>	
Prueba	Resultado Esperado
Un usuario envía una solicitud de contacto a otro usuario.	Se envía la solicitud de contacto al usuario. El otro usuario tiene una solicitud de contacto más. Se recibe una respuesta HTTP con código 204.
Un usuario envía una solicitud de contacto a otro usuario al que ya envió una solicitud de contacto o que ya es contacto.	No se envía la solicitud de contacto al usuario. El otro usuario no tiene una solicitud de contacto más. Se recibe una respuesta HTTP con código 400.

<i>Aceptar solicitud contacto</i>	
Prueba	Resultado Esperado
Un usuario acepta una solicitud de contacto.	El usuario tiene un nuevo contacto y una solicitud de contacto menos. Se recibe una respuesta HTTP con código 204.
Un usuario acepta una solicitud de contacto que ya había aceptado.	El usuario no tiene un nuevo contacto y una solicitud de contacto menos. Se recibe una respuesta HTTP con código 400.

<i>Ignorar solicitud contacto</i>	
Prueba	Resultado Esperado
Un usuario ignora una solicitud de contacto.	El usuario tiene una solicitud de contacto menos. Se devuelve una respuesta HTTP con código 204.
Un usuario ignora una solicitud de contacto que ya había ignorado.	El usuario no tiene una solicitud de contacto menos. Se recibe una respuesta HTTP con código 400.

<i>Crear sala</i>	
Prueba	Resultado Esperado
Un usuario crea una sala sin enviar solicitudes de unión.	Se añade una nueva sala al sistema donde el usuario es el administrador. Se recibe una respuesta HTTP con código 204.
Un usuario crea una sala y especifica solicitudes de unión con los permisos otorgados.	Se añade una nueva sala al sistema donde el usuario es el administrador y los usuarios especificados reciben una solicitud de unión. Se recibe una respuesta HTTP con código 204.

Un usuario crea una sala con un nombre o descripción no válidos.	No se añade una nueva sala al sistema. Se recibe una respuesta HTTP con código 400.
--	---

Enviar solicitud de unión a sala

Prueba	Resultado Esperado
Enviar solicitud de unión a sala.	Existe una solicitud de unión más en la sala. Se recibe una respuesta HTTP con código 204.
Enviar otra solicitud de unión a sala al mismo usuario.	No existe una solicitud de unión más en la sala. Se recibe una respuesta HTTP con código 400.
Enviar solicitud de unión a sala cuando las solicitudes de las salas y los participantes suman 4.	No existe una solicitud de unión más en la sala. Se recibe una respuesta HTTP con código 400

Aceptar solicitud de unión a sala

Prueba	Resultado Esperado
Un usuario acepta una solicitud de unión a sala.	El usuario participa en una nueva sala y tiene una solicitud de unión a salas menos. Se recibe una respuesta HTTP con código 200.
Un usuario acepta una solicitud de unión a sala que ya había aceptado.	El usuario no participa en ninguna sala nueva. Se recibe una respuesta HTTP con código 400.

Bloquear contacto

Prueba	Resultado Esperado
Bloquear un contacto.	El usuario no tiene un usuario bloqueado más. Se recibe una respuesta HTTP con código 204.
Bloquear un contacto que ya está bloqueado.	El usuario no tiene un usuario bloqueado más. Se devuelve una respuesta HTTP con código 400.

Desbloquear contacto

Prueba	Resultado Esperado
Desbloquear un contacto.	El usuario tiene un usuario bloqueado menos. Se recibe una respuesta HTTP con código 204.

Desbloquear un contacto que no está bloqueado.	El usuario no tiene un usuario bloqueado menos. Se recibe una respuesta HTTP con código 400.
--	--

<i>Ignorar solicitud de unión a sala</i>	
Prueba	Resultado Esperado
Un usuario ignora una solicitud de unión a sala.	El usuario no participa en una nueva sala y tiene una solicitud de unión a salas menos. Se recibe una respuesta HTTP con código 204.
Un usuario ignora una solicitud de unión a sala que ya había ignorado.	El usuario no tiene una solicitud de unión a salas menos. Se recibe una respuesta HTTP con código 400.

<i>Comprobar en cuantas salas participan los usuarios</i>	
Prueba	Resultado Esperado
Se comprueba el número de salas en las que participa un usuario.	Se devuelve el número correcto de salas. Se recibe una respuesta HTTP con código 200.
Se comprueba el número de salas en un usuario es administrador.	Se devuelve el número correcto de salas. Se recibe una respuesta HTTP con código 200.
Se comprueba el número de salas en un usuario es moderador.	Se devuelve el número correcto de salas. Se recibe una respuesta HTTP con código 200.
Se comprueba el número de salas en un usuario es miembro.	Se devuelve el número correcto de salas. Se recibe una respuesta HTTP con código 200.

<i>Modificar datos salas</i>	
Prueba	Resultado Esperado
Un usuario administrador modifica los datos de una sala.	Se modifican los datos de la sala. Se recibe una respuesta HTTP con código 204.
Un usuario moderador modifica los datos de una sala.	Se modifican los datos de la sala. Se recibe una respuesta HTTP con código 204.

Un usuario miembro modifica los datos de una sala.	No se modifican los datos de la sala. Se recibe una respuesta HTTP con código 400.
--	--

Eliminar solicitud

Prueba	Resultado Esperado
Un usuario administrador elimina una solicitud de unión a una sala enviada.	Existe una solicitud de unión a la sala menos. Se recibe una respuesta HTTP con código 204.
Un usuario moderador elimina una solicitud de unión a una sala enviada.	Existe una solicitud de unión a la sala menos. Se recibe una respuesta HTTP con código 204.
Un usuario miembro elimina una solicitud de unión a una sala enviada.	No existe una solicitud de unión a la sala menos. Se recibe una respuesta HTTP con código 400.
Un usuario elimina una solicitud de unión a una sala que ya fue aceptada.	No existe una solicitud de unión a la sala menos. Se recibe una respuesta HTTP con código 400.

Cambiar permisos miembros

Prueba	Resultado Esperado
Cambiar permisos de miembro por administrador.	El usuario debería tener una sala más como moderador. Se recibe una respuesta HTTP con código 204.
Cambiar permisos de miembro por moderador.	El usuario debería tener una sala más como moderador. Se recibe una respuesta HTTP con código 204.
Cambiar permisos de miembro por miembro.	El usuario no debería tener una sala más como moderador. Se recibe una respuesta HTTP con código 400.

Cambiar permisos moderadores

Prueba	Resultado Esperado
Cambiar permisos de moderador por administrador.	El usuario debería tener una sala más como miembro. Se recibe una respuesta HTTP con código 204.
Cambiar permisos de moderador por moderador.	El usuario no debería tener una sala más como miembro. Se recibe una respuesta HTTP con código 400.

Cambiar permisos de moderador por miembro.	El usuario no debería tener una sala más como miembro. Se recibe una respuesta HTTP con código 400.
--	---

<i>Eliminar miembro</i>	
Prueba	Resultado Esperado
Eliminar miembro de una sala por administrador.	Existe un participante menos en la sala. Se recibe una respuesta HTTP con código 204.
Eliminar miembro de una sala por moderador.	Existe un participante menos en la sala. Se recibe una respuesta HTTP con código 204.
Eliminar miembro de una sala por miembro.	No existe un participante menos en la sala. Se recibe una respuesta HTTP con código 400.

<i>Eliminar moderadores</i>	
Prueba	Resultado Esperado
Eliminar moderador de una sala por administrador.	Existe un participante menos en la sala. Se recibe una respuesta HTTP con código 204.
Eliminar moderador de una sala por moderador.	No existe un participante menos en la sala. Se recibe una respuesta HTTP con código 400.
Eliminar moderador de una sala por miembro.	No existe un participante menos en la sala. Se recibe una respuesta HTTP con código 400.

<i>Cambiar permisos solicitud</i>	
Prueba	Resultado Esperado
Cambiar permisos de una solicitud de unión a sala por administrado.	Se otorgan otros permisos. Se recibe una respuesta HTTP con código 204.
Cambiar permisos de una solicitud de unión a sala por moderador.	Se otorgan otros permisos. Se recibe una respuesta HTTP con código 204.
Cambiar permisos de una solicitud de unión a sala por miembro.	No se otorgan otros permisos. Se recibe una respuesta HTTP con código 400.

<i>Abandonar sala</i>	
Prueba	Resultado Esperado
Administrador abandona la sala	El usuario no abandona la sala. Se recibe una respuesta HTTP con código 400.
Moderador abandona la sala.	El usuario abandona la sala. Se recibe una respuesta HTTP con código 204.
Miembro abandona la sala.	El usuario abandona la sala. Se recibe una respuesta HTTP con código 204.
Un usuario que ya abandonó una sala la intenta abandonar.	El usuario no abandona. Se recibe una respuesta HTTP con código 400.

<i>Eliminar sala</i>	
Prueba	Resultado Esperado
El administrador elimina la sala.	Los participantes son expulsados y existe una sala menos en el sistema. Se recibe una respuesta HTTP con código 204.
Un moderador elimina la sala.	Los participantes no son expulsados y no existe una sala menos en el sistema. Se recibe una respuesta HTTP con código 400.
Un miembro elimina la sala.	Los participantes no son expulsados y no existe una sala menos en el sistema. Se recibe una respuesta HTTP con código 400.
Eliminar sala que no existe.	No se modifica ningún dato. Se recibe una respuesta HTTP con código 400.

<i>Modificar datos cuenta</i>	
Prueba	Resultado Esperado
Se envía un nombre, apellidos y email válidos.	Se modifican los datos del usuario. Se recibe una respuesta HTTP con código 204.
Se envía un nombre, apellidos o email no válidos.	No se modifican los datos del usuario. Se recibe una respuesta HTTP con código 400.

<i>Modificar contraseña</i>	
Prueba	Resultado Esperado
La contraseña actual es correcta.	Se modifica la contraseña del usuario. Se recibe una respuesta HTTP con código 204.

La contraseña actual es incorrecta.	No se modifica la contraseña del usuario. Se recibe una respuesta HTTP con código 403.
-------------------------------------	--

<i>Eliminar cuenta</i>	
Prueba	Resultado Esperado
Eliminar cuenta de un usuario.	Existe un usuario menos en el sistema. Se recibe una respuesta HTTP con código 204.
Eliminar cuenta de un usuario que no existe.	No se modifica ningún dato. Se recibe una respuesta HTTP con código 400.

Las pruebas unitarias también serán utilizadas para dejar la base de datos con algunos nodos y relaciones, de modo que la aplicación pueda ser probada de forma más sencilla. Por ello, eliminará toda la información de la base de datos al inicio y realizará pruebas repetidas para dejar suficiente información.

6.7.2 Pruebas de Integración y del Sistema

Las pruebas de integración y del sistema consistirán en probar de forma manual el correcto funcionamiento de la aplicación utilizando los navegadores Firefox y Chrome. De esta forma se comprobará que la parte del cliente y del servidor funcionan correctamente de manera conjunta.

Las pruebas incluirán pruebas realizadas en las pruebas unitarias, pero esta vez utilizando un navegador para las comunicaciones con la API REST. Como la interfaz bloqueará muchas acciones no permitidas, las pruebas repetidas no tendrán que ser tan exhaustivas.

Las pruebas mostradas a continuación, son una ampliación y mejora de aquellas que se definieron en la fase de análisis:

<i>Registrarse</i>	
Prueba	Resultado Esperado
Registrar usuario con nombre y email válidos y nombre de usuario no utilizado.	El sistema devuelve un token de sesión y el usuario accede a la página principal de la aplicación.
Registrar usuario con nombre de usuario ya utilizado.	No se permite al usuario registrarse y se muestra un mensaje indicándole que el nombre de usuario está en uso.
Registrar usuario con nombre y email erróneos o	No se registra al usuario en el sistema y se muestra un mensaje notificando el problema.

contraseña y contraseña de confirmación que no coinciden.	
---	--

<i>Iniciar sesión</i>	
Prueba	Resultado Esperado
Iniciar sesión con nombre de usuario y contraseña correctos.	El sistema devuelve un token de sesión y el usuario accede a la página principal de la aplicación.
Iniciar sesión con nombre de usuario y contraseña incorrectos.	Se muestra un mensaje notificando que el usuario o la contraseña son incorrectos.

<i>Cambiar idioma</i>	
Prueba	Resultado Esperado
Se pulsa el botón para cambiar idioma a inglés.	Los textos mostrados en la aplicación están en inglés.
Se pulsa el botón para cambiar idioma a español.	Los textos mostrados en la aplicación están en español.

<i>Modificar datos cuenta</i>	
Prueba	Resultado Esperado
Se envía un nombre, apellidos y email válidos.	Se modifican los datos del usuario y se muestra un mensaje notificándolo.
Se envía un nombre, apellidos o email no válidos.	No se modifican los datos del usuario y se muestra un mensaje notificándolo.

<i>Crear sala</i>	
Prueba	Resultado Esperado
Un usuario crea una sala con un nombre y descripción válido.	Se añade una nueva sala al sistema donde el usuario es el administrador. Se muestra una notificación en la que se indica que la sala fue creada.
Un usuario crea una sala con un nombre y descripción válido, y	Se añade una nueva sala al sistema donde el usuario es el administrador y los participantes especificados son moderadores o miembros dependiendo de los permisos

especifica participantes con sus permisos.	datos. Se muestra una notificación en la que se indica que la sala fue creada.
Un usuario crea una sala con un nombre y descripción no válidos.	No se añade una nueva sala al sistema. Se muestra una notificación en la que se indica cuál fue el problema.

Enviar solicitud de contacto

Prueba	Resultado Esperado
Un usuario envía una solicitud de contacto a otro usuario.	Se envía la solicitud de contacto al usuario. Se muestra una notificación en la que se indica que se envió la solicitud.

Aceptar solicitud contacto

Prueba	Resultado Esperado
Un usuario acepta una solicitud de contacto.	El usuario tiene un nuevo contacto en su listado de contactos. Se muestra una notificación indicando que la solicitud fue aceptada.

Ignorar solicitud contacto

Prueba	Resultado Esperado
Un usuario ignora una solicitud de contacto.	El usuario no tiene un nuevo contacto en su listado de contactos.

Aceptar solicitud de unión a sala

Prueba	Resultado Esperado
Un usuario acepta una solicitud de unión a sala.	El usuario tiene una nueva sala en su listado de salas. Se muestra una notificación de que la solicitud fue aceptada.
Un usuario acepta una solicitud de unión a sala justo después de que fuera eliminada.	El usuario no tiene una nueva sala en su listado de salas. Se muestra una notificación de que hubo un problema al aceptar la solicitud.

Ignorar solicitud de unión a sala

Prueba	Resultado Esperado
Un usuario ignora una solicitud de unión a sala.	El usuario no tiene una nueva sala en su listado de salas. Se muestra un mensaje de que la solicitud fue ignorada.

<u>Acceder a sala</u>	
Prueba	Resultado Esperado
Se pulsa en el botón para acceder a una sala.	Se accede a la página de la sala.
Prueba	Resultado Esperado
Se pulsa en el botón para acceder a una sala en la que nos quitaron el permiso de acceso.	No se accede a la página de la sala. Se muestra una notificación diciendo que el usuario no está autorizado para acceder a la sala.
<u>Filtrar salas</u>	
Prueba	Resultado Esperado
Se escribe un texto en el campo para filtrar sala.	Se muestran aquellas salas cuyos nombres contienen el texto.

<u>Filtrar contactos</u>	
Prueba	Resultado Esperado
Se escribe un texto en el campo para filtrar contactos.	Se muestran aquellos contactos cuyos nombres de usuario contienen el texto.

<u>Bloquear contacto</u>	
Prueba	Resultado Esperado
Bloquear un contacto.	El contacto desaparece del listado de contactos y es expulsado de todas las salas donde el usuario que lo bloqueó es administrador. Se muestra el contacto en un listado de contactos bloqueados.

<u>Desbloquear contacto</u>	
Prueba	Resultado Esperado
Desbloquear un contacto.	El contacto aparece en la lista de contactos y se elimina del listado de contactos bloqueados.

<u>Eliminar cuenta</u>	
Prueba	Resultado Esperado
Eliminar cuenta de un usuario.	Se cierra sesión redirigiendo al usuario a la página de inicio de sesión y registro.
<u>Eliminar sala</u>	
Prueba	Resultado Esperado

Eliminar sala.	Existe una sala menos en el listado de salas y los participantes son expulsados.
----------------	--

Cambiar permisos moderadores

Prueba	Resultado Esperado
Se modifican los permisos de un moderador.	El moderador pasa a tener permisos de miembro. Se muestra una notificación indicando que los permisos fueron modificados.

Eliminar moderadores

Prueba	Resultado Esperado
Eliminar moderador de una sala.	Existe un participante menos en la sala. Se muestra una notificación indicando que el usuario fue eliminado.

Cambiar permisos solicitud

Prueba	Resultado Esperado
Cambiar permisos.	En la solicitud se especifican otros permisos. Se muestra una notificación indicando que los permisos fueron modificados.

Eliminar solicitud

Prueba	Resultado Esperado
Eliminar solicitud de unión a sala.	Existe una solicitud de unión a la sala menos. Se muestra una notificación indicando que la solicitud fue eliminada.

Cambiar permisos miembros

Prueba	Resultado Esperado
Cambiar permisos de miembro.	El usuario pasa a tener permisos de moderador. Se muestra una notificación indicando que los permisos fueron modificados.

Enviar solicitud de unión a sala

Prueba	Resultado Esperado
Enviar solicitud de unión a sala.	Existe una solicitud de unión más en la sala. Se muestra una notificación indicando que la solicitud fue enviada.
Enviar solicitud de unión a sala cuando las solicitudes	No se envía la solicitud y se notifica de ello.

de las salas y los participantes suman 4.	
---	--

<i>Eliminar miembro</i>	
Prueba	Resultado Esperado
Eliminar miembro de una sala.	Existe un participante menos en la sala. Se muestra una notificación indicando que el usuario fue eliminado.

<i>Abandonar sala</i>	
Prueba	Resultado Esperado
Abandonar sala.	Se muestra una ventana emergente para confirmar la acción. Si se acepta no se tendrá acceso a la sala.

<i>Abrir servicio</i>	
Prueba	Resultado Esperado
Abrir servicio.	Aparece el servicio seleccionado en la pantalla.
Abrir servicio que ya fue abierto.	No pasa nada y se muestra un mensaje al usuario.

<i>Cerrar servicio</i>	
Prueba	Resultado Esperado
Cerrar servicio.	El servicio desaparece de la pantalla.

<i>Mover servicio</i>	
Prueba	Resultado Esperado
Mover servicio.	El servicio cambia su posición.

<i>Redimensionar servicio</i>	
Prueba	Resultado Esperado
Redimensionar servicio.	El servicio cambia de tamaño y su contenido se adapta a él.

<i>Crear dibujo colaborativo</i>	
Prueba	Resultado Esperado
Seleccionar botón de una figura.	Aparece la figura en el dibujo. El dibujo aparece para los usuarios que tienen el servicio abierto.
Seleccionar el lápiz y dibujar por el canvas.	Aparece un dibujo con la forma del recorrido del puntero. El dibujo aparece para los usuarios que tienen el servicio abierto.
Utilizar el selector para mover elementos por el canvas.	Los cambios aparecen para los usuarios que tienen el servicio abierto.

<i>Compartir URL video</i>	
Prueba	Resultado Esperado
Se comparte URL de video.	Aparece el video en el servicio de video compartido y en el de los usuarios conectados al servicio. Se muestra una notificación a los otros usuarios indicándoles que el video fue modificado.
Se comparte URL de video no soportado.	No se reproduce ningún video.

<i>Compartir presentación</i>	
Prueba	Resultado Esperado
Se comparte presentación.	Aparece la presentación en la pantalla del servicio de presentaciones y en el de los usuarios conectados al servicio. Se muestra una notificación a los otros usuarios indicándoles que la presentación fue modificada.

<i>Participar en videollamada</i>	
Prueba	Resultado Esperado
Abrir servicio de videollamada.	Aparece en el servicio los videos de los usuarios conectados a la videollamada y el video local.
Usuario pulsa botón de silenciar micrófono.	Los otros usuarios conectados a la videollamada no pueden escuchar el audio del usuario.
Usuario pulsa botón de silenciar altavoces.	El usuario no puede escuchar a los otros usuarios conectados a la videollamada.

Usuario pulsa botón de pausar transmisión de video.	Los otros usuarios conectados a la videollamada no pueden ver el video del usuario.
---	---

<i>Enviar mensaje de texto</i>	
Prueba	Resultado Esperado
El usuario envía mensaje de texto.	Aparece el mensaje en el servicio de chat de texto, y los usuarios conectados al servicio pueden ver el mensaje. Además, el usuario ve cuantos participantes recibieron el mensaje.

<i>Enviar fichero</i>	
Prueba	Resultado Esperado
El usuario envía una imagen.	Aparece la imagen en el servicio de chat de texto y los usuarios conectados al servicio pueden verla. Además, el usuario ve cuantos participantes recibieron el mensaje.
El usuario envía un documento.	Aparece una imagen de un documento en el servicio de chat de texto y los usuarios conectados al servicio la ven. Además, el usuario ve cuantos participantes recibieron el mensaje.

<i>Descargar fichero</i>	
Prueba	Resultado Esperado
El usuario pulsa en el fichero para descargarlo.	Aparece la ventana de diálogo por defecto del navegador que permite descargar el fichero.

6.7.3 Pruebas de Usabilidad y Accesibilidad

En esta sección se definirán aquellos procedimientos y herramientas que se utilizarán para el desarrollo de las pruebas de usabilidad y accesibilidad.

6.7.3.1 Diseño de Cuestionarios

Como he explicado anteriormente, las pruebas de usabilidad serán llevadas a cabo mediante la realización de cuestionarios por personas ajenas al proyecto. Estos cuestionarios serán rellenados con las ideas fruto de su experiencia al utilizar la aplicación desarrollada.

El cuestionario estará formado por las siguientes secciones:

- **1º: Preguntas de carácter general** a través de las cuales se determine el tipo de usuario y su nivel de conocimiento informático.
- **2º: Actividades guiadas** para hacer con nuestra aplicación.
- **3º: Batería de preguntas cortas** con los distintos aspectos de la aplicación que se pretendan evaluar.
- **4º: Observaciones**, para que el usuario aporte todo lo que considere oportuno de nuestra aplicación.

A su vez, mientras los usuarios realizan el cuestionario se realizará una evaluación de sus acciones mediante un cuestionario para el responsable de las pruebas.

Entre los usuarios que realizarán las pruebas se encontrará gente con conocimientos básicos y más avanzados en cuanto al uso de aplicaciones web se refiere. Estas personas además diferirán en cuanto a sus estudios y edades, de modo que se realizarán un estudio de la aplicación desde diferentes puntos de vista.

6.7.3.2 Actividades de las Pruebas de Usabilidad

A continuación, se detallan las secciones del cuestionario que realizarán los usuarios.

6.7.3.2.1 Preguntas de carácter general

¿Usa un ordenador frecuentemente?
<ol style="list-style-type: none"> 1. Todos los días 2. Varias veces a la semana 3. Ocasionalmente 4. Nunca o casi nunca
¿Qué tipo de actividades realiza con el ordenador?
<ol style="list-style-type: none"> 1. Es parte de mi trabajo o profesión 2. Lo uso básicamente para ocio 3. Solo empleo aplicaciones estilo Office 4. Únicamente leo el correo y navego ocasionalmente
¿Ha usado alguna vez software como el de esta prueba?
<ol style="list-style-type: none"> 1. Sí, he empleado software similar 2. No, aunque si empleo otros programas que me ayudan a realizar tareas similares 3. No, nunca
¿Qué busca Vd. Principalmente en un programa?
<ol style="list-style-type: none"> 1. Que sea fácil de usar 2. Que sea intuitivo 3. Que sea rápido 4. Que tenga todas las funciones necesarias

¿Suele registrarse en las páginas web que le parecen interesantes?
<ol style="list-style-type: none">1. Siempre2. Normalmente, aunque en ocasiones busco páginas alternativas si me obligan a registrarme3. Nunca
¿Conoce algún software que le permita realizar dibujos colaborativos con sus contactos?
<ol style="list-style-type: none">1. Sí, y lo he utilizado2. Sí, pero nunca lo he utilizado3. No, pero puede ser de utilidad4. No, no me interesa esa función
¿Le parece interesante que sus contactos compartan enlaces de videos o música y que usted pueda reproducirlos en la misma aplicación?
<ol style="list-style-type: none">1. Si, ya utilizo aplicaciones que me dan esa posibilidad2. Sí, me parece una función interesante3. No, me parece una función innecesaria

6.7.3.2.2 Actividades guiadas

Debido a las características de esta aplicación, será necesario la utilización simultanea de dos navegadores, de este modo se simulará su utilización por dos usuarios. Se pedirá a los participantes que realicen las siguientes actividades:

- Registrarse en la aplicación
- Modificar idioma de la aplicación
- Mandar solicitud de contacto
- Aceptar solicitud de contacto
- Bloquear contacto
- Desbloquear contacto
- Crear una sala enviando una solicitud de unión a un usuario
- Cambiar los permisos de la solicitud de unión enviada al crear la sala
- Eliminar la solicitud de unión enviada
- Enviar una solicitud de unión a la sala después de haberla creado
- Aceptar solicitud de unión a la sala
- Cambiar permisos de un participante de la sala
- Cambiar datos de una sala
- Acceder a una sala
- Ver participantes conectados
- Abrir y cerrar servicios
- Mover y redimensionar servicios

- Participar en una videollamada
- Enviar mensaje de texto y ficheros
- Compartir presentación
- Compartir URL de un video
- Compartir URL de una radio o canción
- Realizar dibujo colaborativo
- Eliminar participantes de una sala
- Eliminar una sala
- Salir de una sala
- Cambiar datos personales
- Cambiar contraseña de la cuenta
- Eliminar cuenta de usuario

6.7.3.2.3 Preguntas Cortas sobre la Aplicación y Observaciones

Facilidad de Uso	Siempre	Frecuentemente	Ocasionalmente	Nunca
<i>¿Sabe dónde está dentro de la aplicación?</i>				
<i>¿La aplicación le informa del resultado de sus acciones?</i>				
<i>¿Sabe cuándo las notificaciones son de errores, informativas o de éxito?</i>				
<i>¿Le resulta sencillo el uso de la aplicación?</i>				
<i>¿Sabe a dónde tiene que dirigirse para realizar las acciones que desea?</i>				
<i>¿Sabe cómo encontrar información de sus contactos?</i>				
<i>¿Sabe si es administrador, moderador o miembro de las salas?</i>				
<i>¿Sabe que usuarios están conectados en una sala?</i>				
Funcionalidad	Siempre	Frecuentemente	Ocasionalmente	Nunca
<i>¿Puede cambiar el idioma de la aplicación?</i>				

<i>¿Puede añadir nuevos contactos?</i>				
<i>¿Puede bloquear y desbloquear contactos?</i>				
<i>¿Sabe cómo acceder a los ajustes de su cuenta?</i>				
<i>¿Sabe crear salas?</i>				
<i>¿Puede añadir usuarios a sus salas?</i>				
<i>¿Puede acceder a las salas?</i>				
<i>¿Identifica los servicios ofrecidos por la aplicación?</i>				
<i>¿Puede abrir y cerrar servicios?</i>				
<i>¿Sabe cómo mover y redimensionar servicios?</i>				
<i>¿Sabe cómo dibujar en el servicio “Dibujos”?</i>				
<i>¿Puede unirse a videollamadas?</i>				
<i>¿Puede enviar mensajes de texto y ficheros?</i>				
<i>¿Conoce cuántos usuarios reciben sus mensajes?</i>				
<i>¿Puede descargar los ficheros recibidos?</i>				
<i>¿Sabe cómo modificar la presentación mostrada en el servicio “Presentaciones”?</i>				
<i>¿Sabe cómo modificar el video mostrado en el servicio “Video compartido”?</i>				
<i>¿Sabe cómo modificar la radio/canción reproducida en el servicio “Radio”?</i>				
<i>¿Sabe cuándo la presentación es</i>				

<i>modificada por otro usuario en el servicio "Presentaciones"?</i>				
<i>¿Sabe cuándo el video es modificado por otro usuario en el servicio "Video compartido"?</i>				
<i>¿Sabe cuándo se modifica el dibujo por otro usuario en el servicio "Dibujos"?</i>				
<i>¿Sabe cuándo la radio/canción es modificada por otro usuario en el servicio "Radio"?</i>				
Calidad del Interfaz				
Aspectos gráficos	Muy Adecuado	Adecuado	Poco Adecuado	Nada Adecuado
<i>El tipo y tamaño de letra es</i>				
<i>Los iconos e imágenes usados son</i>				
<i>Los colores empleados son</i>				
<i>La distribución de los elementos en las vistas es</i>				
<i>Las animaciones son</i>				
<i>La distribución de las salas en servicios que se pueden abrir, cerrar, mover y redimensionar es</i>				
<i>La información mostrada en las tablas es</i>				
<i>Mostrar todas las herramientas para la gestión de todas las salas en una única interfaz es</i>				
Diseño de la Interfaz		Si	No	A veces
<i>¿Le resulta fácil de usar?</i>				
<i>¿El diseño de las pantallas es claro y atractivo?</i>				
<i>¿Cree que la aplicación está bien estructurada?</i>				

¿Cree que es sencillo utilizar los servicios de la sala?			
¿Cree necesario mostrar más notificaciones?			
¿Piensa que es adecuado la utilización de cuadros de diálogos?			
Observaciones			
Cualquier comentario del usuario			

6.7.3.2.4 Cuestionario para el Responsable de las Pruebas

Aspecto Observado	Notas
<i>El usuario comienza a trabajar de forma rápida por las tareas</i>	
<i>Tiempo en realizar cada tarea</i>	
<i>Errores leves cometidos</i>	
<i>Errores graves cometidos</i>	
<i>Preguntas</i>	
<i>Mayor dificultad</i>	

6.7.3.3 Guía de usabilidad

Como parte de las pruebas de usabilidad, también se completará una guía de usabilidad desarrollada por Yusef Hassan Montero.

6.7.3.4 Pruebas de Accesibilidad

Las pruebas de accesibilidad serán llevadas a cabo realizando las siguientes actividades:

- Se utilizará el software “Sim Daltonism” para comprobar que las tonalidades utilizadas en la aplicación son las correctas y no generan ningún problema para las personas con Daltonismo.
- Se utilizará el analizador WCAG 2.0 de TAW.

6.7.4 Pruebas de Rendimiento

Se realizarán dos tipos de pruebas de rendimiento utilizando distintas herramientas. Primero se analizará el tiempo de respuesta de la aplicación utilizando las herramientas proporcionadas por el navegador de Google Chrome, y luego se utilizará el software Apache JMeter para simular la utilización del software por varios usuarios de manera simultánea.

En esta sección se establecerá bajo qué criterios se analizará la aplicación:

- El servidor y navegador se ejecutarán en la misma máquina. Esta es un MacBook Pro con procesador Intel Core i5 2,4 GHz.
- Las fotografías de perfil y de las salas se encuentran en un servidor externo. Este servicio es proporcionado por Cloudinary.
- Se comprobará el tiempo que tarda la aplicación en iniciar sesión y mostrar la página principal. Para ello se usará la cuenta de un usuario que dispone de 7 contactos y de 7 salas, por lo que sumado con la propia foto del usuario requerirá recuperar del servidor externo 15 fotos.
- Se comprobará el tiempo de respuesta de la aplicación a la hora de realizar un registro y mostrar la página principal. Para ello se creará un usuario con una foto de perfil cuyo peso es de 5MB, y que se almacenará en el servidor de Cloudinary.
- Antes de realizar cada prueba se eliminará la cache del navegador.

Capítulo 7. Implementación del Sistema

7.1 Estándares y Normas Seguidos

7.1.1 CSS3

Las hojas de estilos siguen los estándares de W3C. Se ha comprobado con el servicio de validación de CSS del W3C que son documentos CSS versión 3 válidos.



Figura 7.1. CSS validado por W3C

7.1.2 JSON

Para el intercambio de peticiones y respuestas entre la parte del cliente y el servidor, se ha utilizado JavaScript Object Notation (JSON). De esta forma se pueden enviar los objetos JavaScript en formato JSON, y transformarlos de nuevo en objetos JavaScript en el destino.

ECMA International fue la organización que desarrolló el Estándar Standard ECMA-404, en el que se recoge la especificación de JSON.



Figura 7.2. JSON

7.2 Lenguajes

7.2.1 Lenguajes de Programación

7.2.1.1 JavaScript

JavaScript fue el único lenguaje de programación utilizado en este proyecto, tanto para el desarrollo de la parte del servidor como la lógica incluida en el cliente.

JavaScript es un lenguaje de programación interpretado multiparadigma estandarizado por ECMA International mediante ECMAScript.



Figura 7.3. JavaScript

7.2.2 Lenguajes de etiquetas

7.2.2.1 EJS

EJS es un lenguaje de etiquetas que nos permite generar código HTML combinando la sintaxis de HTML y la salida de información procesada mediante código JavaScript. Para ello se utilizan las etiquetas “<%” y “%>” para indicar código JavaScript a ejecutar, y las etiquetas “<%=” “%>” para indicar código JavaScript cuya salida se tiene que incluir en el fichero HTML generado. Se ha utilizado la versión 2.5.6.



Figura 7.4. EJS

7.2.3 Lenguajes de diseño gráfico

7.2.3.1 CSS3

CSS3 es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado. Se ha utilizado la versión 3, de modo que los ficheros generados fueron validados por el servicio de validación de CSS del W3C.



Figura 7.5. CSS3

7.3 Herramientas y Programas Usados para el Desarrollo

En este apartado se explicarán las herramientas y programas utilizados durante el desarrollo de la aplicación.

7.3.1 IntelliJ IDEA Ultimate Edition

IntelliJ IDEA es un entorno de desarrollo integrado desarrollado por JetBrains permite el desarrollo de productos software utilizando diversos lenguajes como JavaScript, Java, HTML, CSS, Scala, Perl, SQL, entre otros.

Existen dos versiones de este programa, la edición *Community* y la edición *Ultimate*. Para el desarrollo de este proyecto se ha utilizado la edición *Ultimate*, en su versión 2017.1.4, ya que está dirigida al desarrollo de aplicaciones Web a diferencia de la edición *Community*. Entre las características que proporciona es el soporte y ayuda para el desarrollo de aplicaciones utilizando JavaScript, Node.js, AngularJS o NPM.



Figura 7.6. IntelliJ IDEA Ultimate

7.3.2 Node.js

Node.js es un entorno de ejecución de JavaScript orientado a eventos asíncronos, multiplataforma y de código abierto. Este permite la ejecución de JavaScript en el lado del servidor y crear aplicaciones altamente escalables. Se ha utilizado la versión 6.11.0.



Figura 7.7. Node.js

7.3.3 AngularJS

AngularJS es un framework MVC para JavaScript de código abierto que proporciona librerías útiles para el desarrollo de aplicaciones web. Con este framework ha sido posible las comunicaciones entre las vistas y los controladores en el cliente, además del uso de directivas o servicios. Se ha utilizado la versión 1.6.4 en este proyecto.



Figura 7.8. AngularJS

7.3.4 Neo4j

Neo4j es uno de los sistemas de gestión de bases de datos orientado a grafos más populares. Con su lenguaje de consultas Cypher, ha permitido la representación de los usuarios, salas y todas sus relaciones de una forma sencilla.

Neo4j además proporciona una interfaz Web muy intuitiva en la que se puede gestionar el estado actual de la base de datos. Se ha utilizado la *Community Edition* en su versión 3.1.1.



Figura 7.9. Neo4j

7.3.5 Express

Express es un framework web rápido, minimalista, abierto y flexible para Node.js, que está construido sobre Connect, un framework extensible de manejo de servidores HTTP. Se ha utilizado la versión 4.14.1.



Figura 7.10. Express

7.3.6 Otras herramientas

En los apartados anteriores se han descrito las herramientas necesarias para el desarrollo de una aplicación *ANNE* (AngularJS – Node.js – Neo4j – Express) como la implementada en este proyecto. Sin embargo, han sido necesarias otras herramientas que han sido explicadas con anterioridad en el capítulo [3. Aspectos Teóricos](#).

7.4 Creación del Sistema

La documentación generada en las primeras etapas de este proyecto fue de gran ayuda para establecer la forma y los límites a la hora de desarrollar la plataforma. Sin embargo, esto no supuso que no se encontraran problemas y dificultades durante el desarrollo.

7.4.1 Problemas Encontrados

A continuación, se detallarán los problemas más destacados a la hora de implementar el sistema.

7.4.1.1 Nuevas tecnologías

Este proyecto supuso una oportunidad de conocer nuevas tecnologías y herramientas que no se impartieron durante la carrera. Por ejemplo, el funcionamiento de los servidores de WebSockets, la API de WebRTC, Node.js, AngularJS, etc. Esto implicó la necesidad de hacer un esfuerzo extra para poder definir de forma correcta esta documentación y desarrollar finalmente la plataforma Copernicus.

7.4.1.2 Utilización de controladores en los servicios

Para que las acciones que realiza un usuario sobre los servicios se lleven a cabo, es necesario la inyección de un controlador en el EJS del servicio para que se comunique con su manager. El problema reside en el hecho de que es necesario añadir el código de estos servicios de manera dinámica a la interfaz, por lo que los servicios se mostraban, pero no funcionaban porque no era posible que se comunicara con el controlador.

El módulo Gridstack.js, utilizado para poder añadir estos elementos y moverlos por la pantalla, no está pensando para añadir dinámicamente código HTML enlazado con controladores. Por lo que la solución fue hacer que el controlador que gestiona los servicios “serviciosController” guardase en variables los EJS de los servicios, de esta forma el controlador puede añadir los EJS a la interfaz y, a continuación, compilar el contenedor utilizando la instrucción `$compile('.grid-stack')($scope)`.

7.4.1.3 Pruebas en Travis CI

Uno de los mayores problemas fue que se necesitaba ejecutar la aplicación en Travis CI para poder pasar los pruebas, y no podía hacer uso de la aplicación en Heroku ya que, además de que modificaría la base de datos en producción, los tiempos de respuesta serían muy grandes cuando el servidor de Heroku estuviese “dormido”. Estaba claro que la solución era ejecutar la aplicación en el propio servidor de Travis CI. El problema de esto

es que ejecutaría la aplicación y se quedaría a la espera de peticiones HTTP, de modo que los tests nunca se ejecutaría.

Finalmente, la solución que, a pesar de ser sencilla, llevó un tiempo encontrarla. Lo que se hizo fue añadir el carácter “&” al comando “node app.js” (node app.js &) en el fichero .travis.yml. Esto hace que la aplicación se ejecute en segundo plano, mientras Travis CI puede continuar con la ejecución de los tests.

7.4.1.4 Salas

La creación de las salas fue uno de los retos más complicados de este proyecto. Desde el aprendizaje del funcionamiento de las comunicaciones con el servidor de WebSockets o la utilización de la API WebRTC para realizar las videollamadas, hasta dar la posibilidad al usuario de añadir elementos dinámicamente a la interfaz.

La participación en las salas tiene muchos puntos que hay que tener en cuenta, desde almacenar la información de la sala que fue accedida, hasta gestionar las conexiones de los usuarios de modo que la información de cierta sala no se filtre en otra.

Fue necesario dedicar mucho tiempo para hallar todas las tecnologías necesarias y, sobre todo, conseguir adaptarlas a las condiciones de este proyecto.

7.4.1.5 Sincronización con servicios no abiertos

Existe un problema con los mensajes del servidor de WebSockets que no se pudo resolver de forma óptima, esto es, la sincronización con aquellos servicios que no están abiertos por el usuario.

El servidor de WebSockets funciona de manera que cuando recibe un mensaje, este lo redirige a todos los usuarios conectados en la sala. De este modo aquellos que no estuviesen conectados no recibirían este mensaje. A esto se le suma la interfaz dinámica en la que podemos abrir y cerrar los servicios. Si un usuario no tiene abierto, por ejemplo, el chat de video y recibe un mensaje con una URL de un video, esta no se podría cambiar ya que no existe ningún reproductor de video en ese momento en la pantalla. Para arreglar caso como este en algunos servicios, se comprueba si estos están abiertos, si no es así se almacenan los cambios en variables. Esto fue realizado en todos los servicios excepto en el de dibujos, que es necesario que las dos personas tengan el servicio abierto. Se decidió esto debido a los periodos de tiempo para terminar la aplicación y a la existencia de otras prioridades.

7.4.1.6 *Utilización de bases de datos diferentes para producción y testing*

Fue necesario dedicar bastante tiempo para crear el entorno de producción y testing.

El objetivo era conseguir que la aplicación utilizara diferentes bases de datos dependiendo de si está ejecutando en local, en Travis CI o en Heroku. Tras un tiempo buscando posibles soluciones se llegó a la conclusión de que la mejor forma para hacer esto es utilizando variables de entorno diferentes dependiendo de donde se encuentre la aplicación. De este modo si la aplicación se está ejecutando en Travis CI, esta obtendrá una URL para una base de datos diferente que ejecutándose en Heroku o en local.

7.4.1.7 *Servicio de presentaciones*

El objetivo principal del servicio de presentaciones era dar la posibilidad de añadir presentaciones en HTML que utilizaran el framework reveal.js incluido como una dependencia de la aplicación. Esto no se consiguió realizar y la única solución fue utilizar unas presentaciones con reveal.js incrustado. Fue un contratiempo al que se dedicó un tiempo para intentar solucionarlo e incluso para buscar alguna alternativa. Pero, finalmente, no se encontró ninguna solución mejor.

7.4.1.8 *Servicio de video compartido*

Para la realización de este módulo ya se tenía en mente una tecnología que permitiría la reproducción de videos de los que tuviéramos su URL, y de videos alojados en YouTube. Sin embargo, durante su utilización no se consiguió reproducir videos de YouTube. Tras varios intentos para solucionar el problema, se tuvo que buscar una solución adicional. Esta solución fue la utilización de 2 reproductores diferentes, uno de ellos dedicado únicamente a la reproducción de video de YouTube. De este modo, dependiendo de si la URL es de YouTube o no se utilizará un reproductor u otro.

7.4.1.9 *Descargar ficheros en Chrome*

Durante las pruebas utilizando navegadores Chrome, se ha detectado que algunos ficheros no pueden descargarse utilizando este navegador. Sin embargo, las pruebas realizadas con Firefox permitieron descargar todos los ficheros sin ningún problema. Este problema fue estudiado durante cierto tiempo, pero sin resultados positivos. La falta de tiempo y otras prioridades de la aplicación acabaron en dejarlo como un problema a solucionar en futuras actualizaciones.

7.4.1.10 Búsqueda y adaptación de módulos de terceros

Para la realización de este proyecto se utilizaron módulos creados por terceras personas, ya que no se buscaba reinventar la rueda para, por ejemplo, proporcionar campos con autocompletado o mostrar notificaciones.

Esto conllevó dos problemas, la necesidad de realizar estudios para evaluar las posibles necesidades del proyecto y las herramientas que las harían funcionar, pero también tener que adaptar estas últimas a los objetivos de la aplicación. Uno de los casos más notables es Gridstack.js y la dificultad para añadir dinámicamente código HTML enlazado con controladores, o el uso de jQuery para hacer funcionar correctamente los campos con autocompletado.

7.4.1.11 Videollamadas

Para probar la función de videollamada se utilizaba principalmente un ordenador (MacBook Pro) con dos navegadores, y en ocasiones este ordenador con otro de sobremesa (al que se tenía un acceso limitado). Todas las pruebas realizadas eran óptimas, pero, cuando ya se había finalizado el proyecto, se realizó una prueba con dos navegadores en un ordenador “Acer”. Este ordenador no permitía simular que dos personas se conectaban a la videollamada porque al estar usando un navegador la cámara, se bloqueaba para el otro navegador. Con esto se quiere aclarar que la prueba de videollamada con la aplicación en local puede o no realizarse dependiendo del ordenador y sistema operativo.

Para poder realizar pruebas con otras máquinas, se puede utilizar la aplicación desplegada en la página <https://copernicus-platform.herokuapp.com>. Pero hay que tener en cuenta que se utilizó un servidor gratuito, por lo que la aplicación podría no responder de forma óptima. Esto no debería afectar a las videollamadas, si utilizamos una conexión a internet decente, ya que son punto a punto.

En el caso de utilizar la aplicación desplegada en Heroku, también hay que tener en cuenta que hubo problemas utilizando la conexión a internet de la Universidad de Oviedo. Esta conexión impide el correcto funcionamiento de las videollamadas. Una forma de solucionar este problema podría ser utilizando un servidor “TURN”, el problema es que sería necesario pagar por este servicio.

7.4.1.12 Límite de las salas

La idea original era poner un límite de 8 personas a las salas, y no 4 como si hizo finalmente. Esto fue debido, principalmente, a la imposibilidad de probar las videollamadas con más de 4 personas en las salas. Las pruebas en local permitían realizar videollamadas utilizando como máximo 4 ventanas de navegadores. A partir de ahí la

máquina no podía soportarlo y se ralentizaba tanto hasta el punto de bloquearse en algunas ocasiones.

Por ello, como se comentará más adelante, una posible ampliación será aumentar el número de participantes en las salas cuando se disponga de los recursos necesarios.

7.4.1.13 Depurar la aplicación

Al estar utilizando Node.js y no tener (o conocer) un *debugger* con un funcionamiento similar al que se puede encontrar en el IDE Eclipse utilizado durante toda la carrera, implicó que el proceso de encontrar los motivos de ciertos problemas y resolverlo fuese bastante tedioso y complicado en ciertas ocasiones.

Una forma de facilitar esta labor fue utilizando un *logger* para Node.js llamado Morgan. Esta herramienta permitió ver las peticiones recibidas por el servidor y el código HTTP de la respuesta.

7.4.2 Descripción Detallada de las Clases

La documentación de este proyecto fue realizada con la herramienta ngdocs. El objetivo era adjuntar la documentación en formato HTML junto con los ficheros entregados, pero no ha sido posible porque se han encontrado problemas con su visualización.

Por ello, se ha creado un repositorio en GitHub encargado de almacenar la documentación del proyecto. Gracias a GitHub Pages, la documentación puede ser accedida a través del siguiente enlace: <https://carlvilla.github.io/Copernicus-doc>.

Capítulo 8. Desarrollo de las Pruebas

8.1 Pruebas Unitarias

En esta sección se muestran los resultados de las pruebas definidas en el diseño.

<i>Registrarse</i>		
Prueba	Resultado Esperado	Superada
Comprobar que un nombre de usuario no utilizado está disponible.	Se recibe una respuesta HTTP con código 204.	Sí
Comprobar que un nombre de usuario utilizado no está disponible.	Se recibe una respuesta HTTP con código 400.	Sí
Registrar usuario con nombre y email válidos y nombre de usuario no utilizado.	El sistema posee un usuario más y se recibe un token de sesión. Se recibe una respuesta HTTP con código 200.	Sí
Registrar usuario con nombre de usuario ya utilizado	El sistema no posee un usuario más. Se recibe una respuesta HTTP con código 400.	Sí

<i>Iniciar sesión</i>		
Prueba	Resultado Esperado	Superada
Iniciar sesión con nombre de usuario y contraseña correctos.	El sistema devuelve un token de sesión. Se recibe una respuesta HTTP con código 200.	Sí
Iniciar sesión con nombre de usuario y contraseña incorrectos.	El sistema no devuelve un token de sesión. Se recibe una respuesta HTTP con código 403.	Sí

<i>Enviar solicitud de contacto</i>		
Prueba	Resultado Esperado	Superada
Un usuario envía una solicitud de contacto a otro usuario.	Se envía la solicitud de contacto al usuario. El otro usuario tiene una solicitud de contacto más. Se recibe una respuesta HTTP con código 204.	Sí

Un usuario envía una solicitud de contacto a otro usuario al que ya envió una solicitud de contacto o que ya es contacto.	No se envía la solicitud de contacto al usuario. El otro usuario no tiene una solicitud de contacto más. Se recibe una respuesta HTTP con código 400.	Sí
---	---	----

Aceptar solicitud contacto

Prueba	Resultado Esperado	Superada
Un usuario acepta una solicitud de contacto.	El usuario tiene un nuevo contacto y una solicitud de contacto menos. Se recibe una respuesta HTTP con código 204.	Sí
Un usuario acepta una solicitud de contacto que ya había aceptado.	El usuario no tiene un nuevo contacto y una solicitud de contacto menos. Se recibe una respuesta HTTP con código 400.	Sí

Ignorar solicitud contacto

Prueba	Resultado Esperado	Superada
Un usuario ignora una solicitud de contacto.	El usuario tiene una solicitud de contacto menos. Se devuelve una respuesta HTTP con código 204.	Sí
Un usuario ignora una solicitud de contacto que ya había ignorado.	El usuario no tiene una solicitud de contacto menos. Se recibe una respuesta HTTP con código 400.	Sí

Crear sala

Prueba	Resultado Esperado	Superada
Un usuario crea una sala sin enviar solicitudes de unión.	Se añade una nueva sala al sistema donde el usuario es el administrador. Se recibe una respuesta HTTP con código 204.	Sí
Un usuario crea una sala y especifica solicitudes de unión con los permisos otorgados.	Se añade una nueva sala al sistema donde el usuario es el administrador y los usuarios especificados reciben una solicitud de unión. Se recibe una respuesta HTTP con código 204.	Sí

Un usuario crea una sala con un nombre o descripción no válidos.	No se añade una nueva sala al sistema. Se recibe una respuesta HTTP con código 400.	Sí
--	---	----

<i>Enviar solicitud de unión a sala</i>		
Prueba	Resultado Esperado	Superada
Enviar solicitud de unión a sala.	Existe una solicitud de unión más en la sala. Se recibe una respuesta HTTP con código 204.	Sí
Enviar otra solicitud de unión a sala al mismo usuario.	No existe una solicitud de unión más en la sala. Se recibe una respuesta HTTP con código 400.	Sí
Enviar solicitud de unión a sala cuando las solicitudes de las salas y los participantes suman 4.	No existe una solicitud de unión más en la sala. Se recibe una respuesta HTTP con código 400	Sí

<i>Aceptar solicitud de unión a sala</i>		
Prueba	Resultado Esperado	Superada
Un usuario acepta una solicitud de unión a sala.	El usuario participa en una nueva sala y tiene una solicitud de unión a salas menos. Se recibe una respuesta HTTP con código 200.	Sí
Un usuario acepta una solicitud de unión a sala que ya había aceptado.	El usuario no participa en ninguna sala nueva. Se recibe una respuesta HTTP con código 400.	Sí

<i>Bloquear contacto</i>		
Prueba	Resultado Esperado	Superada
Bloquear un contacto.	El usuario no tiene un usuario bloqueado más. Se recibe una respuesta HTTP con código 204.	Sí
Bloquear un contacto que ya está bloqueado.	El usuario no tiene un usuario bloqueado más. Se devuelve una respuesta HTTP con código 400.	Sí

<i>Desbloquear contacto</i>		
Prueba	Resultado Esperado	Superada
Desbloquear un contacto.	El usuario tiene un usuario bloqueado menos. Se recibe una respuesta HTTP con código 204.	Sí
Desbloquear un contacto que no está bloqueado.	El usuario no tiene un usuario bloqueado menos. Se recibe una respuesta HTTP con código 400.	Sí

<i>Ignorar solicitud de unión a sala</i>		
Prueba	Resultado Esperado	Superada
Un usuario ignora una solicitud de unión a sala.	El usuario no participa en una nueva sala y tiene una solicitud de unión a salas menos. Se recibe una respuesta HTTP con código 204.	Sí
Un usuario ignora una solicitud de unión a sala que ya había ignorado.	El usuario no tiene una solicitud de unión a salas menos. Se recibe una respuesta HTTP con código 400.	Sí

<i>Comprobar en cuantas salas participan los usuarios</i>		
Prueba	Resultado Esperado	Superada
Se comprueba el número de salas en las que participa un usuario.	Se devuelve el número correcto de salas. Se recibe una respuesta HTTP con código 200.	Sí
Se comprueba el número de salas en un usuario es administrador.	Se devuelve el número correcto de salas. Se recibe una respuesta HTTP con código 200.	Sí
Se comprueba el número de salas en un usuario es moderador.	Se devuelve el número correcto de salas. Se recibe una respuesta HTTP con código 200.	Sí
Se comprueba el número de salas en un usuario es miembro.	Se devuelve el número correcto de salas. Se recibe una respuesta HTTP con código 200.	Sí

<i>Modificar datos salas</i>		
Prueba	Resultado Esperado	Superada
Un usuario administrador modifica los datos de una sala.	Se modifican los datos de la sala. Se recibe una respuesta HTTP con código 204.	Sí
Un usuario moderador modifica los datos de una sala.	Se modifican los datos de la sala. Se recibe una respuesta HTTP con código 204.	Sí
Un usuario miembro modifica los datos de una sala.	No se modifican los datos de la sala. Se recibe una respuesta HTTP con código 400.	Sí

<i>Eliminar solicitud</i>		
Prueba	Resultado Esperado	Superada
Un usuario administrador elimina una solicitud de unión a una sala enviada.	Existe una solicitud de unión a la sala menos. Se recibe una respuesta HTTP con código 204.	Sí
Un usuario moderador elimina una solicitud de unión a una sala enviada.	Existe una solicitud de unión a la sala menos. Se recibe una respuesta HTTP con código 204.	Sí
Un usuario miembro elimina una solicitud de unión a una sala enviada.	No existe una solicitud de unión a la sala menos. Se recibe una respuesta HTTP con código 400.	Sí
Un usuario elimina una solicitud de unión a una sala que ya fue aceptada.	No existe una solicitud de unión a la sala menos. Se recibe una respuesta HTTP con código 400.	Sí

<i>Cambiar permisos miembros</i>		
Prueba	Resultado Esperado	Superada
Cambiar permisos de miembro por administrador.	El usuario debería tener una sala más como moderador. Se recibe una respuesta HTTP con código 204.	Sí
Cambiar permisos de miembro por moderador.	El usuario debería tener una sala más como moderador. Se recibe una respuesta HTTP con código 204.	Sí

Cambiar permisos de miembro por miembro.	El usuario no debería tener una sala más como moderador. Se recibe una respuesta HTTP con código 400.	Sí
--	---	----

Cambiar permisos moderadores

Prueba	Resultado Esperado	Superada
Cambiar permisos de moderador por administrador.	El usuario debería tener una sala más como miembro. Se recibe una respuesta HTTP con código 204.	Sí
Cambiar permisos de moderador por moderador.	El usuario no debería tener una sala más como miembro. Se recibe una respuesta HTTP con código 400.	Sí
Cambiar permisos de moderador por miembro.	El usuario no debería tener una sala más como miembro. Se recibe una respuesta HTTP con código 400.	Sí

Eliminar miembro

Prueba	Resultado Esperado	Superada
Eliminar miembro de una sala por administrador.	Existe un participante menos en la sala. Se recibe una respuesta HTTP con código 204.	Sí
Eliminar miembro de una sala por moderador.	Existe un participante menos en la sala. Se recibe una respuesta HTTP con código 204.	Sí
Eliminar miembro de una sala por miembro.	No existe un participante menos en la sala. Se recibe una respuesta HTTP con código 400.	Sí

Eliminar moderadores

Prueba	Resultado Esperado	Superada
Eliminar moderador de una sala por administrador.	Existe un participante menos en la sala. Se recibe una respuesta HTTP con código 204.	Sí
Eliminar moderador de una sala por moderador.	No existe un participante menos en la sala. Se recibe una respuesta HTTP con código 400.	Sí

Eliminar moderador de una sala por miembro.	No existe un participante menos en la sala. Se recibe una respuesta HTTP con código 400.	Sí
---	--	----

Cambiar permisos solicitud

Prueba	Resultado Esperado	Superada
Cambiar permisos de una solicitud de unión a sala por administrado.	Se otorgan otros permisos. Se recibe una respuesta HTTP con código 204.	Sí
Cambiar permisos de una solicitud de unión a sala por moderador.	Se otorgan otros permisos. Se recibe una respuesta HTTP con código 204.	Sí
Cambiar permisos de una solicitud de unión a sala por miembro.	No se otorgan otros permisos. Se recibe una respuesta HTTP con código 400.	Sí

Abandonar sala

Prueba	Resultado Esperado	Superada
Administrador abandona la sala	El usuario no abandona la sala. Se recibe una respuesta HTTP con código 400.	Sí
Moderador abandona la sala.	El usuario abandona la sala. Se recibe una respuesta HTTP con código 204.	Sí
Miembro abandona la sala.	El usuario abandona la sala. Se recibe una respuesta HTTP con código 204.	Sí
Un usuario que ya abandonó una sala la intenta abandonar.	El usuario no abandona. Se recibe una respuesta HTTP con código 400.	Sí

Eliminar sala

Prueba	Resultado Esperado	Superada
El administrador elimina la sala.	Los participantes son expulsados y existe una sala menos en el sistema. Se recibe una respuesta HTTP con código 204.	Sí
Un moderador elimina la sala.	Los participantes no son expulsados y no existe una sala menos en el	Sí

	sistema. Se recibe una respuesta HTTP con código 400.	
Un miembro elimina la sala.	Los participantes no son expulsados y no existe una sala menos en el sistema. Se recibe una respuesta HTTP con código 400.	Sí
Eliminar sala que no existe.	No se modifica ningún dato. Se recibe una respuesta HTTP con código 400.	Sí

Modificar datos cuenta

Prueba	Resultado Esperado	Superada
Se envía un nombre, apellidos y email válidos.	Se modifican los datos del usuario. Se recibe una respuesta HTTP con código 204.	Sí
Se envía un nombre, apellidos o email no válidos.	No se modifican los datos del usuario. Se recibe una respuesta HTTP con código 400.	Sí

Modificar contraseña

Prueba	Resultado Esperado	Superada
La contraseña actual es correcta.	Se modifica la contraseña del usuario. Se recibe una respuesta HTTP con código 204.	Sí
La contraseña actual es incorrecta.	No se modifica la contraseña del usuario. Se recibe una respuesta HTTP con código 403.	Sí

Eliminar cuenta

Prueba	Resultado Esperado	Superada
Eliminar cuenta de un usuario.	Existe un usuario menos en el sistema. Se recibe una respuesta HTTP con código 204.	Sí
Eliminar cuenta de un usuario que no existe.	No se modifica ningún dato. Se recibe una respuesta HTTP con código 400.	Sí

El framework “Mocha” permite que, para cada una de las anteriores tablas, se agrupen todas las pruebas necesarias bajo el nombre de la tabla.

Debido a que para realizar ciertas pruebas es necesario que antes los usuarios realicen acciones como intercambiar solicitudes de contacto, de salas, que se acepten las solicitudes, etc. Se han realizado algunas pruebas adicionales de cada caso para poder incluir en la base de datos la información necesaria. Esto también fue útil para probar las distintas acciones en diferentes contextos. Teniendo esto último en cuenta, se han realizado en total 150 pruebas.

A continuación, se muestra un pantallazo con el objetivo de mostrar cómo se organizan las pruebas en “Mocha” bajo el nombre de la tabla:

```

Cambiar permisos moderadores
✓ El usuario3 debería ser miembro en 0 salas
✓ El usuario1 al no ser administrador de la sala 'Sala de pruebas' no debería cambiar los permisos al usuario3 ya que este es moderador
✓ El usuario3 debería ser miembro en 0 salas
✓ El usuario2 al ser administrador de la sala 'Sala de pruebas' debería cambiar los permisos al moderador usuario3 a miembro
✓ El usuario3 debería ser miembro en 1 sala

Eliminar miembro
✓ Debería de haber 4 participantes en la sala 'Sala de pruebas'
✓ El usuario5 al ser miembro de la sala 'Sala de pruebas' no debería eliminar al usuario3 de la sala ya que es miembro
✓ Debería de haber 4 participantes en la sala 'Sala de pruebas'
✓ El usuario1 al ser moderador de la sala 'Sala de pruebas' debería eliminar al usuario2 de la sala ya que es miembro (47ms)
✓ El usuario2 al ser administrador de la sala 'Sala de pruebas' debería eliminar al usuario5 de la sala ya que es miembro
✓ Debería de haber 2 participantes en la sala 'Sala de pruebas'

Eliminar moderador
✓ El usuario2 al ser administrador de la sala 'Sala de pruebas' debería enviar una solicitud de unión a usuario3
✓ El usuario2 al ser administrador de la sala 'Sala de pruebas' debería enviar una solicitud de unión a usuario5
✓ El usuario2 al ser administrador de la sala 'Sala de pruebas' cambió los permisos otorgados a usuario5 en la solicitud de unión a moderador
✓ El usuario3 debería aceptar la solicitud de unión a la sala del usuario2
✓ El usuario5 debería aceptar la solicitud de unión a la sala del usuario2
✓ Debería de haber 4 participantes en la sala 'Sala de pruebas'
✓ El usuario3 al ser miembro de la sala 'Sala de pruebas' no debería eliminar de la sala al usuario1 que es moderador
✓ El usuario5 al ser moderador de la sala 'Sala de pruebas' no debería eliminar de la sala al usuario1 que es moderador
✓ El usuario2 al ser administrador de la sala 'Sala de pruebas' debería eliminar de la sala al usuario1 que es moderador
✓ Debería de haber 3 participantes en la sala 'Sala de pruebas'

Cambiar permisos solicitud
✓ El usuario2 al ser administrador de la sala 'Sala de pruebas' debería mandar una solicitud de unión a la sala como miembro al usuario1
✓ El usuario3 al ser miembro de la sala 'Sala de pruebas' no debería cambiar los permisos de la solicitud de unión a la sala del usuario1 de miembro a moderador
✓ El usuario2 al ser administrador de la sala 'Sala de pruebas' debería cambiar los permisos de la solicitud de unión a la sala del usuario1 de miembro a moderador
✓ El usuario1 debería aceptar la solicitud de unión a la sala del usuario2
✓ El usuario1 debería participar como moderador en 1 sala
✓ El usuario2 al ser administrador de la sala 'Sala de pruebas' eliminar al usuario5 de la sala
✓ El usuario2 al ser administrador de la sala 'Sala de pruebas' debería mandar una solicitud de unión a la sala como miembro al usuario5
✓ El usuario1 al ser moderador de la sala 'Sala de pruebas' debería cambiar los permisos de la solicitud de unión a la sala del usuario5 de miembro a moderador
✓ El usuario5 debería aceptar la solicitud de unión a la sala del usuario2 (38ms)
✓ El usuario5 debería participar como moderador en 1 sala

Abandonar sala
✓ Debería de haber 4 participantes en la sala 'Sala de pruebas'
✓ El usuario3 al ser miembro de la sala 'Sala de pruebas' debería salir de la sala
✓ El usuario1 al ser moderador de la sala 'Sala de pruebas' debería salir de la sala
✓ El usuario2 al ser administrador no abandona la sala 'Sala de pruebas'
✓ Debería de haber 2 participantes en la sala 'Sala de pruebas'

Eliminar sala
✓ El usuario5 al ser miembro de la sala 'Sala de pruebas' no la debería eliminar
✓ El usuario3 al ser moderador de la sala 'Sala de pruebas 4' no la debería eliminar
✓ El usuario1 al ser administrador de la sala 'Sala de pruebas 4' debería eliminarla
✓ El usuario1 no debería eliminar la 'Sala de pruebas 4' porque no existe
✓ El usuario3 no debería ser moderador en ninguna sala ya que se eliminó la sala 'Sala de pruebas 4'
✓ El usuario1 debería aceptar la solicitud de la sala 'Sala de pruebas 2'
✓ El usuario1 al ser miembro de la 'Sala de pruebas 2' no debería eliminarla

Modificar datos cuenta
✓ El usuario1 debería cambiar su nombre a 'Jose', sus apellidos a 'Ejemplos de apellidos' y su email a 'jose@gmail.com'
✓ El usuario1 no debería cambiar su nombre ya que es demasiado largo

```

Figura 8.1. Pruebas unitarias con Mocha

Por último, cabe destacar que se utilizó el “Hook” de mocha “before” para eliminar la información de la base de datos y dejarla así preparada antes de comenzar las pruebas.

8.2 Pruebas de Integración y del Sistema

Como se ha comentado con anterioridad, estas pruebas fueron realizadas utilizando los navegadores Firefox y Chrome, simulando que dos usuarios independientes están utilizando la aplicación. A continuación, se muestran los resultados:

<i>Registrarse</i>		
Prueba	Resultado Esperado	Superada
Registrar usuario con nombre y email válidos y nombre de usuario no utilizado.	El sistema devuelve un token de sesión y el usuario accede a la página principal de la aplicación.	Sí
Registrar usuario con nombre de usuario ya utilizado.	No se permite al usuario registrarse y se muestra un mensaje indicándole que el nombre de usuario está en uso.	Sí
Registrar usuario con nombre y email erróneos o contraseña y contraseña de confirmación que no coinciden.	No se registra al usuario en el sistema y se muestra un mensaje notificando el problema.	Sí

<i>Caso de Uso: Iniciar sesión</i>		
Prueba	Resultado Esperado	Superada
Iniciar sesión con nombre de usuario y contraseña correctos.	El sistema devuelve un token de sesión y el usuario accede a la página principal de la aplicación.	Sí
Iniciar sesión con nombre de usuario y contraseña incorrectos.	Se muestra un mensaje notificando que el usuario o la contraseña son incorrectos.	Sí

<i>Caso de Uso: Cambiar idioma</i>		
Prueba	Resultado Esperado	Superada
Se pulsa el botón para cambiar idioma a inglés.	Los textos mostrados en la aplicación están en inglés.	Sí
Se pulsa el botón para cambiar idioma a español.	Los textos mostrados en la aplicación están en español.	Sí

<i>Modificar datos cuenta</i>		
Prueba	Resultado Esperado	Superada
Se envía un nombre, apellidos y email válidos.	Se modifican los datos del usuario y se muestra un mensaje notificándolo.	Sí

Se envía un nombre, apellidos o email no válidos.	No se modifican los datos del usuario y se muestra un mensaje notificándolo.	Sí
---	--	----

Crear sala

Prueba	Resultado Esperado	Superada
Un usuario crea una sala con un nombre y descripción válido.	Se añade una nueva sala al sistema donde el usuario es el administrador. Se muestra una notificación en la que se indica que la sala fue creada.	Sí
Un usuario crea una sala con un nombre y descripción válido, y especifica participantes con sus permisos.	Se añade una nueva sala al sistema donde el usuario es el administrador y los participantes especificados son moderadores o miembros dependiendo de los permisos dados. Se muestra una notificación en la que se indica que la sala fue creada.	Sí
Un usuario crea una sala con un nombre y descripción no válidos.	No se añade una nueva sala al sistema. Se muestra una notificación en la que se indica cuál fue el problema.	Sí

Enviar solicitud de contacto

Prueba	Resultado Esperado	Superada
Un usuario envía una solicitud de contacto a otro usuario.	Se envía la solicitud de contacto al usuario. Se muestra una notificación en la que se indica que se envió la solicitud.	Sí

Aceptar solicitud contacto

Prueba	Resultado Esperado	Superada
Un usuario acepta una solicitud de contacto.	El usuario tiene un nuevo contacto en su listado de contactos. Se muestra una notificación indicando que la solicitud fue aceptada.	Sí

<i>Ignorar solicitud contacto</i>		
Prueba	Resultado Esperado	Superada
Un usuario ignora una solicitud de contacto.	El usuario no tiene un nuevo contacto en su listado de contactos.	Sí

<i>Aceptar solicitud de unión a sala</i>		
Prueba	Resultado Esperado	Superada
Un usuario acepta una solicitud de unión a sala.	El usuario tiene una nueva sala en su listado de salas. Se muestra una notificación de que la solicitud fue aceptada.	Sí
Un usuario acepta una solicitud de unión a sala justo después de que fuera eliminada.	El usuario no tiene una nueva sala en su listado de salas. Se muestra una notificación de que hubo un problema al aceptar la solicitud.	Sí

<i>Ignorar solicitud de unión a sala</i>		
Prueba	Resultado Esperado	Superada
Un usuario ignora una solicitud de unión a sala.	El usuario no tiene una nueva sala en su listado de salas. Se muestra un mensaje de que la solicitud fue ignorada.	Sí

<i>Acceder a sala</i>		
Prueba	Resultado Esperado	Superada
Se pulsa en el botón para acceder a una sala.	Se accede a la página de la sala.	Sí
Prueba	Resultado Esperado	
Se pulsa en el botón para acceder a una sala en la que nos quitaron el permiso de acceso.	No se accede a la página de la sala. Se muestra una notificación diciendo que el usuario no está autorizado para acceder a la sala.	Sí

<i>Filtrar salas</i>		
Prueba	Resultado Esperado	Superada
Se escribe un texto en el campo para filtrar sala.	Se muestran aquellas salas cuyos nombres contienen el texto.	Sí

<i>Filtrar contactos</i>		
Prueba	Resultado Esperado	Superada
Se escribe un texto en el campo para filtrar contactos.	Se muestran aquellos contactos cuyos nombres de usuario contienen el texto.	Sí

<i>Bloquear contacto</i>		
Prueba	Resultado Esperado	Superada
Bloquear un contacto.	El contacto desaparece del listado de contactos y es expulsado de todas las salas donde el usuario que lo bloqueó es administrador. Se muestra el contacto en un listado de contactos bloqueados.	Sí

<i>Desbloquear contacto</i>		
Prueba	Resultado Esperado	Superada
Desbloquear un contacto.	El contacto aparece en la lista de contactos y se elimina del listado de contactos bloqueados.	Sí

<i>Eliminar cuenta</i>		
Prueba	Resultado Esperado	Superada
Eliminar cuenta de un usuario.	Se cierra sesión redirigiendo al usuario a la página de inicio de sesión y registro.	Sí

<i>Eliminar sala</i>		
Prueba	Resultado Esperado	Superada
Eliminar sala.	Existe una sala menos en el listado de salas y los participantes son expulsados.	Sí

<i>Cambiar permisos moderadores</i>		
Prueba	Resultado Esperado	Superada
Se modifican los permisos de un moderador.	El moderador pasa a tener permisos de miembro. Se muestra una notificación indicando que los permisos fueron modificados.	Sí

<i>Eliminar moderadores</i>		
Prueba	Resultado Esperado	Superada
Eliminar moderador de una sala.	Existe un participante menos en la sala. Se muestra una notificación indicando que el usuario fue eliminado.	Sí

<i>Cambiar permisos solicitud</i>		
Prueba	Resultado Esperado	Superada
Cambiar permisos.	En la solicitud se especifican otros permisos. Se muestra una notificación indicando que los permisos fueron modificados.	Sí

<i>Eliminar solicitud</i>		
Prueba	Resultado Esperado	Superada
Eliminar solicitud de unión a sala.	Existe una solicitud de unión a la sala menos. Se muestra una notificación indicando que la solicitud fue eliminada.	Sí

<i>Cambiar permisos miembros</i>		
Prueba	Resultado Esperado	Superada
Cambiar permisos de miembro.	El usuario pasa a tener permisos de moderador. Se muestra una notificación indicando que los permisos fueron modificados.	Sí

<i>Enviar solicitud de unión a sala</i>		
Prueba	Resultado Esperado	Superada
Enviar solicitud de unión a sala.	Existe una solicitud de unión más en la sala. Se muestra una notificación indicando que la solicitud fue enviada.	Sí
Enviar solicitud de unión a sala cuando las solicitudes de las salas y los participantes suman 4.	No se envía la solicitud y se notifica de ello.	Sí

<i>Eliminar miembro</i>		
Prueba	Resultado Esperado	Superada
Eliminar miembro de una sala.	Existe un participante menos en la sala. Se muestra una notificación indicando que el usuario fue eliminado.	Sí

<i>Abandonar sala</i>		
Prueba	Resultado Esperado	Superada
Abandonar sala.	Se muestra una ventana emergente para confirmar la acción. Si se acepta no se tendrá acceso a la sala.	Sí

<i>Abrir servicio</i>		
Prueba	Resultado Esperado	Superada
Abrir servicio.	Aparece el servicio seleccionado en la pantalla.	Sí
Abrir servicio que ya fue abierto.	No pasa nada y se muestra un mensaje al usuario.	Sí

<i>Cerrar servicio</i>		
Prueba	Resultado Esperado	Superada
Cerrar servicio.	El servicio desaparece de la pantalla.	Sí

<i>Mover servicio</i>		
Prueba	Resultado Esperado	Superada
Mover servicio.	El servicio cambia su posición.	Sí

<i>Redimensionar servicio</i>		
Prueba	Resultado Esperado	Superada
Redimensionar servicio.	El servicio cambia de tamaño y su contenido se adapta a él.	Sí

<i>Crear dibujo colaborativo</i>		
Prueba	Resultado Esperado	Superada
Seleccionar botón de una figura.	Aparece la figura en el dibujo. El dibujo aparece para los usuarios que tienen el servicio abierto.	Sí
Seleccionar el lápiz y dibujar por el canvas.	Aparece un dibujo con la forma del recorrido del puntero. El dibujo aparece para los usuarios que tienen el servicio abierto.	Sí
Utilizar el selector para mover elementos por el canvas.	Los cambios aparecen para los usuarios que tienen el servicio abierto.	Sí

<i>Compartir URL video</i>		
Prueba	Resultado Esperado	Superada
Se comparte URL de video.	Aparece el video en el servicio de video compartido y en el de los usuarios conectados al servicio. Se muestra una notificación a los otros	Sí

	usuarios indicándoles que el video fue modificado.	
Se comparte URL de video no soportado.	No se reproduce ningún video.	Sí

Compartir presentación

Prueba	Resultado Esperado	Superada
Se comparte presentación.	Aparece la presentación en la pantalla del servicio de presentaciones y en el de los usuarios conectados al servicio. Se muestra una notificación a los otros usuarios indicándoles que la presentación fue modificada.	Sí

Participar en videollamada

Prueba	Resultado Esperado	Superada
Abrir servicio de videollamada.	Aparece en el servicio los videos de los usuarios conectados a la videollamada y el video local.	Sí
Usuario pulsa botón de silenciar micrófono.	Los otros usuarios conectados a la videollamada no pueden escuchar el audio del usuario.	Sí
Usuario pulsa botón de silenciar altavoces.	El usuario no puede escuchar a los otros usuarios conectados a la videollamada.	Sí
Usuario pulsa botón de pausar transmisión de video.	Los otros usuarios conectados a la videollamada no pueden ver el video del usuario.	Sí

Enviar mensaje de texto

Prueba	Resultado Esperado	Superada
El usuario envía mensaje de texto.	Aparece el mensaje en el servicio de chat de texto, y los usuarios conectados al servicio pueden ver el mensaje. Además, el usuario ve cuantos participantes recibieron el mensaje.	Sí

<i>Enviar fichero</i>		
Prueba	Resultado Esperado	Superada
El usuario envía una imagen.	Aparece la imagen en el servicio de chat de texto y los usuarios conectados al servicio pueden verla. Además, el usuario ve cuantos participantes recibieron el mensaje.	Sí
El usuario envía un documento.	Aparece una imagen de un documento en el servicio de chat de texto y los usuarios conectados al servicio la ven. Además, el usuario ve cuantos participantes recibieron el mensaje.	Sí

<i>Enviar mensaje de texto</i>		
Prueba	Resultado Esperado	Superada
El usuario pulsa en el fichero para descargarlo.	Aparece la ventana de diálogo por defecto del navegador que permite descargar el fichero.	Sí, pero se han detectado problemas al descargar algunos ficheros únicamente con Chrome.

8.3 Pruebas de Usabilidad y Accesibilidad

8.3.1 Pruebas de Usabilidad

En esta sección se muestran los resultados de los cuestionarios realizados por personas ajenas a este proyecto y sus consecuencias, y el resultado obtenido al realizar la guía de usabilidad realizada por Yusef Hassan Montero.

8.3.1.1 *Actividades de las Pruebas de Usabilidad*

A continuación, se muestran los resultados del cuestionario realizado en el apartado 6.7.3.2 Actividades de las Pruebas de Usabilidad:

8.3.1.2 Usuario 1

8.3.1.2.1 Datos personales

- Profesión: Estudiante de secundaria
- Edad: 15 años

8.3.1.2.2 Preguntas de carácter general

¿Usa un ordenador frecuentemente?
<ol style="list-style-type: none">1. Todos los días *2. Varias veces a la semana3. Ocasionalmente4. Nunca o casi nunca
¿Qué tipo de actividades realiza con el ordenador?
<ol style="list-style-type: none">1. Es parte de mi trabajo o profesión2. Lo uso básicamente para ocio *3. Solo empleo aplicaciones estilo Office4. Únicamente leo el correo y navego ocasionalmente
¿Ha usado alguna vez software como el de esta prueba?
<ol style="list-style-type: none">1. Sí, he empleado software similar *2. No, aunque si empleo otros programas que me ayudan a realizar tareas similares3. No, nunca
¿Qué busca Vd. Principalmente en un programa?
<ol style="list-style-type: none">1. Que sea fácil de usar2. Que sea intuitivo *3. Que sea rápido4. Que tenga todas las funciones necesarias
¿Suele registrarse en las páginas web que le parecen interesantes?
<ol style="list-style-type: none">1. Siempre *2. Normalmente, aunque en ocasiones busco páginas alternativas si me obligan a registrarme3. Nunca
¿Conoce algún software que le permita realizar dibujos colaborativos con sus contactos?
<ol style="list-style-type: none">1. Sí, y lo he utilizado *2. Sí, pero nunca lo he utilizado3. No, pero puede ser de utilidad

4. No, no me interesa esa función
¿Le parece interesante que sus contactos compartan enlaces de videos o música y que usted pueda reproducirlos en la misma aplicación?
<ol style="list-style-type: none"> 1. Si, ya utilizo aplicaciones que me dan esa posibilidad 2. Sí, me parece una función interesante * 3. No, me parece una función innecesaria

8.3.1.2.3 Preguntas Cortas sobre la Aplicación y Observaciones

Facilidad de Uso	Siempre	Frecuentemente	Ocasionalmente	Nunca
<i>¿Sabe dónde está dentro de la aplicación?</i>	X			
<i>¿La aplicación le informa del resultado de sus acciones?</i>	X			
<i>¿Sabe cuándo las notificaciones son de errores, informativas o de éxito?</i>	X			
<i>¿Le resulta sencillo el uso de la aplicación?</i>	X			
<i>¿Sabe a dónde tiene que dirigirse para realizar las acciones que desea?</i>	X			
<i>¿Sabe cómo encontrar información de sus contactos?</i>	X			
<i>¿Sabe si es administrador, moderador o miembro de las salas?</i>	X			
<i>¿Sabe que usuarios están conectados en una sala?</i>	X			
Funcionalidad	Siempre	Frecuentemente	Ocasionalmente	Nunca
<i>¿Puede cambiar el idioma de la aplicación?</i>	X			
<i>¿Puede añadir nuevos contactos?</i>	X			
<i>¿Puede bloquear y desbloquear contactos?</i>	X			
<i>¿Sabe cómo acceder a los ajustes de su cuenta?</i>	X			
<i>¿Sabe crear salas?</i>	X			

<i>¿Puede añadir usuarios a sus salas?</i>	X			
<i>¿Puede acceder a las salas?</i>	X			
<i>¿Identifica los servicios ofrecidos por la aplicación?</i>	X			
<i>¿Puede abrir y cerrar servicios?</i>	X			
<i>¿Sabe cómo mover y redimensionar servicios?</i>	X			
<i>¿Sabe cómo dibujar en el servicio “Dibujos”?</i>	X			
<i>¿Puede unirse a videollamadas?</i>	X			
<i>¿Puede enviar mensajes de texto y ficheros?</i>	X			
<i>¿Conoce cuántos usuarios reciben sus mensajes?</i>	X			
<i>¿Puede descargar los ficheros recibidos?</i>	X			
<i>¿Sabe cómo modificar la presentación mostrada en el servicio “Presentaciones”?</i>	X			
<i>¿Sabe cómo modificar el video mostrado en el servicio “Video compartido”?</i>	X			
<i>¿Sabe cómo modificar la radio/canción reproducida en el servicio “Radio”?</i>	X			
<i>¿Sabe cuándo la presentación es modificada por otro usuario en el servicio “Presentaciones”?</i>	X			
<i>¿Sabe cuándo el video es modificado por otro usuario en el servicio “Video compartido”?</i>	X			
<i>¿Sabe cuándo se modifica el dibujo por otro usuario en el servicio “Dibujos”?</i>	X			

<i>¿Sabe cuándo la radio/canción es modificada por otro usuario en el servicio "Radio"?</i>	X			
Calidad del Interfaz				
Aspectos gráficos	Muy Adecuado	Adecuado	Poco Adecuado	Nada Adecuado
<i>El tipo y tamaño de letra es</i>		X		
<i>Los iconos e imágenes usados son</i>			X	
<i>Los colores empleados son</i>	X			
<i>La distribución de los elementos en las vistas es</i>		X		
<i>Las animaciones son</i>		X		
<i>La distribución de las salas en servicios que se pueden abrir, cerrar, mover y redimensionar es</i>	X			
<i>La información mostrada en las tablas es</i>		X		
<i>Mostrar todas las herramientas para la gestión de todas las salas en una única interfaz es</i>	X			
Diseño de la Interfaz		Si	No	A veces
<i>¿Le resulta fácil de usar?</i>		X		
<i>¿El diseño de las pantallas es claro y atractivo?</i>		X		
<i>¿Cree que la aplicación está bien estructurada?</i>		X		
<i>¿Cree que es sencillo utilizar los servicios de la sala?</i>		X		
<i>¿Cree necesario mostrar más notificaciones?</i>				X
<i>¿Piensa que es adecuado la utilización de cuadros de diálogos?</i>		X		
Observaciones				
Mostrar iconos en el menú lateral.				

8.3.1.2.4 Cuestionario para el Responsable de las Pruebas

Aspecto Observado	Notas
<i>El usuario comienza a trabajar de forma rápida por las tareas</i>	Se registra en la aplicación sin ningún problema y comienza a realizar las actividades guiadas sin ningún problema destacable.

<i>Tiempo en realizar cada tarea</i>	Las tareas fueron realizadas en el tiempo previsto.
<i>Errores leves cometidos</i>	Ninguno.
<i>Errores graves cometidos</i>	Ninguno.
<i>Preguntas</i>	¿Puedo mandar un mensaje a un contacto sin crear una sala?
<i>Mayor dificultad</i>	No se ha presenciado ninguna dificultad a destacar.

8.3.1.3 Usuario 2

8.3.1.3.1 Datos personales

- Profesión: Estudiante de Lenguas Modernas
- Edad: 21 años

8.3.1.3.2 Preguntas de carácter general

¿Usa un ordenador frecuentemente?
<ol style="list-style-type: none"> 1. Todos los días 2. Varias veces a la semana * 3. Ocasionalmente 4. Nunca o casi nunca
¿Qué tipo de actividades realiza con el ordenador?
<ol style="list-style-type: none"> 1. Es parte de mi trabajo o profesión 2. Lo uso básicamente para ocio 3. Solo empleo aplicaciones estilo Office 4. Únicamente leo el correo y navego ocasionalmente *
¿Ha usado alguna vez software como el de esta prueba?
<ol style="list-style-type: none"> 1. Sí, he empleado software similar * 2. No, aunque si empleo otros programas que me ayudan a realizar tareas similares 3. No, nunca
¿Qué busca Vd. Principalmente en un programa?
<ol style="list-style-type: none"> 1. Que sea fácil de usar 2. Que sea intuitivo 3. Que sea rápido 4. Que tenga todas las funciones necesarias *
¿Suele registrarse en las páginas web que le parecen interesantes?

<ol style="list-style-type: none"> 1. Siempre 4. Normalmente, aunque en ocasiones busco páginas alternativas si me obligan a registrarme * 5. Nunca
¿Conoce algún software que le permita realizar dibujos colaborativos con sus contactos?
<ol style="list-style-type: none"> 1. Sí, y lo he utilizado * 2. Sí, pero nunca lo he utilizado 3. No, pero puede ser de utilidad 4. No, no me interesa esa función
¿Le parece interesante que sus contactos compartan enlaces de videos o música y que usted pueda reproducirlos en la misma aplicación?
<ol style="list-style-type: none"> 1. Si, ya utilizo aplicaciones que me dan esa posibilidad 2. Sí, me parece una función interesante * 3. No, me parece una función innecesaria

8.3.1.3.3 Preguntas Cortas sobre la Aplicación y Observaciones

Facilidad de Uso	Siempre	Frecuentemente	Ocasionalmente	Nunca
¿Sabe dónde está dentro de la aplicación?	X			
¿La aplicación le informa del resultado de sus acciones?	X			
¿Sabe cuándo las notificaciones son de errores, informativas o de éxito?	X			
¿Le resulta sencillo el uso de la aplicación?	X			
¿Sabe a dónde tiene que dirigirse para realizar las acciones que desea?		X		
¿Sabe cómo encontrar información de sus contactos?		X		
¿Sabe si es administrador, moderador o miembro de las salas?	X			
¿Sabe que usuarios están conectados en una sala?	X			

Funcionalidad	Siempre	Frecuentemente	Ocasionalmente	Nunca
<i>¿Puede cambiar el idioma de la aplicación?</i>	X			
<i>¿Puede añadir nuevos contactos?</i>	X			
<i>¿Puede bloquear y desbloquear contactos?</i>	X			
<i>¿Sabe cómo acceder a los ajustes de su cuenta?</i>	X			
<i>¿Sabe crear salas?</i>	X			
<i>¿Puede añadir usuarios a sus salas?</i>	X			
<i>¿Puede acceder a las salas?</i>	X			
<i>¿Identifica los servicios ofrecidos por la aplicación?</i>	X			
<i>¿Puede abrir y cerrar servicios?</i>	X			
<i>¿Sabe cómo mover y redimensionar servicios?</i>	X			
<i>¿Sabe cómo dibujar en el servicio “Dibujos”?</i>	X			
<i>¿Puede unirse a videollamadas?</i>	X			
<i>¿Puede enviar mensajes de texto y ficheros?</i>	X			
<i>¿Conoce cuántos usuarios reciben sus mensajes?</i>		X		
<i>¿Puede descargar los ficheros recibidos?</i>	X			
<i>¿Sabe cómo modificar la presentación mostrada en el servicio “Presentaciones”?</i>	X			
<i>¿Sabe cómo modificar el video mostrado en el servicio “Video compartido”?</i>	X			
<i>¿Sabe cómo modificar la radio/canción reproducida en el servicio “Radio”?</i>	X			
<i>¿Sabe cuándo la presentación es modificada por otro</i>	X			

<i>usuario en el servicio “Presentaciones”?</i>				
<i>¿Sabe cuándo el video es modificado por otro usuario en el servicio “Video compartido”?</i>		X		
<i>¿Sabe cuándo se modifica el dibujo por otro usuario en el servicio “Dibujos”?</i>		X		
<i>¿Sabe cuándo la radio/canción es modificada por otro usuario en el servicio “Radio”?</i>		X		
Calidad del Interfaz				
Aspectos gráficos	Muy Adecuado	Adecuado	Poco Adecuado	Nada Adecuado
<i>El tipo y tamaño de letras</i>	X			
<i>Los iconos e imágenes usados son</i>	X			
<i>Los colores empleados son</i>	X			
<i>La distribución de los elementos en las vistas es</i>	X			
<i>Las animaciones son</i>	X			
<i>La distribución de las salas en servicios que se pueden abrir, cerrar, mover y redimensionar es</i>	X			
<i>La información mostrada en las tablas es</i>	X			
<i>Mostrar todas las herramientas para la gestión de todas las salas en una única interfaz es</i>	X			
Diseño de la Interfaz		Si	No	A veces
<i>¿Le resulta fácil de usar?</i>		X		
<i>¿El diseño de las pantallas es claro y atractivo?</i>		X		
<i>¿Cree que la aplicación está bien estructurada?</i>		X		
<i>¿Cree que es sencillo utilizar los servicios de la sala?</i>		X		
<i>¿Cree necesario mostrar más notificaciones?</i>			X	

¿Piensa que es adecuado la utilización de cuadros de diálogos?	X		
Observaciones			
Ninguna.			

8.3.1.3.4 Cuestionario para el Responsable de las Pruebas

Aspecto Observado	Notas
<i>El usuario comienza a trabajar de forma rápida por las tareas</i>	Se registra en la aplicación sin ningún problema y comienza a realizar las actividades guiadas sin ningún problema destacable, hasta que el usuario tuvo dudas referentes a las solicitudes y a añadir usuarios a salas.
<i>Tiempo en realizar cada tarea</i>	Las tareas fueron realizadas en el tiempo previsto.
<i>Errores leves cometidos</i>	Para acceder a la gestión de las salas, accedió directamente a una sala pensando que se encontraba ahí. Resolvió el problema sin muchas complicaciones al volver a la página principal.
<i>Errores graves cometidos</i>	Pensaba que los usuarios tenían acceso a una sala que creó con tan solo añadirlos.
<i>Preguntas</i>	<ul style="list-style-type: none"> - ¿Por qué no aparece la sala para un usuario que acabo de añadir? - ¿Cómo funcionan los permisos de una solicitud y de un participante?
<i>Mayor dificultad</i>	Entender el funcionamiento de las solicitudes y permisos. Una vez explicadas las preguntas que tenía el usuario, pudo utilizar la aplicación sin ningún problema.

8.3.1.4 Usuario 3

8.3.1.4.1 Datos personales

- Profesión: Hostelero
- Edad: 50 años

8.3.1.4.2 Preguntas de carácter general

¿Usa un ordenador frecuentemente?
<ul style="list-style-type: none"> 5. Todos los días 6. Varias veces a la semana * 7. Ocasionalmente 8. Nunca o casi nunca
¿Qué tipo de actividades realiza con el ordenador?
<ul style="list-style-type: none"> 5. Es parte de mi trabajo o profesión * 6. Lo uso básicamente para ocio 7. Solo empleo aplicaciones estilo Office 8. Únicamente leo el correo y navego ocasionalmente
¿Ha usado alguna vez software como el de esta prueba?
<ul style="list-style-type: none"> 4. Sí, he empleado software similar 5. No, aunque si empleo otros programas que me ayudan a realizar tareas similares * 6. No, nunca
¿Qué busca Vd. Principalmente en un programa?
<ul style="list-style-type: none"> 5. Que sea fácil de usar * 6. Que sea intuitivo 7. Que sea rápido 8. Que tenga todas las funciones necesarias
¿Suele registrarse en las páginas web que le parecen interesantes?
<ul style="list-style-type: none"> 6. Siempre * 7. Normalmente, aunque en ocasiones busco páginas alternativas si me obligan a registrarme 8. Nunca
¿Conoce algún software que le permita realizar dibujos colaborativos con sus contactos?
<ul style="list-style-type: none"> 5. Sí, y lo he utilizado 6. Sí, pero nunca lo he utilizado 7. No, pero puede ser de utilidad * 8. No, no me interesa esa función
¿Le parece interesante que sus contactos compartan enlaces de videos o música y que usted pueda reproducirlos en la misma aplicación?
<ul style="list-style-type: none"> 4. Si, ya utilizo aplicaciones que me dan esa posibilidad

5. Sí, me parece una función interesante *
6. No, me parece una función innecesaria

8.3.1.4.3 Preguntas Cortas sobre la Aplicación y Observaciones

Facilidad de Uso	Siempre	Frecuentemente	Ocasionalment e	Nunca
<i>¿Sabe dónde está dentro de la aplicación?</i>		X		
<i>¿La aplicación le informa del resultado de sus acciones?</i>		X		
<i>¿Sabe cuándo las notificaciones son de errores, informativas o de éxito?</i>	X			
<i>¿Le resulta sencillo el uso de la aplicación?</i>	X			
<i>¿Sabe a dónde tiene que dirigirse para realizar las acciones que desea?</i>		X		
<i>¿Sabe cómo encontrar información de sus contactos?</i>		X		
<i>¿Sabe si es administrador, moderador o miembro de las salas?</i>		X		
<i>¿Sabe que usuarios están conectados en una sala?</i>		X		
Funcionalidad	Siempre	Frecuentemente	Ocasionalment e	Nunca
<i>¿Puede cambiar el idioma de la aplicación?</i>	X			
<i>¿Puede añadir nuevos contactos?</i>	X			
<i>¿Puede bloquear y desbloquear contactos?</i>	X			
<i>¿Sabe cómo acceder a los ajustes de su cuenta?</i>	X			
<i>¿Sabe crear salas?</i>	X			
<i>¿Puede añadir usuarios a sus salas?</i>	X			
<i>¿Puede acceder a las salas?</i>	X			

<i>¿Identifica los servicios ofrecidos por la aplicación?</i>	X			
<i>¿Puede abrir y cerrar servicios?</i>	X			
<i>¿Sabe cómo mover y redimensionar servicios?</i>	X			
<i>¿Sabe cómo dibujar en el servicio “Dibujos”?</i>	X			
<i>¿Puede unirse a videollamadas?</i>	X			
<i>¿Puede enviar mensajes de texto y ficheros?</i>	X			
<i>¿Conoce cuántos usuarios reciben sus mensajes?</i>	X			
<i>¿Puede descargar los ficheros recibidos?</i>	X			
<i>¿Sabe cómo modificar la presentación mostrada en el servicio “Presentaciones”?</i>	X			
<i>¿Sabe cómo modificar el video mostrado en el servicio “Video compartido”?</i>	X			
<i>¿Sabe cómo modificar la radio/canción reproducida en el servicio “Radio”?</i>	X			
<i>¿Sabe cuándo la presentación es modificada por otro usuario en el servicio “Presentaciones”?</i>		X		
<i>¿Sabe cuándo el video es modificado por otro usuario en el servicio “Video compartido”?</i>		X		
<i>¿Sabe cuándo se modifica el dibujo por otro usuario en el servicio “Dibujos”?</i>	X			
<i>¿Sabe cuándo la radio/canción es modificada por otro usuario en el servicio “Radio”?</i>		X		

Calidad del Interfaz				
Aspectos gráficos	Muy Adecuado	Adecuado	Poco Adecuado	Nada Adecuado
<i>El tipo y tamaño de letras</i>		X		
<i>Los iconos e imágenes usados son</i>		X		
<i>Los colores empleados son</i>		X		
<i>La distribución de los elementos en las vistas es</i>	X			
<i>Las animaciones son</i>	X			
<i>La distribución de las salas en servicios que se pueden abrir, cerrar, mover y redimensionar es</i>	X			
<i>La información mostrada en las tablas es</i>		X		
<i>Mostrar todas las herramientas para la gestión de todas las salas en una única interfaz es</i>	X			
Diseño de la Interfaz		Si	No	A veces
<i>¿Le resulta fácil de usar?</i>		X		
<i>¿El diseño de las pantallas es claro y atractivo?</i>		X		
<i>¿Cree que la aplicación está bien estructurada?</i>		X		
<i>¿Cree que es sencillo utilizar los servicios de la sala?</i>		X		
<i>¿Cree necesario mostrar más notificaciones?</i>				X
<i>¿Piensa que es adecuado la utilización de cuadros de diálogos?</i>		X		
Observaciones				
Ninguna.				

8.3.1.4.4 Cuestionario para el Responsable de las Pruebas

Aspecto Observado	Notas
<i>El usuario comienza a trabajar de forma rápida por las tareas</i>	Se registra en la aplicación sin ningún problema y comienza a realizar las actividades guiadas sin ningún problema destacable, aunque tuvo ciertas dudas con los efectos de añadir usuarios a una sala que estaba creando.

<i>Tiempo en realizar cada tarea</i>	Las tareas fueron realizadas en el tiempo previsto.
<i>Errores leves cometidos</i>	Ninguno.
<i>Errores graves cometidos</i>	Ninguno.
<i>Preguntas</i>	Habías dicho que los usuarios reciben solicitudes para unirse a la sala, ¿Por qué al crear una sala dice que los añade directamente?
<i>Mayor dificultad</i>	No se ha presenciado ninguna dificultad a destacar.

8.3.1.5 Cambios producidos por los anteriores cuestionarios

Tras la realización de los cuestionarios, se han detectado ciertos cambios que incrementarían la usabilidad de la aplicación. Estos cambios son:

- **Modificaciones de textos confusos o poco explicativos:** Se han cambiado algunos textos que podrían ser confusos o poco explicativos para los usuarios. Uno de los más destacados fue cambiar “Participantes” por “Enviar solicitudes de unión a la sala” en el cuadro de diálogo para crear una sala. Uno de los mayores problemas detectados es que los usuarios no sabían con seguridad cual era el efecto de añadir usuarios a una sala que estaban creando. No sabían si al crear la sala ya tendrían acceso los otros usuarios, o si les llegaría una solicitud de unión a la sala que tenían que aceptar. Por ello se realizó ese cambio.
- **Iconos de la aplicación:** Un usuario comentó que encontrar los accesos a las distintas vistas podría ser más intuitivo mostrando iconos. Por ello se modificó el menú lateral de la página principal mostrando algunos iconos lo suficientemente representativos para cada acción. En la siguiente imagen se muestra el estado final de ese menú:

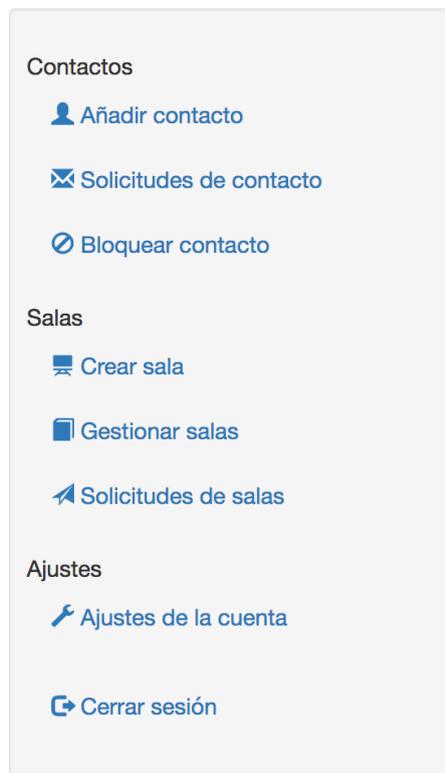


Figura 8.2. Menú lateral página principal

- **Acceso a la página principal:** El acceso a la página principal solo se podía realizar haciendo clic en el logo de la aplicación, y se detectó que a algún usuario le costaba un pequeño periodo de tiempo encontrar la forma de regresar a esa página. Por ello se añadió una opción adicional en la barra de navegación para volver a la página principal. En la siguiente imagen se puede apreciar la barra de navegación con la nueva opción:



Figura 8.3. Barra de navegación

- **Más notificaciones:** Tras la realización de estas pruebas también quedó claro la necesidad de mostrar ciertas notificaciones que no se habían planteado. Como, por ejemplo, cuando otro usuario cambia la URL del video compartido o de la radio.

Finalmente, cabe destacar la pregunta de un usuario de si es posible enviar mensajes a los contactos sin crear una sala. Esta funcionalidad no era un requisito de este proyecto, ya que solo se quería proporcionar comunicaciones a través de salas. No era posible implementar un cambio de tal magnitud en el momento en el usuario realizó estas pruebas, pero sí que se tendrá en cuenta para futuras ampliaciones.

8.3.1.6 Guía de usabilidad

Como parte de las pruebas de accesibilidad, se ha realizado la guía de usabilidad realizada por Yusef Hassan Montero. A continuación, se muestran los resultados.

Criterios	¿Cumplido?
<u>Generales</u>	
¿Cuáles son los objetivos del sitio web? ¿Son concretos y bien definidos? ¿Los contenidos y servicios que ofrece se corresponden con esos objetivos?	SI
¿Tiene una URL correcta, clara y fácil de recordar? ¿Y las URL de sus páginas internas? ¿Son claras y permanentes?	SI
¿Muestra de forma precisa y completa qué contenidos o servicios ofrece realmente el sitio web? El diseño de la página de inicio debe ser diferente al resto de páginas y cumplir la función de 'escaparate' del sitio.	SI
¿La estructura general del sitio web está orientada al usuario? Los sitios web deben estructurarse pensando en el usuario, sus objetivos y necesidades. La estructura interna de la empresa u organización, cómo funciona o se organiza no interesan al usuario.	SI
¿El look & feel general se corresponde con los objetivos, características, contenidos y servicios del sitio web? Ciertas combinaciones de colores ofrecen imágenes más o menos formales, serias o profesionales.	SI
¿Es coherente el diseño general del sitio web? Se debe mantener una coherencia y uniformidad en las estructuras y colores de todas las páginas. Esto sirve para que el usuario no se desoriente en su navegación.	SI
¿Es reconocible el diseño general del sitio web? Cuánto más se parezca el sitio web al resto de sitios web, más fácil será de usar.	SI
¿El sitio web se actualiza periódicamente? ¿Indica cuándo se actualiza? Las fechas que se muestren en la página deben corresponderse con actualizaciones, noticias, eventos...no con la fecha del sistema del usuario.	-
<u>Identidad e Información</u>	

¿Se muestra claramente la identidad de la empresa-sitio a través de todas las páginas?	SI
El Logotipo, ¿es significativo, identificable y suficientemente visible?	SI
El eslogan o <i>tagline</i> , ¿expresa realmente qué es la empresa y qué servicios ofrece?	-
¿Se ofrece algún enlace con información sobre la empresa, sitio web, 'webmaster',...?	SI
¿Se proporciona mecanismos para ponerse en contacto con la empresa? (email, teléfono, dirección postal, fax...)	SI
¿Se proporciona información sobre la protección de datos de carácter personal de los clientes o los derechos de autor de los contenidos del sitio web?	SI
En artículos, noticias, informes... ¿Se muestra claramente información sobre el autor, fuentes y fechas de creación y revisión del documento?	-
<u>Lenguaje y Redacción</u>	
¿El sitio web habla el mismo lenguaje que sus usuarios? Se debe evitar usar un lenguaje corporativista. Así mismo, hay que prestarle especial atención al idioma, y ofrecer versiones del sitio en diferentes idiomas cuando sea necesario.	SI
¿Emplea un lenguaje claro y conciso?	SI
¿Es amigable, familiar y cercano? Es decir, lo contrario a utilizar un lenguaje constantemente imperativo, mensajes crípticos, o tratar con "desprecio" al usuario.	SI
¿1 párrafo = 1 idea? Cada párrafo es un objeto informativo. Transmite ideas, mensajes...Se deben evitar párrafos vacíos o varios mensajes en un mismo párrafo.	-
<u>Rotulado</u>	
Los rótulos, ¿son significativos? Ejemplo: evitar rótulos del tipo "haga clic aquí".	SI

¿Usa rótulos estándar? Siempre que exista un "estándar" comúnmente aceptado para el caso concreto, como "Mapa del Sitio" o "Acerca de...".	-
¿Usa un único sistema de organización, bien definido y claro? No se deben mezclar diferentes. Los sistemas de organización son: alfabético, geográfico, cronológico, temático, orientado a tareas, orientado al público y orientado a metáforas.	-
¿Utiliza un sistema de rotulado controlado y preciso? Por ejemplo, si un enlace tiene el rótulo "Quiénes somos", no puede dirigir a una página cuyo encabezamiento sea "Acerca de"	SI
El título de las páginas, ¿Es correcto? ¿Ha sido planificado? Relacionado con la capacidad para poder buscar y encontrar el sitio <i>web</i> .	SI
<u>Estructura y Navegación</u>	
La estructura de organización y navegación, ¿Es la más adecuada? Hay varios tipos de estructuras: jerárquicas, hipertextual, facetada,...	SI
En el caso de ser puramente hipertextual, ¿Están todos los clúster de nodos comunicados? Aquí se mide la distancia entre nodos.	SI
¿Los enlaces son fácilmente reconocibles como tales? ¿Su caracterización indica su estado (visitados, activos,...)? Los enlaces no sólo deben reconocerse como tales, sino que su caracterización debe indicar su estado, y ser reconocidos como una unidad	SI
En menús de navegación, ¿Se ha controlado el número de elementos y de términos por elemento para no producir sobrecarga memorística? No se deben superar los 7 ± 2 elementos, ni los 2 o, como mucho, 3 términos por elemento.	SI
¿Es predecible la respuesta del sistema antes de hacer clic sobre el enlace? Relacionado con el nivel de significación del rótulo del enlace, aunque también con: el uso de globos de texto, información contextual, la barra de estado del navegador,...	SI
¿Se ha controlado que no haya enlaces que no lleven a ningún sitio? Enlaces que no llevan a ningún sitio: Los enlaces rotos, y los que enlazan con la misma página que se está visualizando (por ejemplo enlaces a la "home" desde la misma página de inicio)	SI

¿Existen elementos de navegación que orienten al usuario acerca de dónde está y cómo deshacer su navegación? ...como <i>breadcrumbs</i> , enlaces a la página de inicio,...recuerde que el logo también es recomendable que enlace con la página de inicio.	SI
Las imágenes enlace, ¿se reconocen como clicables? ¿Incluyen un atributo 'title' describiendo la página de destino? En este sentido, también hay que cuidar que no haya imágenes que parezcan enlaces y en realidad no lo sean.	-
¿Se ha evitado la redundancia de enlaces?	SI
¿Se ha controlado que no haya páginas "huérfanas"? Páginas huérfanas: que aún siendo enlazadas desde otras páginas, éstas no enlacen con ninguna.	SI
<u>Layout de la Página</u>	
¿Se aprovechan las zonas de alta jerarquía informativa de la página para contenidos de mayor relevancia? (como por ejemplo la zona central)	SI
¿Se ha evitado la sobrecarga informativa? Esto se consigue haciendo un uso correcto de colores, efectos tipográficos y agrupaciones para discriminar información. Los grupos diferentes de objetos informativos de una página deben ser 7 ± 2 .	NO
¿Es una interfaz limpia, sin ruido visual?	SI
¿Existen zonas en "blanco" entre los objetos informativos de la página para poder descansar la vista?	SI
¿Se hace un uso correcto del espacio visual de la página? Es decir, que no se desaproveche demasiado espacio con elementos de decoración, o grandes zonas en "blanco", y que no se adjudique demasiado espacio a elementos de menor importancia.	SI
¿Se utiliza correctamente la jerarquía visual para expresar las relaciones del tipo "parte de" entre los elementos de la página? (La jerarquía visual se utiliza para orientar al usuario)	SI
¿Se ha controlado la longitud de página? Se debe evitar en la medida de lo posible el <i>scrolling</i> . Si la página es muy extensa, se debe fraccionar.	SI

<u>Búsqueda (si es necesario, por la extensión del sitio, incorporar un buscador interno)</u>	
¿Se encuentra fácilmente accesible? Es decir: directamente desde la home, y a ser posible desde todas las páginas del sitio, y colocado en la zona superior de la página.	-
¿Es fácilmente reconocible como tal?	-
¿Permite la búsqueda avanzada? (siempre y cuando, por las características del sitio web, fuera de utilidad que la ofreciera)	-
¿Muestra los resultados de la búsqueda de forma comprensible para el usuario?	-
¿La caja de texto es lo suficientemente ancha?	-
¿Asiste al usuario en caso de no poder ofrecer resultados para una consultada dada?	-
<u>Elementos Multimedia</u>	
¿Las fotografías están bien recortadas? ¿Son comprensibles? ¿Se ha cuidado su resolución?	SI
¿Las metáforas visuales son reconocibles y comprensibles por cualquier usuario? (prestar especial atención a usuarios de otros países y culturas)	-
¿El uso de imágenes o animaciones proporciona algún tipo de valor añadido?	-
¿Se ha evitado el uso de animaciones cíclicas?	-
<u>Ayuda</u>	
Si posee una sección de Ayuda, ¿Es verdaderamente necesaria? Siempre que se pueda prescindir de ella simplificando los elementos de navegación e interacción, debe omitirse esta sección.	-
En enlace a la sección de Ayuda, ¿Está colocado en una zona visible y "estándar"? La zona de la página más normal para incluir el enlace a la sección de Ayuda, es la superior derecha.	-
¿Se ofrece ayuda contextual en tareas complejas? (transferencias bancarias, formularios de registro...)	-

Si posee <i>FAQs</i> , ¿Es correcta tanto la elección como la redacción de las preguntas? ¿Y las respuestas?	-
<u>Accesibilidad (debería cubrirse con los test de Accesibilidad posteriores)</u>	
¿El tamaño de fuente se ha definido de forma relativa, o por lo menos, la fuente es lo suficientemente grande como para no dificultar la legibilidad del texto?	SI
¿El tipo de fuente, efectos tipográficos, ancho de línea y alineación empleadas facilitan la lectura?	SI
¿Existe un alto contraste entre el color de fuente y el fondo?	SI
¿Incluyen las imágenes atributos 'alt' que describan su contenido?	SI
¿Es compatible el sitio web con los diferentes navegadores? ¿Se visualiza correctamente con diferentes resoluciones de pantalla? Se debe prestar atención a: <i>JScript</i> , <i>CSS</i> , tablas, fuentes...	SI
¿Puede el usuario disfrutar de todos los contenidos del sitio web sin necesidad de tener que descargar e instalar <i>plugins</i> adicionales?	SI
¿Se ha controlado el peso de la página? Se deben optimizar las imágenes, controlar el tamaño del código <i>JScript</i> ...	SI
¿Se puede imprimir la página sin problemas? Leer en pantalla es molesto, por lo que muchos usuarios preferirán imprimir las páginas para leerlas. Se debe asegurar que se puede imprimir la página (no salen partes cortadas), y que el resultado es legible.	SI
<u>Control y Retroalimentación</u>	
¿Tiene el usuario todo el control sobre el interfaz? Se debe evitar el uso de ventanas pop-up, ventanas que se abren a pantalla completa, banners intrusivos...	SI
¿Se informa constantemente al usuario acerca de lo que está pasando? Si el usuario tiene que esperar hasta que se termine una operación, se debe mostrar un mensaje indicándole y que debe esperar, con el tiempo de espera estimado o una barra de progreso.	-
¿Se informa al usuario de lo que ha pasado? Por ejemplo, cuando un usuario valora un artículo o responde a una encuesta, se le debe informar de que su voto ha sido procesado correctamente.	SI

Quando se produce un error, ¿se informa de forma clara y no alarmista al usuario de lo ocurrido y de cómo solucionar el problema? Siempre es mejor intentar evitar que se produzcan errores a tener que informar al usuario del error.	SI
¿Posee el usuario libertad para actuar? NO restringir la libertad del usuario: Uso de animaciones que no pueden ser "saltadas", páginas en las que desaparecen los botones de navegación, no impida al usuario poder usar el botón derecho de su ratón...	SI
¿Se ha controlado el tiempo de respuesta? Esto tiene que ver con el peso de cada página (accesibilidad) y tiene relación con el tiempo que tarda el servidor en finalizar una tarea y responder. El tiempo máximo que esperará un usuario son 10 segundos	SI
<u>Aclaraciones</u>	
¿Se ha evaluado adecuadamente la orientación del usuario? (Donde estoy, como volver, que he visitado, que va a pasar)	SI
¿Se ha usado correctamente la publicidad?	-

Después de mostrar la guía de usabilidad, quiere hacerse algunas aclaraciones de la mismas:

- La URL de la página corresponde con una gratuita dada por Heroku. Dentro de las opciones disponibles se ha buscado la más simple posible (copernicus-platform.herokuapp.com).
- Las interfaces de la aplicación fueron desarrolladas para adaptarse lo mejor posible a diferentes tamaños de pantallas. Para las pruebas han sido utilizadas pantallas de 13, 17 y 26 pulgadas, y en todas ellas la interfaz se mostraba de manera correcta. También se probó en los navegadores Opera, Firefox y Chrome con éxito. Hay que tener en cuenta que tecnologías como WebRTC no están disponibles para ciertos navegadores.
- En el caso de evitar la redundancia de enlaces, se ha tratado de evitar toda redundancia haciendo que desde la página principal exista solo un enlace para cada vista. Sin embargo, en la barra de navegación se permite acceder a la página principal de tres formas diferentes: pulsando en el logo, pulsando en una opción de la barra de navegación y en un menú desplegable pulsando en el nombre de

perfil. Esto se hizo para reducir los problemas de los usuarios a la hora de regresar a la página principal.

8.3.2 Pruebas de Accesibilidad

A continuación, se detallarán los resultados obtenidos de realizar las pruebas de accesibilidad especificadas en el capítulo de diseño.

8.3.2.1 Accesibilidad para daltónicos

Se ha analizado la aplicación con la herramienta “Sim Daltonism” para comprobar que las tonalidades escogidas fueran las correctas. Se han analizado las interfaces de la página principal y de las salas visualizándolas como aquellas personas con deuteranopía, tritanopía y visión monocromática.

8.3.2.1.1 Deuteranopía

En las siguientes imágenes se muestran la página principal y una sala con el chat de texto y de dibujos visto por una persona con deuteranopía.

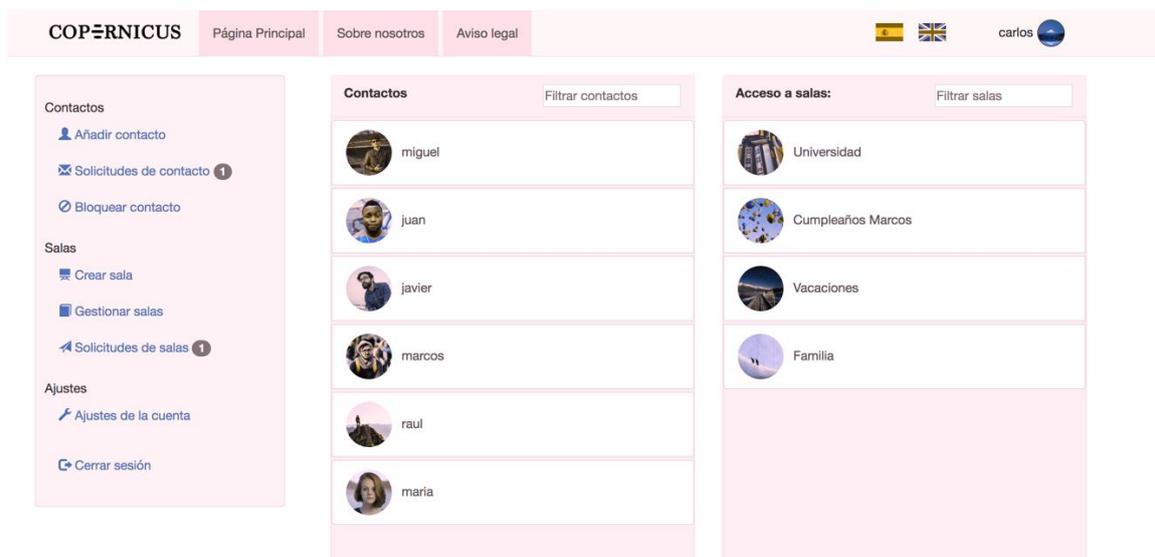


Figura 8.4. Página principal deuteranopía

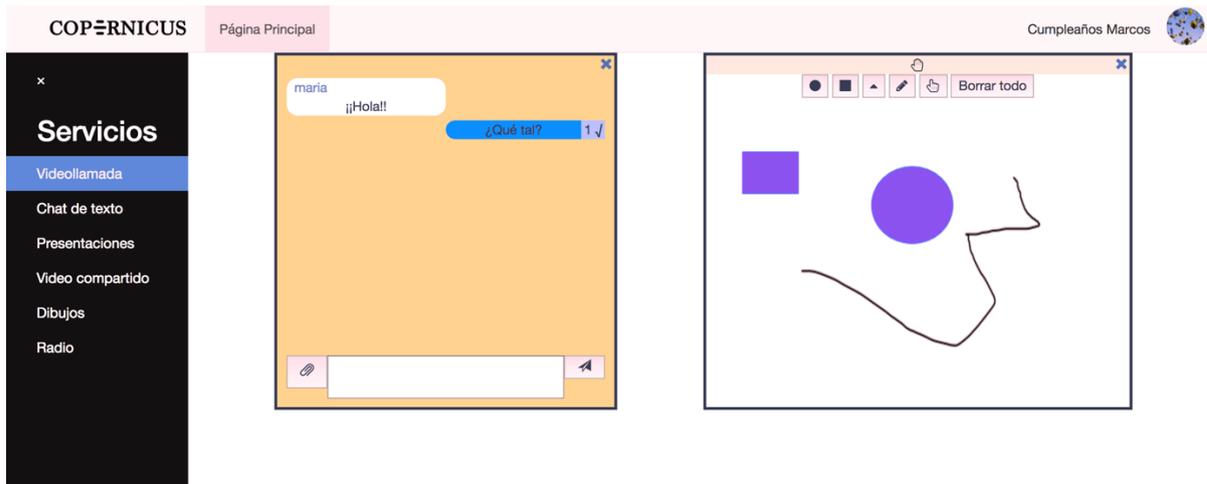


Figura 8.5. Sala deuteranopía

8.3.2.1.2 Tritanopía

En las siguientes imágenes se muestran la página principal y una sala con el chat de texto y de dibujos visto por una persona con tritanopía.

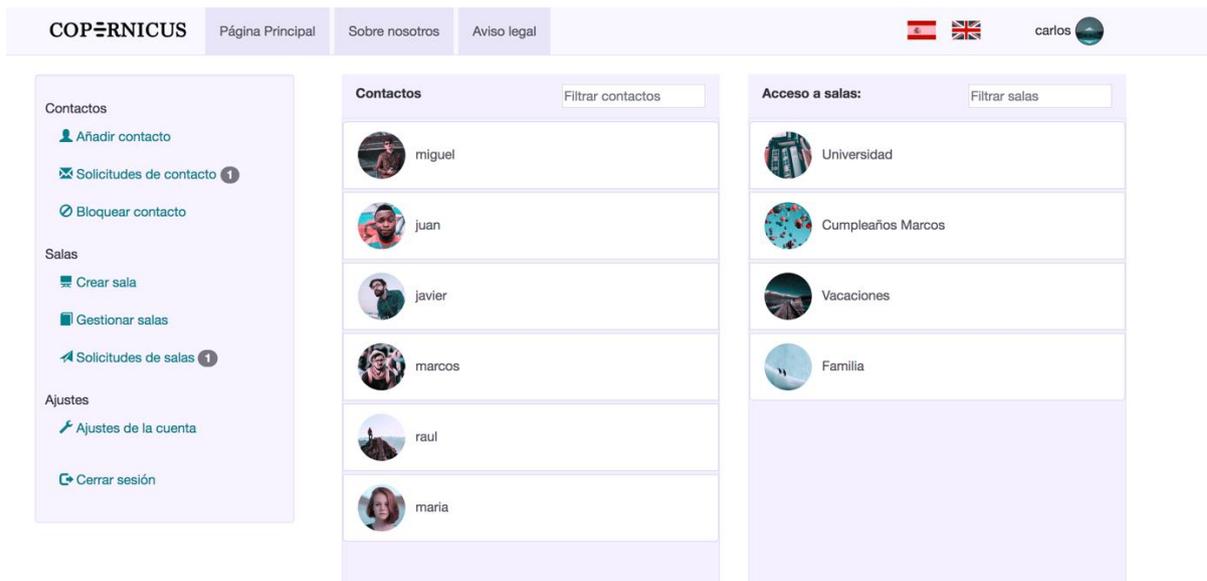


Figura 8.6. Página principal tritanopía

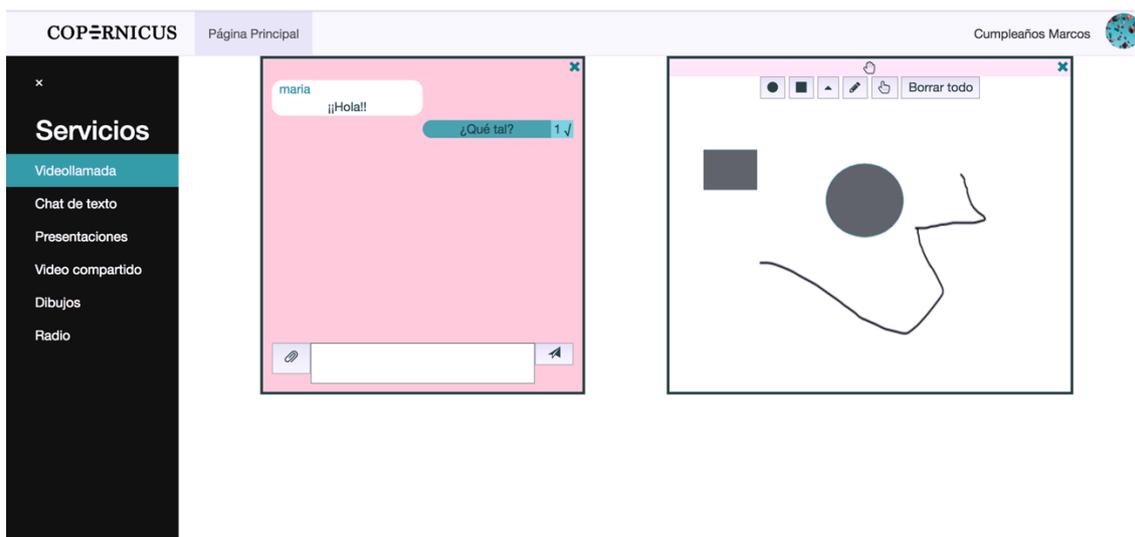


Figura 8.7. Sala tritanopía

8.3.2.1.3 Visión monocromática

En las siguientes imágenes se muestran la página principal y una sala con el chat de texto y de dibujos visto por una persona con visión monocromática.

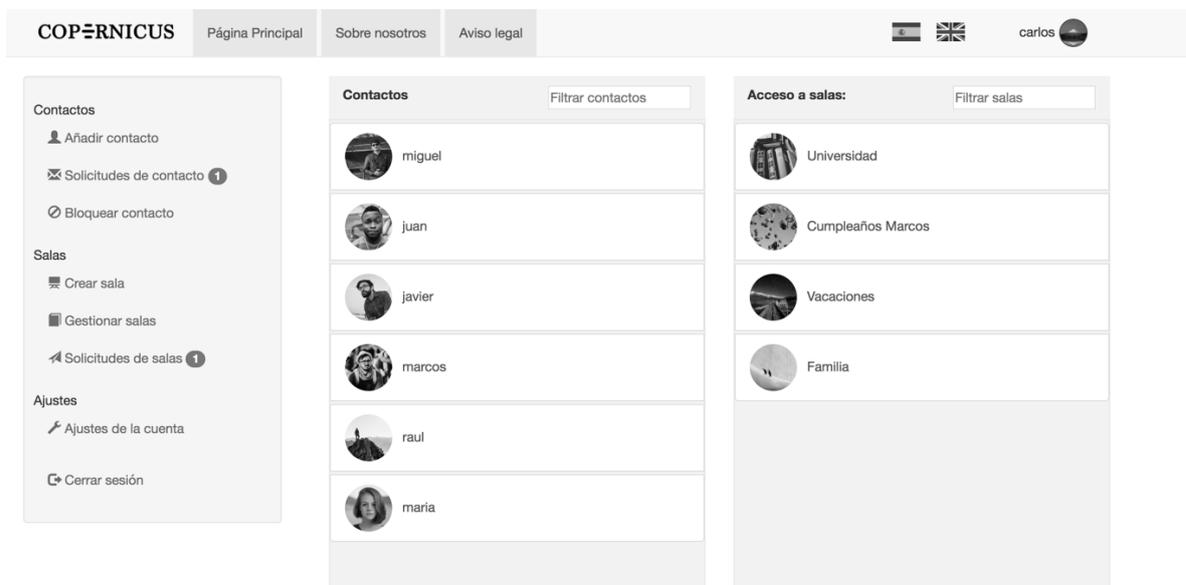


Figura 8.8. Página principal visión monocromática

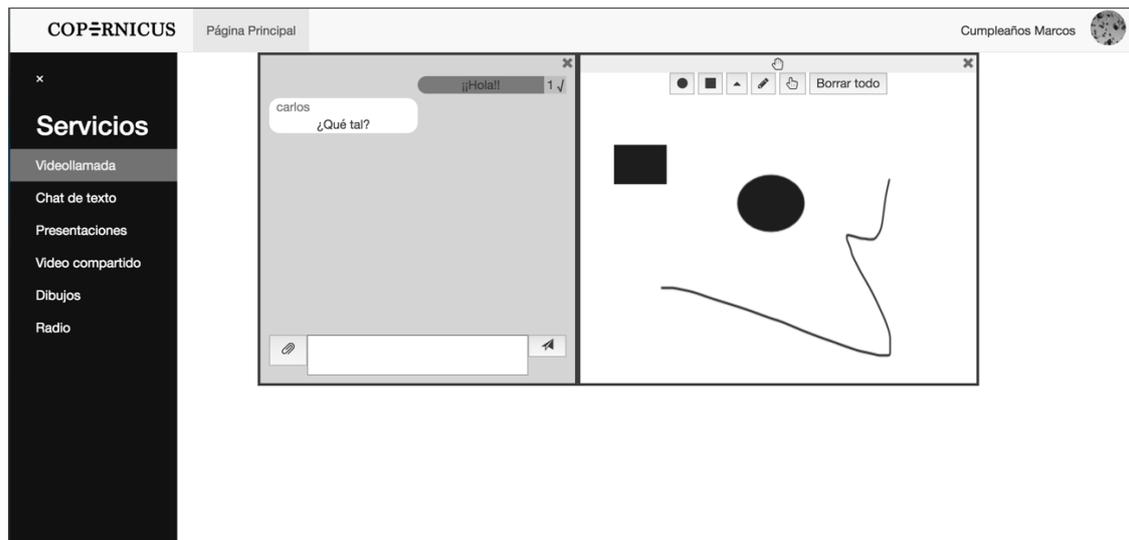


Figura 8.9. Sala visión monocromática

8.3.2.1.4 Conclusiones

Como se puede observar en las anteriores imágenes, no se ha detectado ningún problema en las tonalidades que pueda dificultar el uso de la aplicación por personas con estos tipos de daltonismo.

8.3.2.2 Analizador WCAG 2.0 de TAW

Para completar las pruebas de accesibilidad, se ha utilizado el analizador WCAG 2.0 de TAW.

En este analizador es necesario especificar la URL de nuestra página para que la analice. Como no puede acceder a una aplicación que estamos ejecutando en nuestra máquina local, se ha utilizado la misma versión de la aplicación desplegada en Heroku. El inconveniente de esta prueba es que no se ha podido especificar unas credenciales al analizador, por ello únicamente se pudo evaluar la página de inicio de sesión y de registro.

Se realizó un primer análisis de nivel de análisis AAA, y TAW detectó 56 problemas que necesitaban ser corregidos. Estos problemas eran la falta de textos alternativos para imágenes, la necesidad de utilizar etiquetas junto algunos campos de texto, la falta de un encabezado, etc. Finalmente, se pudieron corregir 40 de estos problemas. Los 16 restantes no se pudieron solucionar porque implicaban problemas relacionados con algunos módulos utilizados, como el de notificaciones o internalización, o porque implicaba cambios no deseados en la interfaz que afectarían a su usabilidad.

8.4 Pruebas de Rendimiento

En esta sección se mostrarán los resultados de las pruebas especificadas en el apartado 6.7.4 Pruebas de Rendimiento. Por supuesto, hay que tener en cuenta que conexiones a internet distintas a la disponible afectarán de manera positiva o negativa a los resultados.

8.4.1 Pruebas con navegador Chrome

8.4.1.1 Prueba inicio de sesión

Para esta prueba se iniciará sesión con una cuenta de usuario que dispone de 7 contactos y 7 salas.

En la siguiente captura se puede observar la herramienta de Google Chrome para estudiar los tiempos de respuesta de la aplicación y el estado de la aplicación en el momento de su estudio.

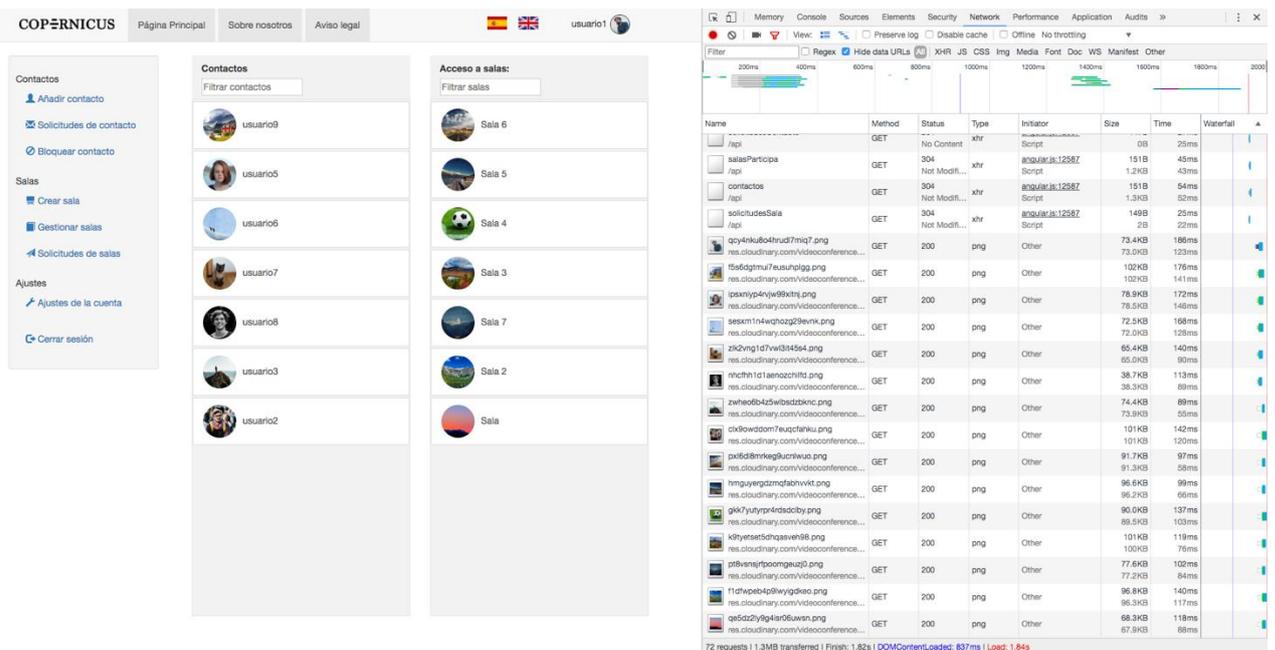


Figura 8.10. Entorno Chrome pruebas rendimiento

En la siguiente imagen se muestra solo la herramienta de Chrome para facilitar su lectura.

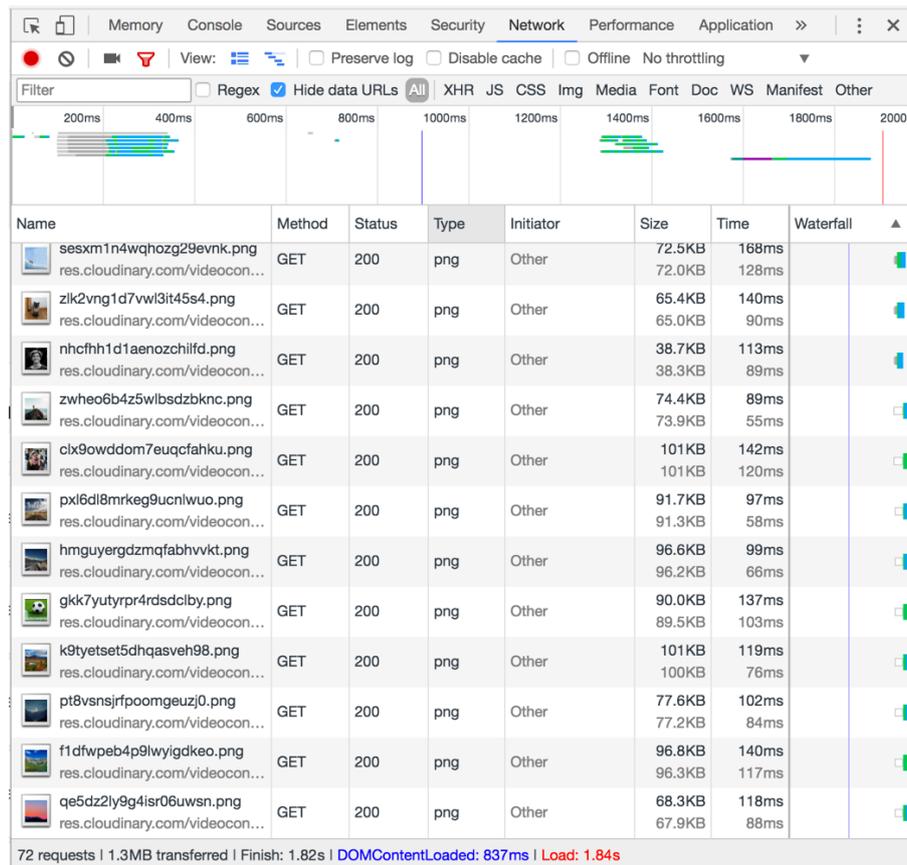


Figura 8.11. Resultados prueba rendimiento iniciar sesión

Como se puede observar, han sido necesarios 1.82 segundos para realizar el inicio de sesión y mostrar la página principal, con un total de 1.3 los MB transferidos. Este es un tiempo aceptable para un inicio de sesión tras borrar la cache y teniendo que recuperar 15 fotografías. Al volver a iniciar sesión nos beneficiamos de la cache del navegador, por lo que no es necesario volver a solicitar las fotografías. Se realizaron pruebas en esas condiciones y los tiempos variaron entre los 1.17 y 1.34 segundos.

Como se puede observar, uno de los puntos más importantes a la hora de disminuir el tiempo de respuesta es una correcta gestión de las fotos de los usuarios y salas. Las fotos utilizadas tenían un tamaño de entre 3 y 7 MB, pero el usuario tiene que recortar una parte de su gusto y se disminuye drásticamente el tamaño hasta conseguir aquellos que se aprecian en la anterior imagen. A pesar de esta disminución, no se aprecia una bajada en la calidad relevante.

8.4.1.2 Prueba registro

En esta prueba se creará una cuenta de usuario utilizando una fotografía de perfil de 5MB y terminará cuando se muestre la página principal del usuario.

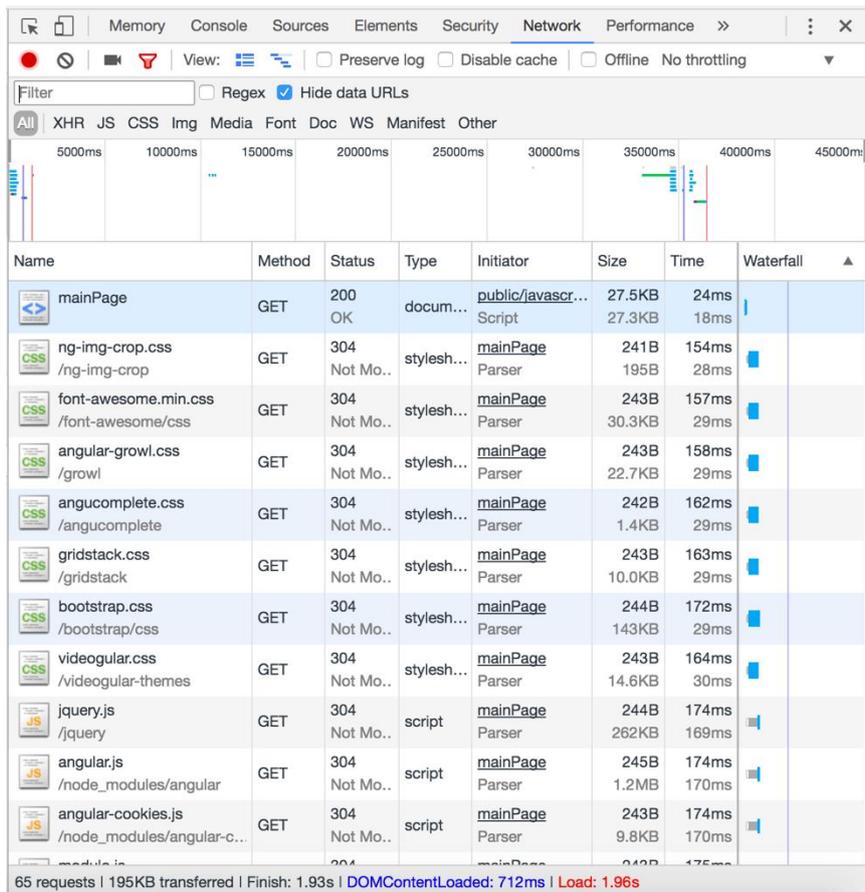


Figura 8.12. Resultados prueba rendimiento registro

Como se puede apreciar en los resultados, llevó 1.93 segundos realizar el registro con las condiciones previamente dichas.

8.4.2 Pruebas con Apache JMeter

Para la realización de estas pruebas, se realizaron 50 peticiones a la API REST en 2 segundos utilizando el software JMeter. De este modo buscamos simular 25 peticiones por segundo. Esta acción será llevada a cabo 20 veces, de modo que se tenga un número de muestras suficiente para la creación de un gráfico con los resultados.

El primer paso fue realizar la configuración del entorno para la ejecución de las pruebas. Esto llevo cierto tiempo debido a que nunca se había utilizado un software similar. Para ello se ha creado un grupo de hilos en JMeter con una petición HTTP que utiliza datos obtenidos de un fichero .csv.

Igual que en las pruebas anteriores, se realizará una prueba para el registro y otra para el inicio de sesión.

8.4.2.1 Prueba inicio de sesión

Como se ha comentado, se ha enviado al servidor una carga de 50 peticiones de inicio de sesión en la aplicación. Estas pruebas incluyen inicios de sesión autorizados y no autorizados, y cada serie de 50 es realizada en tan solo 2 segundos. De este modo, se puede comprobar cómo responde la parte del servidor a varios usuarios intentando iniciar sesión de forma simultánea. En la siguiente captura se puede observar un ejemplo de petición.

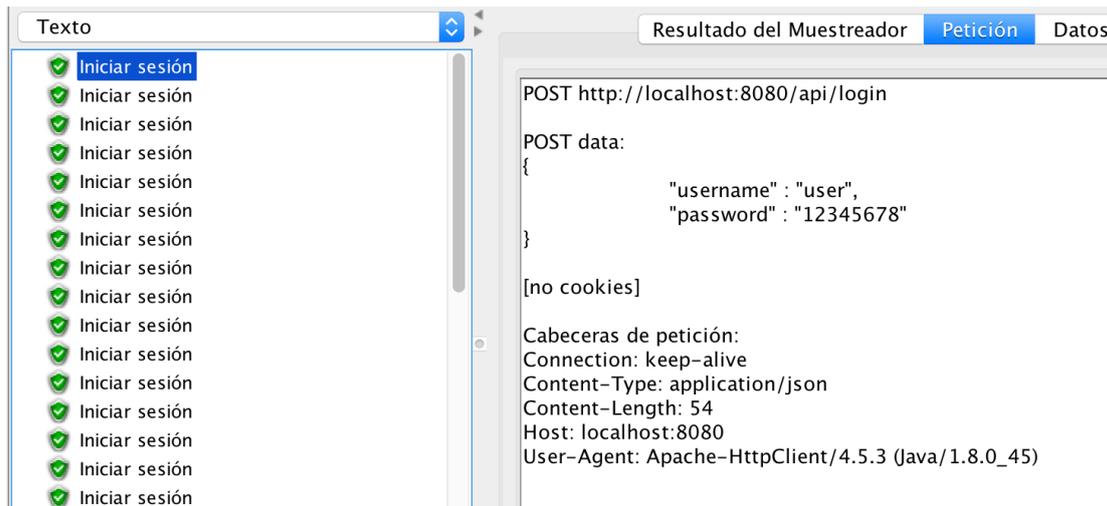


Figura 8.13. Petición iniciar sesión JMeter

También se muestra el gráfico resultante de realizar las 50 peticiones 20 veces, en el que se puede observar el rendimiento, media, mediana y desviación obtenidos.

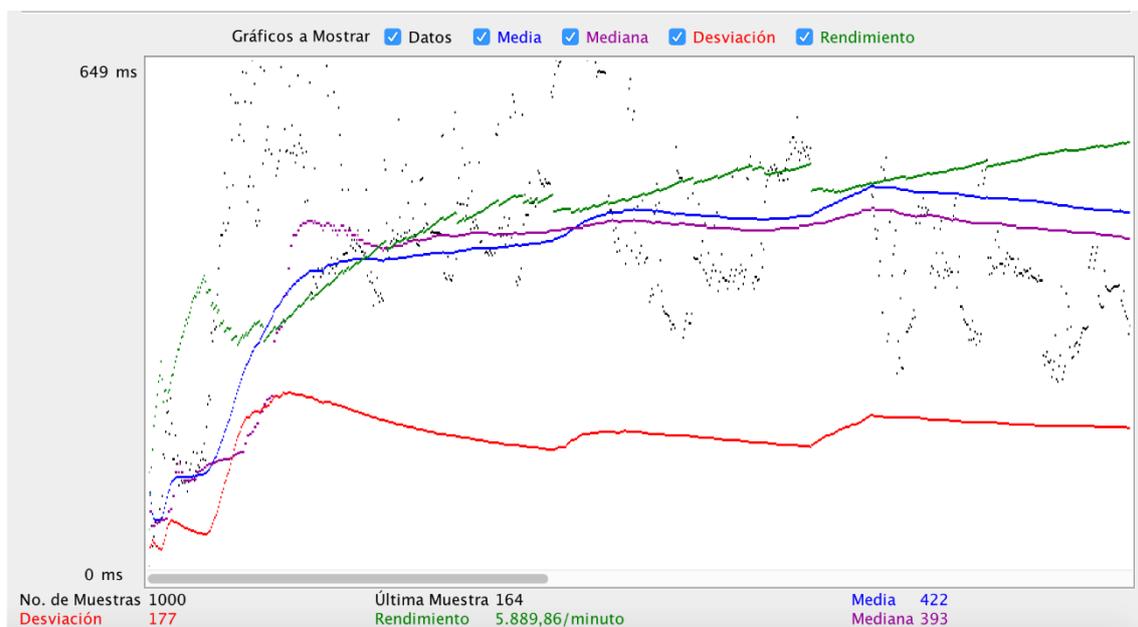


Figura 8.14. Gráfico petición iniciar sesión JMeter

8.4.2.2 Prueba registro

Al igual que en las pruebas de inicio de sesión, se han realizado 50 peticiones de registro a la aplicación en 2 segundos para comprobar cómo responde la parte del servidor a varios usuarios registrándose en la aplicación de forma simultánea. Esto incluye registros válidos y erróneos.

Para ello se ha creado un grupo de hilos en JMeter con una petición HTTP que utiliza datos de registro obtenidos de un fichero .csv. En la siguiente captura se puede observar un ejemplo de petición.

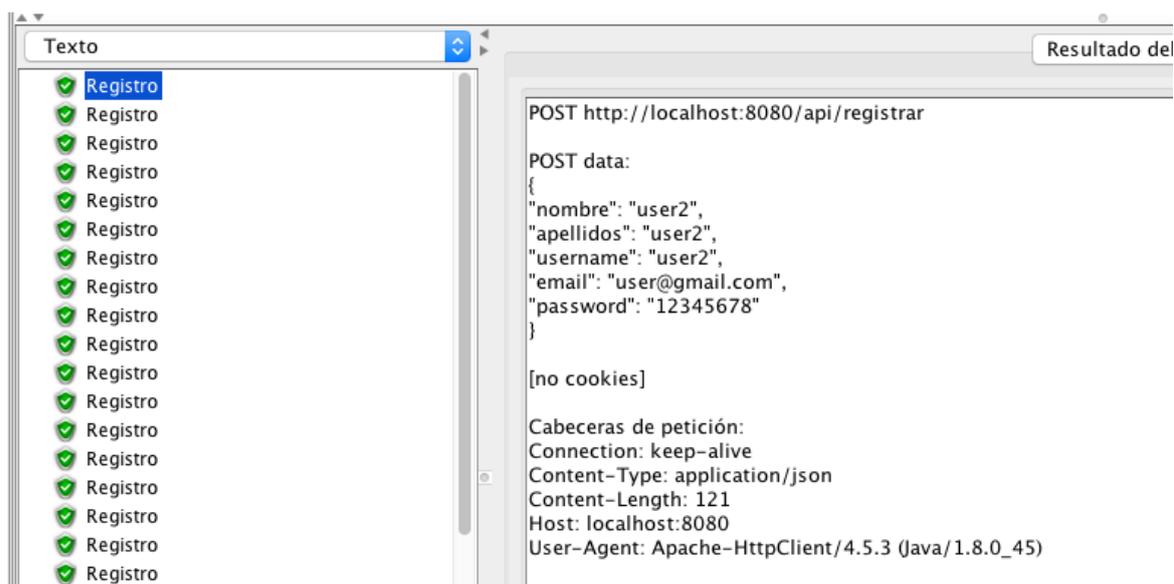


Figura 8.15. Petición registrar JMeter

También se muestra el gráfico resultante de realizar las peticiones. En él se puede observar el rendimiento, media, mediana y desviación resultantes.

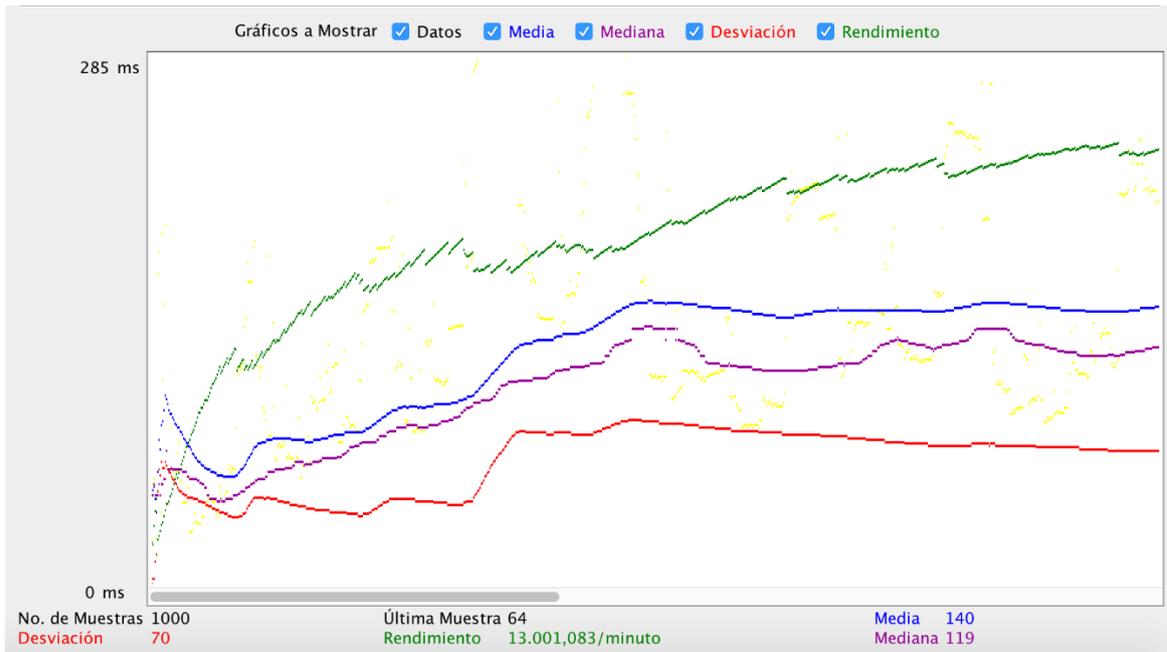


Figura 8.16. Gráfico petición registrar JMeter

Capítulo 9. Manuales del Sistema

9.1 Manual de Instalación

En este manual se explicarán todos los pasos necesarios para realizar una correcta instalación del sistema.

Primero de todo, será necesario descargarse e instalar Node.js, Git y Neo4j Community Edition en la máquina en la que se quiere ejecutar el sistema. A continuación, se muestra la versión que fue utilizada de cada una de estas herramientas y la página desde la que se puede descargar:

- Node.js v6.11.0 LTS: <https://nodejs.org/es/>
- Git 2.13.2: <https://git-scm.com/downloads>
- Neo4j 3.2.1: <https://neo4j.com/download/community-edition/>

Durante la instalación no es necesario especificar ninguna opción en concreto para estas herramientas, por lo que se puede instalar dejando definidos los valores por defecto.

Una vez realizada la descarga e instalación de estas herramientas, procederemos a preparar la base de datos. Para ello ejecutaremos el programa “Neo4j Community Edition”, si no lo teníamos ejecutado ya, y arrancaremos la base de datos utilizando el botón “Start” de la interfaz que aparece. En cuanto aparezca el estado en verde diciendo “Neo4j is ready. Browse to <http://127.0.0.1:7474>”, la base de datos ya estará arrancada.

Tras esto, nos dirigiremos a la dirección “localhost:7474” en un navegador para acceder a la interfaz de Neo4j. Aquí haremos los siguientes pasos:

1. Deberemos introducir unas credenciales y pulsar en “Connect”. Por defecto son el usuario “neo4j” y la contraseña “neo4j” ambos sin las comillas.
2. Una vez introducidas las credenciales, nos pedirá cambiar la contraseña. Se establecerá la contraseña “thesis” sin comillas, ya que es la contraseña definida en el fichero con las variables de entorno de la aplicación.

Tras estos pasos la base de datos estará lista para ser utilizada.

Ahora procederemos a instalar Copernicus. El primer paso será abrir una terminal o el símbolo del sistema en la carpeta “Copernicus/aplicacion” que se encuentra en los archivos adjuntos. Una vez hecho esto instalaremos las dependencias de la aplicación utilizando el comando “npm install”. Si ocurre algún error durante la instalación, puede ser que npm no esté correctamente actualizado. En ese caso, debemos ejecutar la orden “npm install npm -g” para actualizarlo. El proceso de instalación de dependencias puede durar unos minutos.

9.2 Manual de Ejecución

En este manual se especificarán los pasos para ejecutar la aplicación, y vienen justamente después de realizar los pasos del manual de instalación. Por lo tanto, en este punto la base de datos Neo4j debería de estar ejecutándose tal y como se dejó en ese manual.

Una vez terminada la instalación de dependencias, se ejecutará la orden “npm start” en la carpeta “Copernicus/aplicacion”. Esta orden ejecutará la aplicación, la cual estará lista para su uso cuando aparezcan en pantalla “Servidor de WebSockets escuchando en el puerto: 8080”. En el caso de utilizar el sistema operativo Windows, es posible que aparezca una alerta de seguridad en la que se dice que el Firewall bloqueó algunas características de Node.js. En ese caso, debemos pulsar en “Permitir acceso”.

Ahora solo tenemos que dirigirnos a la dirección “localhost:8080” en un navegador. Este navegador, tal y como se dijo anteriormente en esta documentación, deberá soportar WebRTC para poder utilizar todas las funcionalidades de la aplicación. Por ello, deberá ser una versión actualizada de navegadores como Firefox o Google Chrome.

Se han reportado casos en sistemas operativos Windows en los que, al ejecutar la aplicación tras instalar las dependencias, esta no respondía correctamente a las peticiones realizadas desde el navegador. Para solucionar este problema se utilizó la combinación de teclas *ctrl* + *c* una sola vez en el símbolo del sistema, o se puede utilizar un nuevo símbolo del sistema distinto al usado para la instalación de dependencias.

Para detener la ejecución de la aplicación, únicamente deberemos de parar el proceso que se está ejecutando en la terminal o símbolo del sistema.

Hay que tener en cuenta que, aunque la aplicación se ejecute en local, es necesario disponer de conexión a internet. Esto es debido a que se utilizó un servidor remoto para almacenar y recuperar fotografías.

También es posible probar la aplicación dirigiéndose a la dirección <https://copernicus-platform.herokuapp.com/>. Cabe destacar que se utiliza un servidor gratuito de Heroku por lo que el rendimiento y respuesta de la aplicación puede no ser óptimo.

9.3 Manual de Usuario

En este manual se explicarán todas aquellas funciones que proporciona Copernicus a sus usuarios.

9.3.1 Modificar idioma de la aplicación

Copernicus soporta dos idiomas, inglés y español. Por lo que, para poder utilizar correctamente la aplicación, un primer paso sería definir el idioma que se desea utilizar.

Esto es muy sencillo, lo único que debemos hacer es seleccionar la bandera de España o Reino Unido en la barra de navegación dependiendo del idioma que deseemos. Estas banderas pueden ser accedidas desde cualquier parte de la aplicación exceptuando las salas. En la siguiente captura esta señalizada la localización de estas banderas en la pantalla de inicio de sesión y registro.

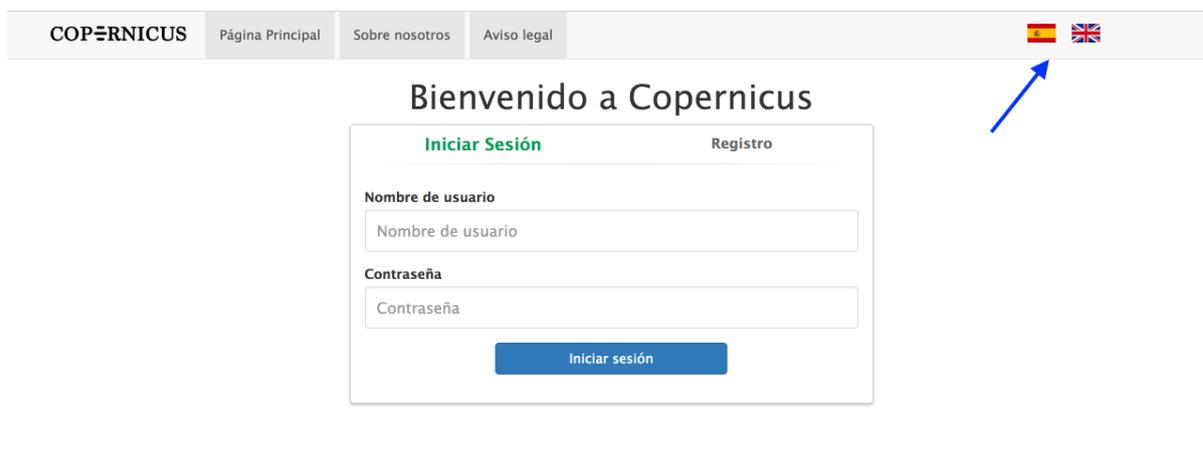


Figura 9.1. Manual usuario - Localización banderas idiomas

9.3.2 Iniciar sesión y registrarse en la aplicación

La primera pantalla que nos encontramos al acceder a la aplicación es la pantalla para iniciar sesión y registrarse en la aplicación. Concretamente, veremos el formulario para iniciar sesión. Simplemente tenemos que introducir el nombre de usuario y contraseña que hemos utilizado al registrarnos en la aplicación.

COPERNICUS [Página Principal](#) [Sobre nosotros](#) [Aviso legal](#)  

Bienvenido a Copernicus

Iniciar Sesión Registro

Nombre de usuario

Contraseña

Figura 9.2. Manual de usuario - Iniciar sesión

En el caso de que no nos hayamos registrado en la aplicación. Pulsaremos en “Registro” para acceder al formulario para crear nuestra cuenta.

COPERNICUS [Página Principal](#) [Sobre nosotros](#) [Aviso legal](#)  

Bienvenido a Copernicus

Iniciar Sesión **Registro**

Nombre

Apellidos

Nombre de usuario

Correo electrónico

Contraseña

Confirmar contraseña

Selecciona una foto de perfil
 Ningún archivo seleccionado

Figura 9.3. Manual de usuario - Registro

Como podemos ver en la captura anterior, es obligatorio introducir un nombre, nombre de usuario, correo electrónico y contraseña. Es necesario introducir la contraseña dos veces para evitar confundirnos al establecerla. Opcionalmente se puede introducir

apellidos y una foto de perfil. En el caso de no indicar foto de perfil se utilizará una por defecto.

Hay que tener en cuenta que los datos introducidos deben cumplir con unos requisitos:

- El nombre no puede ser menor que 3 caracteres ni mayor que 15.
- El apellido no puede ser mayor que 35 caracteres.
- El nombre de usuario no puede ser menor que 3 caracteres ni mayor que 15.
- Se comprobará que el email tenga un formato válido.
- La contraseña no puede ser menor que 8 caracteres.
- La foto de perfil no puede ser mayor que 8MB.

9.3.3 Añadir contactos

Una vez creada nuestra cuenta se mostrará una página como la siguiente.

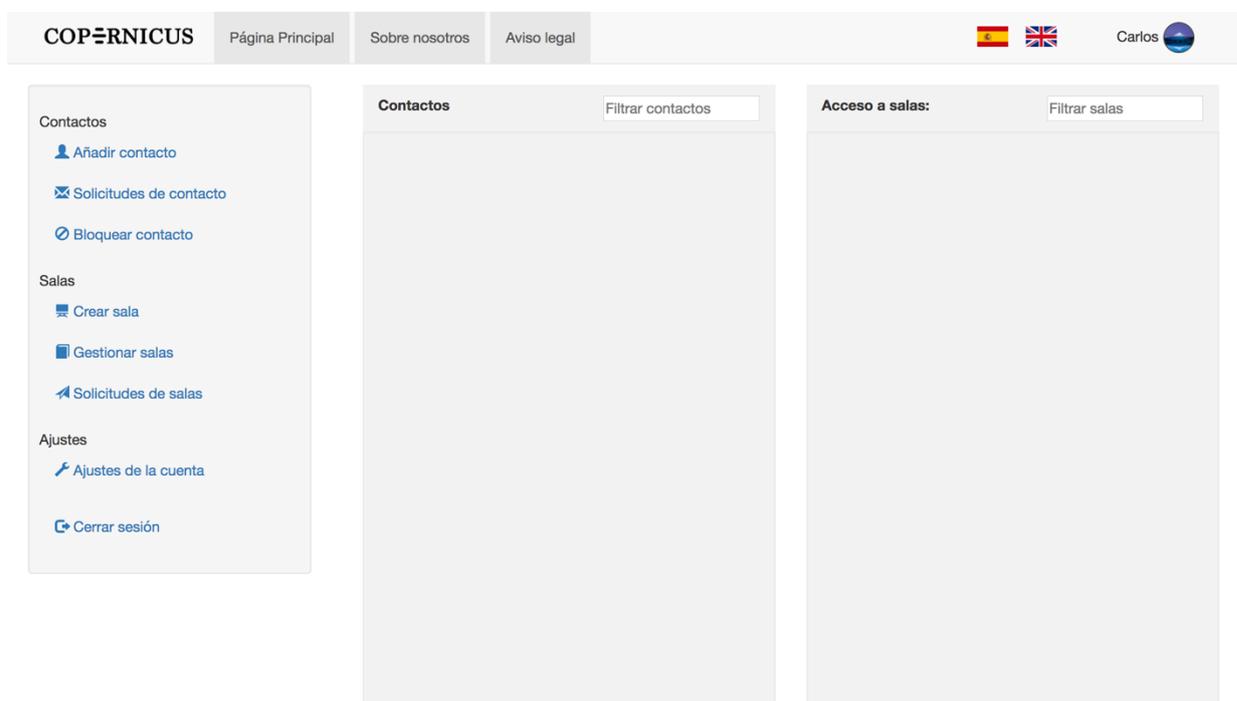


Figura 9.4. Manual de usuario - Página principal

Como podemos ver, no disponemos de ningún contacto o sala, por lo que el siguiente paso será añadir contactos a nuestro listado.

Si nos dirigimos al enlace “Añadir contacto”, se mostrará un cuadro de diálogo en el que podremos especificar un usuario y mensaje para enviar una solicitud de contacto a otro usuario. Para seleccionar el usuario, escribiremos su nombre de usuario en el campo de texto y lo seleccionaremos de un listado que aparece. A continuación, se muestra el diálogo en cuestión y el listado para seleccionar usuarios.

Figura 9.5. Manual de usuario - Añadir contacto

Figura 9.6. Manual de usuario - Listado para seleccionar usuario

El listado anterior será utilizado en todas las partes de la aplicación en las que se solicite introducir un nombre de usuario.

9.3.4 Aceptar o ignorar solicitudes de contacto

En el apartado anterior hemos aprendido como enviar solicitudes de contacto. Ahora es necesario explicar cómo son aceptadas o ignoradas.

En el caso de que tengamos alguna solicitud de contacto, se mostrará una notificación como la de la siguiente captura.



Figura 9.7. Manual de usuario - Notificación solicitudes de contacto

Si creemos que hemos recibido una solicitud, pero no la vemos, será necesario refrescar la página.

Pulsando en el enlace de “Solicitudes de contacto” aparecerá un cuadro de diálogo como el que se muestra a continuación. En él podremos ver nuestras solicitudes de contacto y aceptarlas o ignorarlas.



Figura 9.8. Manual de usuario - Solicitudes de contacto

Cuando hallamos terminado de gestionar nuestras solicitudes, pulsaremos en el botón de cerrar el diálogo. Esto actualizará la página principal y nos mostrará a los nuevos usuarios aceptados en nuestro listado de usuarios. Un ejemplo de listado de contacto es el de la siguiente fotografía.

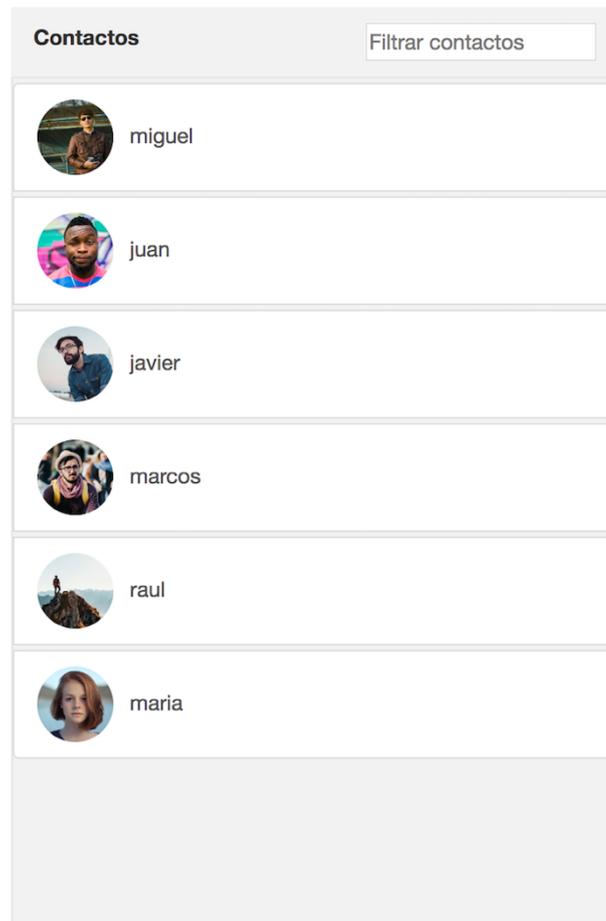


Figura 9.9. Manual de usuario - Listado de contactos

En este listado podremos seleccionar a uno de nuestros contactos para acceder a un cuadro de diálogo con su información. También nos es posible introducir texto en el campo “Filtrar contactos” para mostrar solo aquellos que contengan el texto especificado.

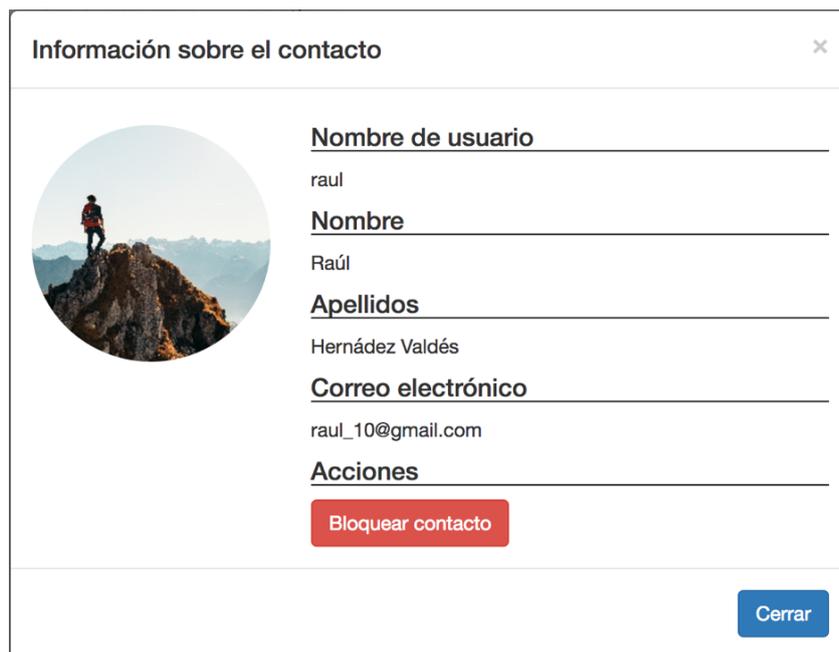


Figura 9.10. Manual de usuario - Información sobre el contacto

9.3.5 Bloquear contactos

Una vez que tenemos contactos, es posible que deseemos bloquear alguno por algún motivo. Para ello utilizamos el enlace del menú lateral “Bloquear contacto”, que mostrará el siguiente diálogo.

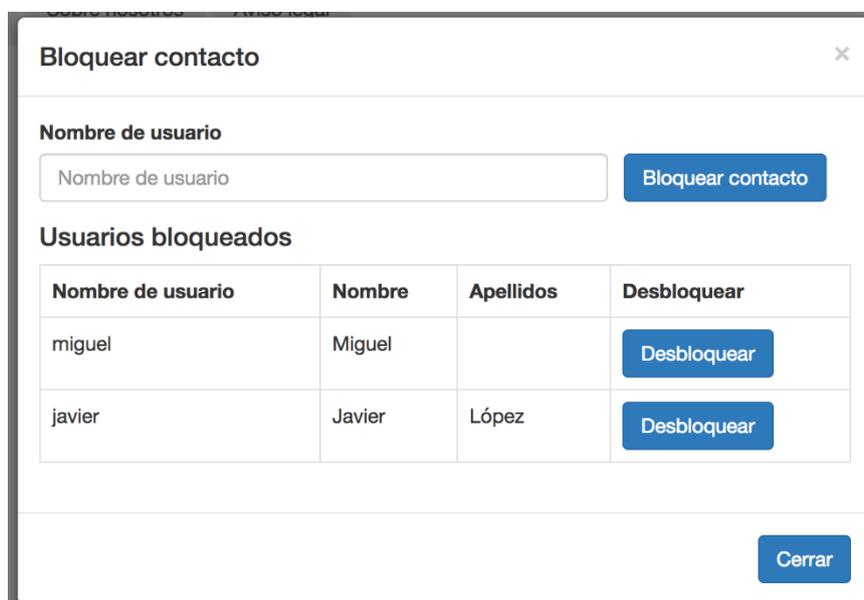


Figura 9.11. Manual de usuario - Bloquear contacto

En ese dialogo especificaremos un nombre usuario (que seleccionaremos de un listado tal y como pasaba al mandar solicitudes de contacto) y pulsaremos en el botón para bloquear

el contacto. En el caso de que nos arrepintamos, siempre podemos desbloquear el contacto en el mismo diálogo utilizando los botones “Desbloquear”.

9.3.6 Crear salas

Ya tenemos una cuenta y usuarios, el siguiente paso será crear nuestras salas para comunicarnos con nuestros contactos. Para ello nos dirigimos a “Crear sala” en la página principal, lo que nos mostrará un diálogo como el siguiente.

Crear sala
×

Datos de la sala

Nombre

Descripción

Enviar solicitudes de unión a la sala

Nombre de usuario

 Añadir contacto

Nombre de usuario	Nombre	Apellidos	Permisos	Eliminar
maría	María	González	Moderador ▾	Eliminar
raul	Raúl	Hernández Valdés	Miembro ▾	Eliminar

Foto de la sala

Seleccionar archivo
balloons-2074347_1280.jpg



Crear sala
Cerrar

Figura 9.12. Manual de usuario - Crear sala

En el anterior ejemplo ya se especificó un nombre, descripción, contactos a los que se enviarán solicitudes de unión y sus permisos y una foto para la sala. Sin embargo, de estos datos únicamente es obligatorio el nombre.

Hay que tener en cuenta que los datos introducidos deben cumplir con unos requisitos:

- El nombre de la sala no puede ser mayor que 50 caracteres.
- La descripción de la sala no puede ser mayor que 200 caracteres.
- Solamente pueden enviarse 3 solicitudes de contacto.
- La foto de la sala no puede ser mayor que 8MB.

Como se puede ver, se han especificado contactos a los que se les enviará una solicitud de unión a esa sala y unos permisos. Estos permisos son los que se otorgaran al usuario cuando este acepte la solicitud y existen tres tipos de ellos. Con todos los permisos se pueden utilizar las salas en igualdad de condiciones, sin embargo, los usuarios tendrán más o menos restricciones a la hora de gestionar la sala. Estos permisos serán explicados en el apartado [9.3.8 Gestionar salas](#).

Existe un límite a la hora de añadir usuarios a la sala, de modo que solamente podemos enviar 3 solicitudes de unión cuando creamos la sala. De este modo la sala tendrá 4 participantes junto al administrador (persona que crea la sala). Si nos hemos equivocado a la hora de seleccionar los contactos a los que enviar solicitudes de unión, podemos utilizar los botones “Eliminar”.

Una vez se ha especificado el nombre y los demás datos deseados, pulsamos en el botón “Crear sala” y se mostrará la página principal con nuestra nueva sala en el listado de salas.

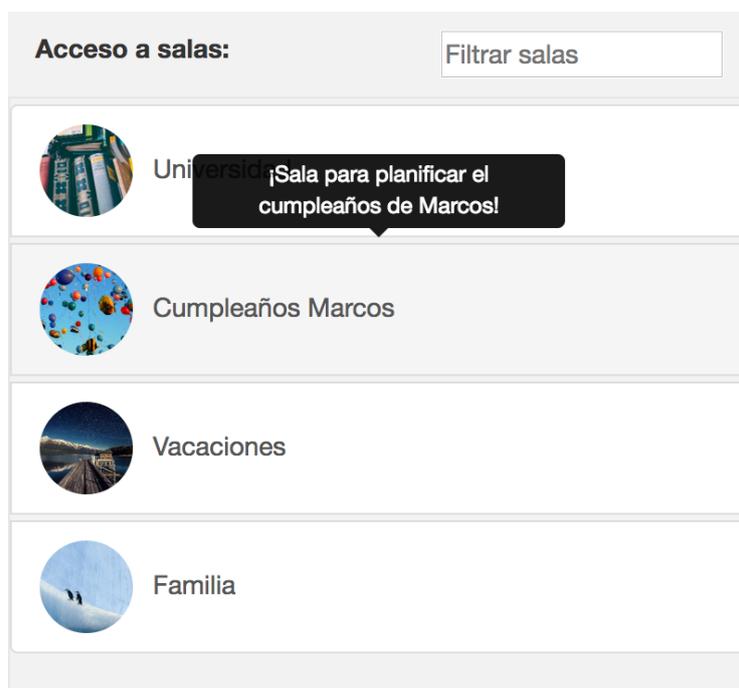


Figura 9.13. Manual de usuario - Listado de salas

En la anterior captura se muestra un ejemplo de un listado de salas. Pulsando sobre una sala podremos acceder a ella, mientras que si dejamos el puntero encima podremos ver su descripción. Al igual que ocurría con el listado de contactos, podemos filtrar las salas que se muestran introduciendo un texto en el campo “Filtrar salas”.

9.3.7 Aceptar o ignorar solicitudes de unión a salas

Una vez mostrado como se crea una sala, se explicará cómo aceptar una solicitud de unión.

Al igual que con las solicitudes de contacto, en el caso de que dispongamos de solicitudes de salas aparecerá una notificación como la que se muestra a continuación.



Figura 9.14. Manual de usuario - Notificación solicitudes de salas

Pulsando sobre “Solicitudes de salas” se mostrará un cuadro de diálogo en el que podremos aceptar o ignorar las solicitudes de unión a salas recibidas.

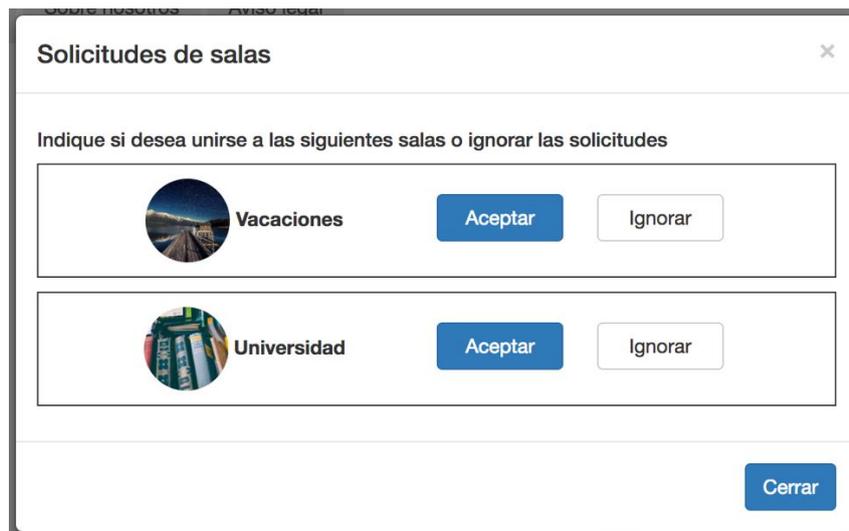


Figura 9.15. Manual de usuario - Solicitudes de salas

9.3.8 Gestionar salas

Una vez una sala es creada, nos es posible gestionar sus datos, solicitudes, participantes y eliminarla o salir de ella accediendo a una página a través del enlace “Gestionar salas”.

En esta página se muestran las salas en tres listados dependiendo de nuestros permisos en ellas.

The screenshot shows the 'Gestionar salas' interface. At the top, there's a navigation bar with 'COPERNICUS', 'Página Principal', 'Sobre nosotros', and 'Aviso legal'. The user 'maria' is logged in. The main content is divided into three sections based on permissions:

- Salas en las que es administrador:** Shows a list with 'Amigos' selected.
- Salas en las que es moderador:** Shows a list with 'Familia' selected.
- Salas en las que es miembro:** Shows a list with 'Vacaciones' and 'Cumpleaños Miguel' selected.

The 'Información de la sala' section for 'Amigos' includes:

- Nombre:** Amigos
- Descripción:** (Empty text area)
- Foto de la sala:** Seleccionar archivo (Image of a red house)
- Actualizar datos** (Button)

The 'Usuarios de la sala' section has a 'Nombre de usuario' input field and an 'Añadir usuario' button.

The 'Solicitudes enviadas' section contains a table with the following structure:

Nombre de usuario	Nombre	Apellidos	Permisos	Estado	Acciones
...	Cela	Villa	...	Todavía se necesita la solicitud	...

Figura 9.16. Manual de usuario - Gestionar salas

Existen tres tipos de permisos que puede poseer un usuario respecto a una sala. A continuación, se explica que pueden realizar los usuarios en esta página en función de sus permisos:

- **Administrador:** Solo existe un administrador por sala y es el usuario que la creó. Este usuario puede modificar los datos de la sala, enviar o eliminar solicitudes de unión, cambiar los permisos de las solicitudes o de cualquier usuario que este participando en la sala, eliminar cualquier usuario que participe en la sala y eliminar la sala.
- **Moderador:** Tiene los mismos permisos que el administrador exceptuando que no puede cambiar los permisos de otros moderadores que estén participando en la sala ni puede eliminarlos. Tampoco puede eliminar la sala, pero sí que puede salir de ella.
- **Miembro:** Este usuario solo puede ver quien participa en la sala y sus permisos o salir de la misma.

Para poder realizar estas funciones, se tiene que seleccionar cualquier sala de los listados de la izquierda y se mostrarán las acciones que se pueden realizar en la derecha.

9.3.9 Utilización de las salas y sus servicios

En este apartado se explicará cómo utilizar las salas y los servicios disponibles.

Primero de todo se mostrará cómo se encuentra la sala una vez esta es accedida. Como podemos ver en la siguiente imagen, tenemos dos posibilidades. O abrir el menú de servicios o el de participantes. También podemos ocultar el menú con la flecha localizada encima.

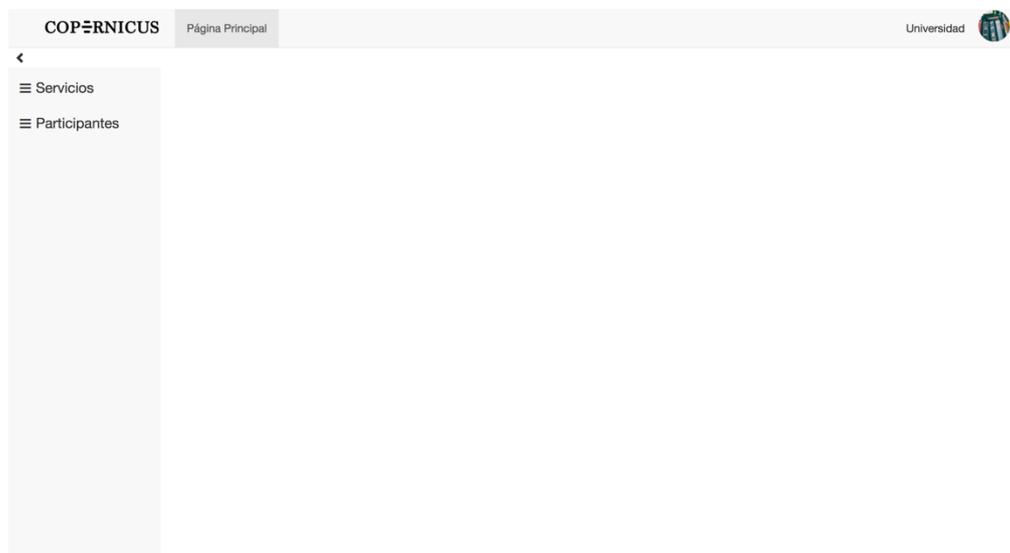


Figura 9.17. Manual de usuario - Sala

En el menú “Servicios” se pueden seleccionar aquellos servicios que queramos abrir.

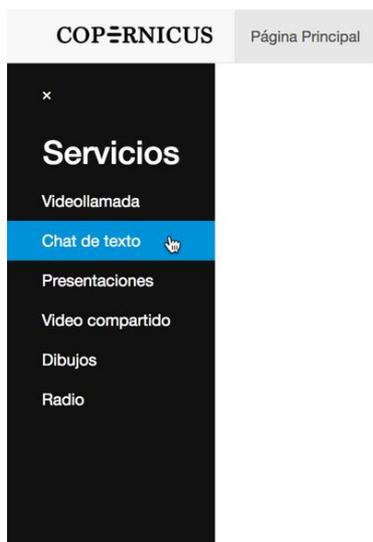


Figura 9.18. Manual de usuario - Menú de servicios

En el menú “Participantes” se muestran los usuarios actualmente conectados en la sala.

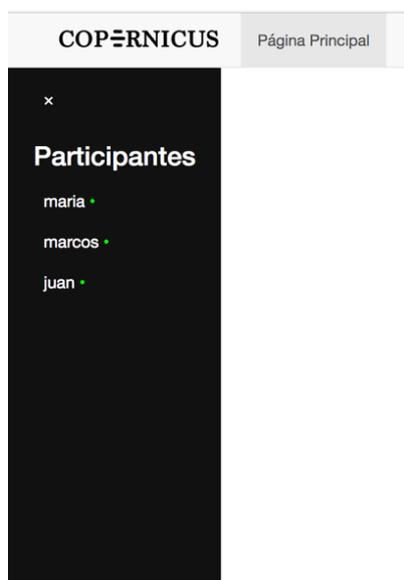


Figura 9.19. Manual de usuario - Menú de participantes

9.3.9.1 Introducción a los servicios

Los servicios son aquellos que proporcionan las comunicaciones en tiempo real entre los participantes de una sala. En esta sección se explicarán algunas características comunes de estos servicios para después explicarlos uno a uno.

Todos los servicios pueden ser abiertos, redimensionados (excepto la radio), movidos y cerrados. Para abrirlos, como ya se explicó, se tiene que abrir el menú lateral de “Servicios” y pulsar en el deseado. Una vez el servicio está abierto podemos moverlo pulsando y arrastrando en la zona del servicio con el siguiente símbolo y color.

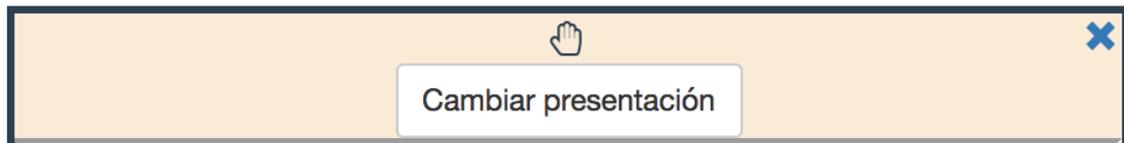


Figura 9.20. Manual de usuario - Desplazar servicio

En el caso de que tal zona no exista, el servicio puede ser desplazado pulsando en cualquier parte del mismo.

Para redimensionar el servicio, pulsaremos sobre unas flechas en la parte inferior derecha del servicio y moveremos el puntero. Estas flechas solo serán visibles al posicionar el puntero en la zona indicada.



Figura 9.21. Manual de usuario - Flechas redimensionar

Finalmente, para cerrar un servicio simplemente tenemos que pulsar en la "X" de la zona superior derecha.



Figura 9.22. Manual de usuario - Símbolo cerrar servicio

9.3.9.2 Videollamada

El servicio de videollamadas permite comunicarnos mediante video con todas aquellas personas conectadas a este servicio. También facilita 3 controles para activar o desactivar el micrófono, los altavoces o la transmisión de nuestro video.

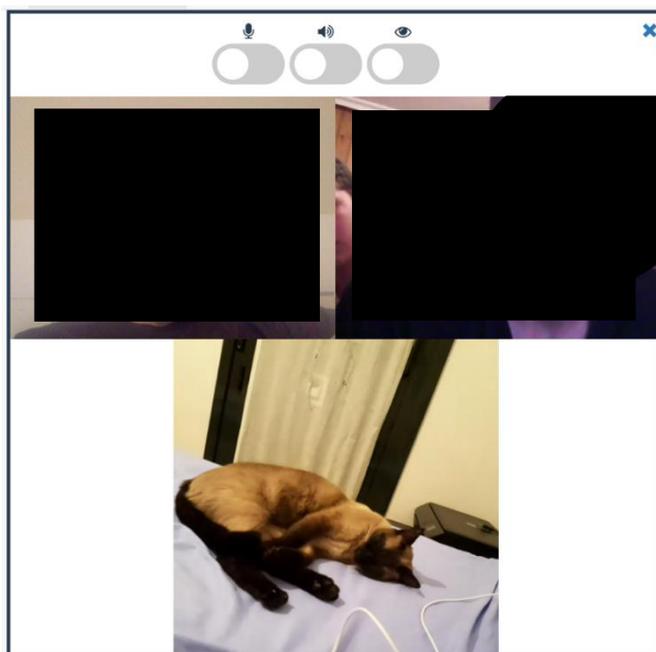


Figura 9.23. Manual de usuario - Videollamada

9.3.9.3 Chat de texto

En el chat de texto podremos comunicarnos con otros usuarios utilizando mensajes de texto o ficheros. Para ello se proporciona un botón para mandar mensajes (también es posible mandarlos pulsando la tecla Enter) y otro botón para seleccionar los ficheros a mandar.

En el chat se mostrarán los mensajes enviados y el número de personas que los vieron (para los ficheros se muestran el número de personas a través de una notificación), y también los mensajes de otros usuarios junto a su nombre.



Figura 9.24. Manual de usuario - Chat de texto

9.3.9.4 Presentaciones

Con el servicio de presentaciones se pueden compartir presentaciones HTML o PDF con otros usuarios de la sala. Simplemente pulsamos en el botón “Cambiar presentación” y seleccionamos el fichero que deseamos utilizar.

En el caso de utilizar una presentación HTML, se muestran unos controles en la parte inferior derecha.

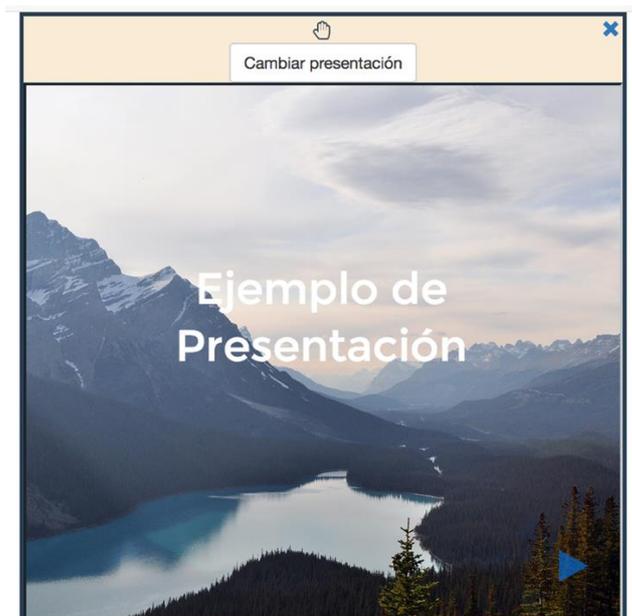


Figura 9.25. Manual de usuario - Presentación HTML

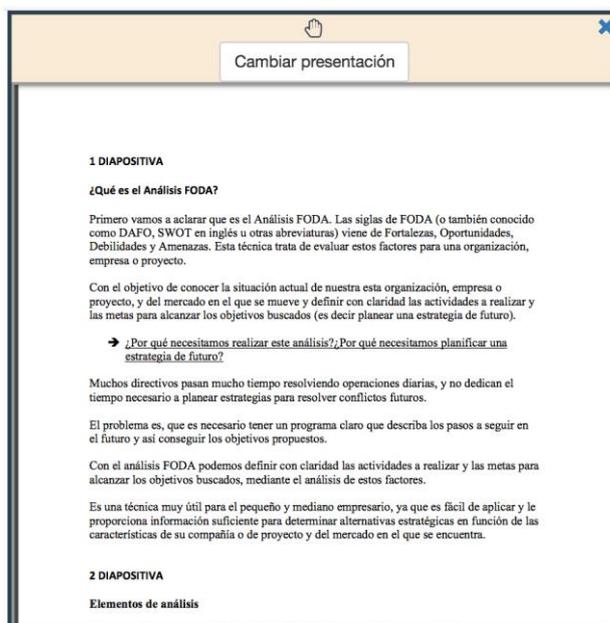


Figura 9.26. Manual de usuario - Presentación PDF

Sabremos cuando otro usuario cambia la presentación a través de una notificación.

9.3.9.5 Video compartido

El servicio de video compartido permite que podamos compartir las URLs de videos de YouTube o alojados en otros servidores con otros usuarios de la sala, y verlos en la misma plataforma. Para compartir un video, simplemente especificamos su URL en el campo facilitado y pulsamos en el botón “Cambiar”.

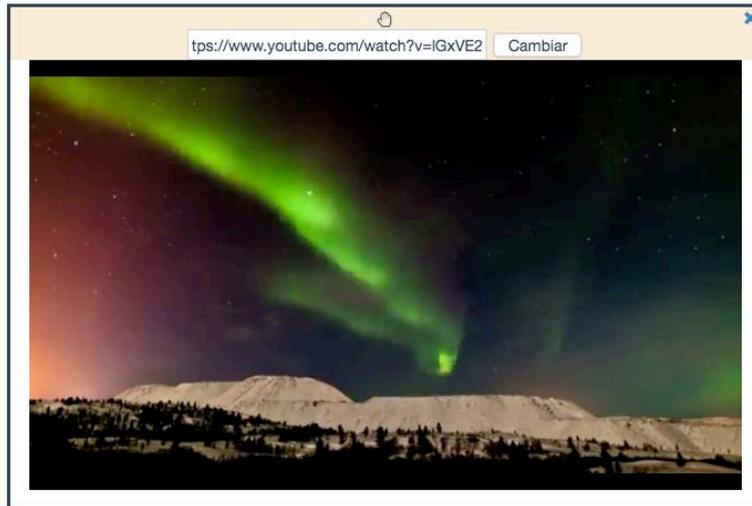


Figura 9.27. Manual de usuario - Video compartido

Sabremos cuando otro usuario cambia la URL a través de una notificación.

9.3.9.6 Dibujos

Utilizando este servicio podemos compartir dibujos con otros usuarios que también lo estén utilizando. Se pueden añadir cuadrados, círculos, triángulos y dibujos realizados con el puntero. También moverlos o borrarlo todo.

Estos cambios se ven en tiempo real en los dibujos de los otros usuarios.

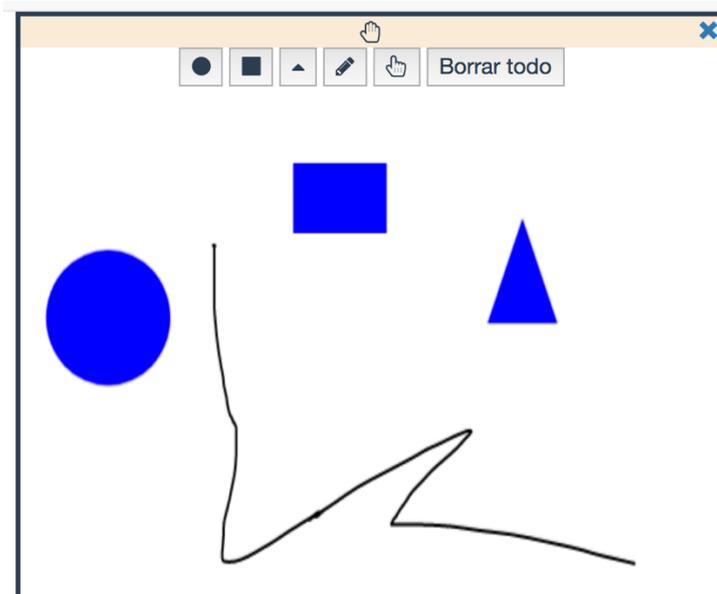


Figura 9.28. Manual de usuario - Dibujos

9.3.9.7 Radio

Con el servicio de radio podemos compartir emisoras y canciones con los usuarios de una sala.

Para ello introducimos una URL en el campo proporcionado y esta será automáticamente compartida con los demás usuarios. Podemos parar y reanudar su reproducción, o aumentar y disminuir el volumen.

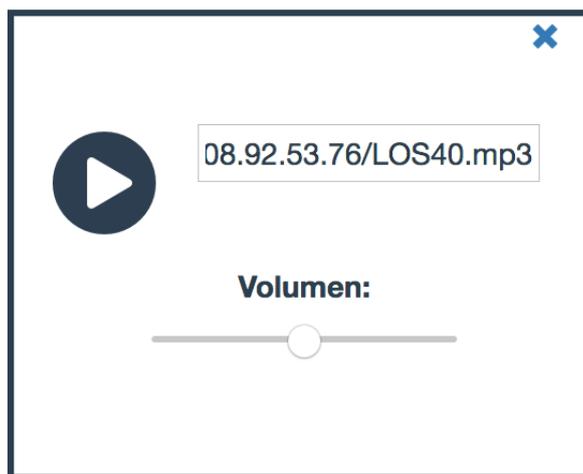


Figura 9.29. Manual de usuario – Radio

Sabremos cuando otro usuario cambia la URL a través de una notificación.

9.3.10 Ajustes de la cuenta

Utilizando el enlace “Ajustes de la cuenta” accedemos a una página en la que podemos modificar nuestros datos personales, la contraseña o eliminar nuestra cuenta.

COPERNICUS [Página Principal](#) [Sobre nosotros](#) [Aviso legal](#)   carlos 

Modificar datos de la cuenta **Cambiar foto de perfil**

Nombre:

Apellidos:

Correo electrónico:

Foto de perfil:

Cambiar contraseña

Contraseña actual:

Nueva Contraseña:

Repetir Contraseña:

Acciones peligrosas

Figura 9.30. Manual de usuario - Ajustes de la cuenta

Hay que tener en cuenta que los datos introducidos deben cumplir con unos requisitos:

- El nombre no puede ser menor que 3 caracteres ni mayor que 15.
- El apellido no puede ser mayor que 35 caracteres.
- Se comprobará que el email tenga un formato válido.
- La foto de perfil no puede ser mayor que 8MB.
- Se debe introducir la contraseña actual para poder cambiarla.
- La contraseña no puede ser menor que 8 caracteres.

9.3.11 Cerrar sesión

Una vez iniciamos sesión, podremos entrar en nuestra cuenta durante un día entero sin tener que volver a introducir las credenciales de nuevo. Si queremos acceder a otra cuenta sin tener que utilizar otro navegador, o simplemente no queremos que otra persona acceda a la nuestra, la solución es cerrar la sesión.

Existen dos formas de cerrar sesión, utilizando el menú lateral de la página principal o utilizando el menú desplegable que aparece al pulsar en nuestro nombre o foto de perfil en la barra de navegación.

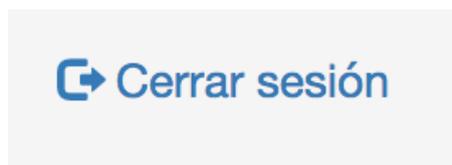


Figura 9.31. Manual de usuario - Cerrar sesión menú lateral

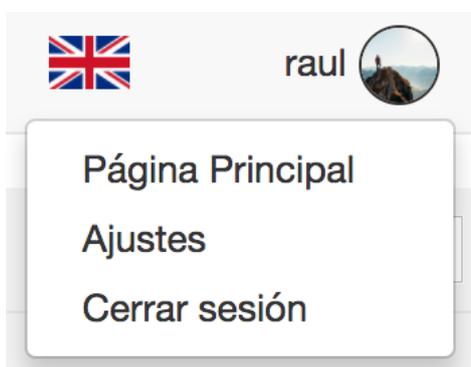


Figura 9.32. Manual de usuario - Cerrar sesión menú desplegable

9.3.12 Otros datos de interés

Si se quiere obtener información acerca del creador de la plataforma o información legal, se puede acceder a las vistas “Sobre nosotros” y “Aviso legal” a través de todas las barras de navegación (exceptuando la barra de navegación de las salas).

9.4 Manual del Programador

Con esta plataforma se busca que terceros programadores puedan incluir sus propios servicios, aumentando así las posibilidades que ofrece Copernicus. Conseguir algo así en una aplicación de estas características no es algo fácil, ya que hay que tener en cuenta que se debe añadir los servicios de manera dinámica en la interfaz. Sin embargo, se ha intentado permitir la inclusión de nuevos servicios de la forma más sencilla posible.

9.4.1 Primeros pasos para crear un servicio

Primero de todo, será necesario la creación de un nuevo fichero EJS que contendrá la interfaz del nuevo servicio. No existe ninguna restricción a la hora de diseñar la nueva vista, exceptuando los siguientes puntos:

- El EJS estará formado por un div con un id que contendrá todas las etiquetas.
- Dentro del div principal existirá un div con la clase “grid-stack-item-content”. Este div representará la zona del servicio que permitirá a los usuarios moverlo.
- Se incluirá en la ruta “/views/servicios”.

Una vez creada la vista, esta deberá ser incluida en una variable de *serviciosController* utilizando jQuery. Como ya se explicó con anterioridad, fue necesario incluir el código de las vistas de los servicios en este controlador por el problema derivado de añadir las vistas de forma dinámica a la interfaz.

Posteriormente, es necesario modificar el método *addServicio* de *serviciosController*. Al invocar este método pasándole el nombre de nuestro servicio como parámetro, este debe mostrarse correctamente en la interfaz. Este paso es muy sencillo, y se podría copiar de otros servicios ya creados.

Una vez modificado el método *addServicio*, este se deberá invocar desde un nuevo enlace incluido en el listado de servicios de la vista *chatroom.ejs*.

Finalmente, sería necesario incluir en la vista del servicio un botón o enlace que invoque al método *eliminarServicio* pasándole el nombre que hemos. De esta forma se podrá eliminar el servicio de la interfaz.

Una vez realizados estos pasos, ya podremos abrir, mover, redimensionar y cerrar nuestro servicio.

9.4.2 Añadir funcionalidad y comunicación en tiempo real a nuestro servicio

Realizando los anteriores pasos, ya dispondremos de un servicio. Sin embargo, si deseamos añadir nueva funcionalidad y permitir las comunicaciones en tiempo real, tendremos que realizar algún cambio más.

Se deberá crear un manager en la ruta “/public/javascript/managers”. Este manager estará encargado de gestionar el servicio, y será instanciado en el servicio *webSocketService*. El objetivo de incluirlo a este último servicio es que cuando este reciba un mensaje del servidor de WebSockets, nosotros podremos analizarlo y ejecutar los métodos que estimemos oportunos de nuestro manager. Al instanciar nuestro manager, será necesario pasarle como parámetro el servidor de WebSockets *wss* localizado en *webSocketService*. Con este parámetro, el manager podrá enviar mensajes al servidor de WebSockets.

Como le enviaremos mensajes al servidor de WebSockets, será necesario modificar el fichero *websockets*, de modo que pueda gestionar los mensajes que reciba y redirigirlos al *webSocketService* de cada usuario.

Finalmente, será necesario definir un controlador en la ruta “/public/javascript/controllers”, que se enlace con la vista del servicio. A este controlador se le inyectará *webSocketService*, de modo que tendremos acceso a la instancia de nuestro manager.

Capítulo 10. Conclusiones y Ampliaciones

10.1 Conclusiones

Con este proyecto se ha creado la base de lo que puede llegar ser una alternativa real para nuestras comunicaciones del día a día. Se han completado los objetivos planteados ofreciendo una plataforma en la que podemos realizar comunicaciones en tiempo real con nuestros contactos a través de salas privadas.

Se ha conseguido satisfactoriamente realizar una plataforma de comunicaciones extensible, pero que, a su vez, permite mantener una interfaz simple e intuitiva que pueda ser personalizada por sus usuarios.

Copernicus ha sido una gran oportunidad para descubrir tecnologías que hasta entonces eran prácticamente desconocidas para mí. Entre ellas cabe destacar las tecnologías que hacen posible las comunicaciones en tiempo real, WebRTC y el servidor de WebSockets. Estas tecnologías abren un mundo de posibilidades gracias a las comunicaciones punto a punto de WebRTC y a las interacciones bidireccionales entre clientes y servidores de WebSockets. Además, parece que su influencia acabará aumentando con el paso de los años.

No ha sido fácil el desarrollo de este proyecto y tanto desde su concepción y diseño hasta su implementación y pruebas se han enfrentado diversos obstáculos y decisiones. Sin embargo, siempre se han mantenido en mente los objetivos inicialmente definidos para esta aplicación.

Copernicus está pensado para seguir en evolución y mejorar gracias a las nuevas ideas de terceras personas. Esta es la única forma para que algún pueda llegar a ser, de verdad, el centro de todas nuestras comunicaciones.

10.2 Ampliaciones

En esta sección se documentarán todas aquellas ampliaciones que, por motivos de tiempo, han sido desestimadas y se les ha dado una menor prioridad respecto a otras.

10.2.1 Enviar mensajes sin utilizar salas

Durante la realización de las pruebas de usabilidad, un usuario preguntó si era posible enviar mensajes a sus contactos sin tener que crear una sala. Como ya se dijo anteriormente, esta funcionalidad no era un requisito de este proyecto ya que solo se planteó proporcionar comunicaciones a través de salas. Por ello, se dejó esta funcionalidad como una ampliación, ya que el proyecto se encontraba en un estado muy avanzado y no era posible llevar a cabo una ampliación de tal magnitud con el tiempo disponible.

10.2.2 Aceptar solicitudes de salas automáticamente

La idea principal de las solicitudes de unión a salas es permitir a los usuarios decidir en qué salas quieren o no quieren participar, de modo que no se saturen sus listados de salas con aquellas indeseadas. Tras una conversación con un usuario, este planteó la idea de dar la posibilidad a los usuarios de decidir si quieren o no aceptar estas solicitudes de manera automática. De este modo, se añadiría una opción en los ajustes de la cuenta para activar o desactivar esta función.

10.2.3 Aplicación para móviles

Al comienzo de este proyecto se evaluó la posibilidad de que la aplicación pudiese ser utilizada perfectamente en los navegadores de los teléfonos móviles o tabletas. Pero, tras una evaluación de las tecnologías a utilizar, quedó claro que WebRTC no podría ser utilizado en algunos de los dispositivos (como es el caso de iOS). Además, resulta más razonable el desarrollo de aplicaciones móviles para estas plataformas, de modo que se le ofrece al usuario la mejor experiencia posible.

El problema es que desarrollar una aplicación móvil que siguiera los objetivos de este proyecto requeriría, a su vez, replantearlo para poder adaptarlo a las características de un dispositivo móvil.

Sin embargo, no se cierra la posibilidad de desarrollar aplicaciones móviles en un futuro que hicieran uso de la API REST construida en este proyecto.

10.2.4 Ampliación del número de servicios

Una de las características de esta plataforma es su interfaz “modular” que le permite añadir y eliminar servicios de una forma sencilla y sin que se complique la estructura de sus interfaces. De esta forma, la plataforma podría seguir evolucionando e incluir nuevas formas de comunicación o incluso hasta videojuegos.

10.2.5 Recuperar contraseña por email

Uno de los motivos por los que se solicita un email al crear una cuenta es porque se planteó la posibilidad de recuperar la contraseña por email. Finalmente, no se dispuso del tiempo suficiente para realizarlo. Por lo que se deja como una ampliación para el futuro.

Capítulo 11. Planificación del Proyecto y Presupuesto finales

A lo largo de la realización de este proyecto, se han encontrado diferentes obstáculos que han derivado en una reorganización de parte de la planificación y, por tanto, del presupuesto.

Las primeras semanas de trabajo se pudo seguir la planificación sin ningún problema, ya que se había diseñado un calendario que fuera compatible con las horas en el trabajo y en la universidad. Sin embargo, la aparición de nuevos proyectos, exámenes y otras cuestiones personales imposibilitaron en algunas ocasiones continuar con lo establecido. También hay que tener en cuenta algún error en la estimación de ciertas actividades.

11.1 Planificación Final

En este apartado se explicará la planificación resultante una vez terminada esta documentación.

Debido a que tuve que compaginar este proyecto con mis clases en la universidad y un trabajo, la planificación final y el calendario seguido difieren respecto a los planificados inicialmente.

Primero cabe destacar que se pudo seguir sin ningún problema el calendario definido las primeras semanas. Sin embargo, debido a los motivos ya explicados, este calendario se tuvo que adaptar para cumplir con las horas definidas para cada semana. Ciertas semanas no se podía trabajar en las horas y días definidos, por lo que el calendario resultante fue el siguiente:

- Lunes a viernes: 4 horas
- Sábado y domingo: 11 horas

También hubo alguna semana que se tuvo que dedicar menos tiempo al proyecto, pero que fue compensada dedicando más horas en otras.

A continuación, se muestra la planificación final y se explican los cambios ocurridos.

ID	Nombre	Inicio	Fin
1	Documentación inicial	30/1/17	12/2/17
2	Reunión con director del proyecto	30/1/17	30/1/17
3	Motivación y objetivos	31/1/17	2/2/17
4	Planificación	3/2/17	5/2/17
5	Presupuesto	3/2/17	5/2/17
6	Estudio de la situación actual	6/2/17	7/2/17
7	Busqueda y estudio de las tecnologías	8/2/17	12/2/17
8	Análisis	13/2/17	5/3/17
9	Requisitos del sistema y casos de usos	13/2/17	19/2/17
10	Definición de las clases	20/2/17	24/2/17
11	Interfaces de usuario	25/2/17	1/3/17
12	Plan de pruebas	2/3/17	5/3/17
13	Diseño	6/3/17	19/3/17
14	Arquitectura del sistema y diagramas	6/3/17	12/3/17
15	Diseño de la base de datos	13/3/17	13/3/17
16	Plan de pruebas	14/3/17	19/3/17
17	Implementación	20/3/17	14/6/17
18	Estructura de la aplicación web	20/3/17	24/3/17
19	Inicio de sesión y registro	25/3/17	31/3/17
20	Interfaz modular de las salas	1/4/17	3/4/17
21	Gestión de contactos	4/4/17	18/4/17
22	Gestión de salas	19/4/17	3/5/17
23	Navegación entre las vistas de la aplicación	4/5/17	7/5/17
24	Servicio de videollamadas	8/5/17	21/5/17
25	Servicio de chat de texto	22/5/17	28/5/17
26	Servicio de dibujo	29/5/17	4/6/17
27	Servicio de presentaciones	5/6/17	9/6/17
28	Servicio de audio y video compartido	10/6/17	14/6/17
29	Pruebas	25/3/17	21/6/17
30	Reuniones de seguimiento	21/4/17	28/6/17
31	Muestra de prototipo inicial	21/4/17	21/4/17
32	Reunión plataforma terminada	14/6/17	14/6/17
33	Reunión dudas finales documentación	28/6/17	28/6/17
34	Manuales del sistema	27/6/17	28/6/17
35	Finalización de la documentación	22/6/17	30/6/17
36	Revisión de la documentación	1/7/17	2/7/17
37	Entrega de la documentación	3/7/17	7/7/17
38	Preparación de la defensa	8/7/17	16/7/17
39	Defensa	17/7/17	21/7/17
40	Fin del proyecto	21/7/17	21/7/17

Figura 11.1. Tareas planificación final

El primer problema encontrado ocurrió durante la búsqueda y estudio de tecnologías. Llevó más horas de las planeadas por lo que se tuvo que extender su periodo de realización. Durante la realización inicial ya se tenía una idea previa de las tecnologías que se iban a utilizar, por eso se estimó un tiempo menor. Sin embargo, la búsqueda de tecnologías como Gridstack.js o Cloudinary y aprender a utilizarlas fue lo que aumentó el tiempo necesario.

Se necesitaron unas horas más para poder realizar un análisis y diseño lo suficientemente detallados. Fue necesario un estudio adicional de algunas tecnologías en estas fases para poder realizar los diagramas más realistas posibles. Aunque tareas como el diseño de la base de datos pudieron realizarse en menos tiempo.

Durante la implementación hubo tareas que no llevaron tanto tiempo como el planeado gracias al análisis y diseño. Sin embargo, si hubo contratiempos no esperados que

aumentaron más tiempo de trabajo. Además, a la hora de realizar las pruebas me di cuenta de que no eran necesarias 5 horas semanales y se pudieron reducir a 2. Esas 3 horas no utilizadas para las pruebas se dedicaron a la implementación, unas horas que fueron muy útiles para la solución de algunos contratiempos encontrados. Uno de los mayores contratiempos apareció durante la realización del primer servicio (el servicio de videollamadas). Como ya se explicó en el apartado [7.4.1 Problemas Encontrados](#) de este documento, Gridstack.js no está pensado para facilitar la adición de vistas que tienen controladores enlazados, por lo que se tuvo que buscar una solución a este problema. También se encontraron ciertos problemas con otros servicios descritos en el mismo apartado.

Las fechas de las reuniones de seguimiento se modificaron de modo que se realizaron cuando era necesario y cuando los directores de este proyecto podían. Se realizó una reunión de más para comentar unas dudas finales acerca de la documentación.

Los manuales del sistema fueron realizados al terminar la implementación y por ello, se realizaron a la vez que otros apartados del documento que no se pudieron realizar hasta ese momento.

En el momento en que se escribe esta sección, se está realizando la finalización de la documentación. Por lo tanto, las siguientes tareas todavía están por realizar.

Actualmente se han utilizado 325 horas para la realización de este proyecto, junto a las tareas que se realizarán hasta concluir el proyecto el día 21 de Julio se espera haber realizado 370 horas. Solo 6 horas más de las planificadas inicialmente.

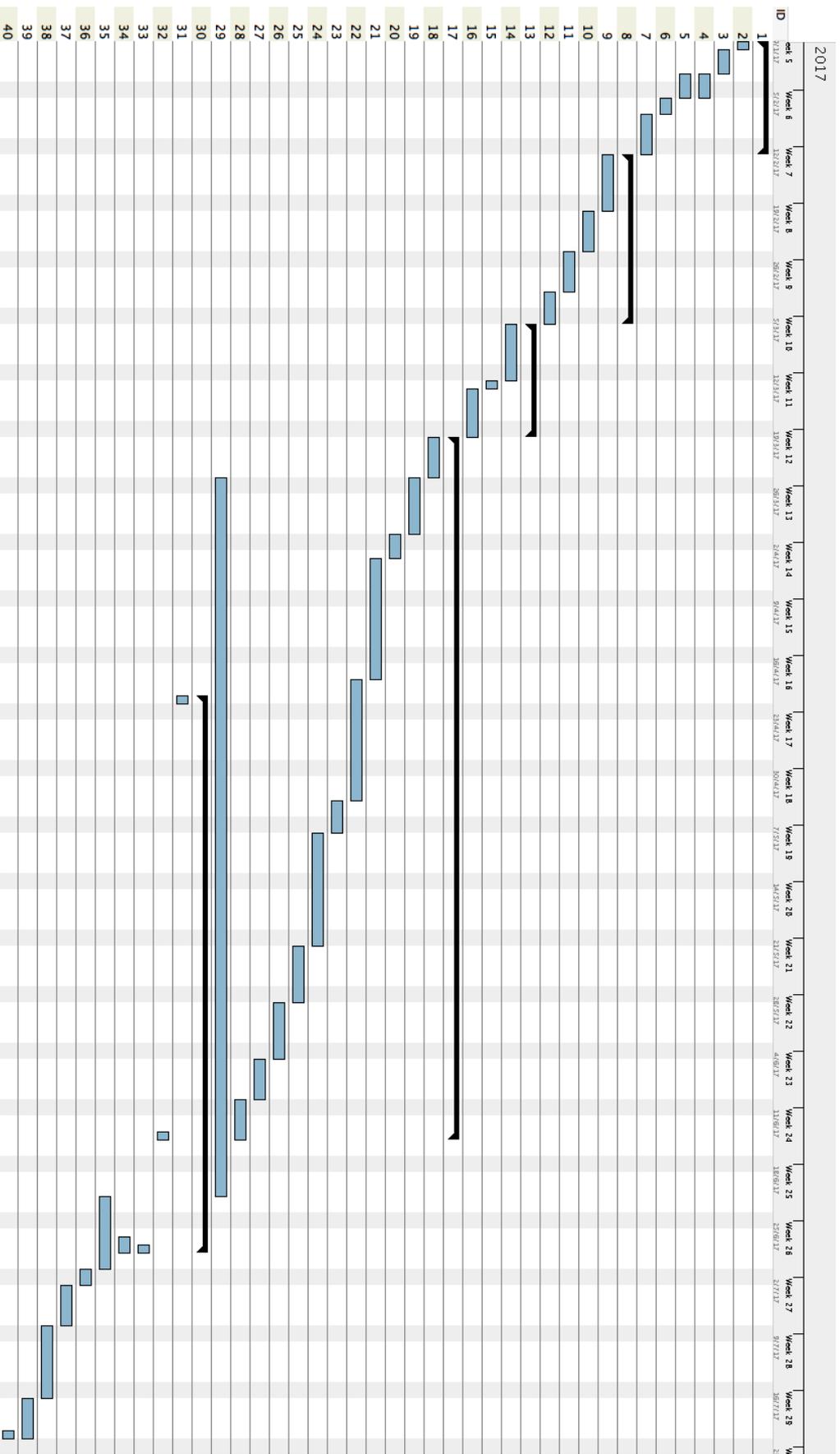


Figura 11.2. Diagrama de Gantt planificación final

11.2 Presupuesto Final

En esta sección se mostrará el presupuesto interno y el presupuesto para el cliente finales del proyecto.

11.2.1 Presupuesto Interno

Para la realización del presupuesto interno final se actualizarán los recursos humanos y hardware, las licencias y los gastos indirectos del presupuesto interno inicial. Los cálculos para obtener los costes de cada concepto serán los mismos que los utilizados para el presupuesto interno inicial, pero utilizando las 370 horas finales.

Item	Concepto	Cantidad	Amortización	Precio unitario (€)	Total (€)
1	Recursos humanos				
1.1	Documentación inicial	30	100%	25,00 €	750,00 €
1.2	Análisis del sistema	45	100%	40,00 €	1.800,00 €
1.3	Diseño del sistema	30	100%	50,00 €	1.500,00 €
1.4	Implementación del sistema	156	100%	30,00 €	4.680,00 €
1.5	Pruebas	39	100%	35,00 €	1.365,00 €
1.6	Finalización del proyecto	70	100%	25,00 €	1.750,00 €
2	Recursos Hardware				
2.1	Portátil	1	1,33%	1.570,00 €	20,88 €
2.2	Ordenador de sobremesa	1	1,33%	800,00 €	10,64 €
2.3	Servidor Heroku	1	100,00%	0,00 €	0,00 €
3	Licencias				
3.1	Windows 10	1	1,33%	279,00 €	3,71 €
3.2	macOS Sierra	1	1,33%	0,00 €	0,00 €
3.3	Microsoft Office 365	6	100,00%	4,20 €	25,20 €
3.4	Neo4j Community Edition	1	100%	0,00 €	0,00 €
3.5	AngularJS 1.6	1	100%	0,00 €	0,00 €
3.6	Node.js v7.5.0	1	100%	0,00 €	0,00 €
3.7	Express 4	1	100%	0,00 €	0,00 €
3.8	Bootstrap v3.3.7	1	100%	0,00 €	0,00 €
3.9	GanttProject v2.8.1	1	100%	0,00 €	0,00 €
3.10	IntelliJ IDEA Ultimate Edition	6	100%	49,90 €	299,40 €
3.11	Otro software gratuito	-	100%	0,00 €	0,00 €
4	Gastos indirectos				
4.1	Internet	6	8,33%	50,00 €	24,99 €
4.2	Electricidad	370	100,00%	0,12 €	44,40 €
4.3	Pizarra blanca	1	2,00%	60,00 €	1,20 €
4.4	Rotuladores Pizarra	4	50,00%	2,00 €	4,00 €
4.5	Papel	100	50,00%	0,07 €	3,25 €
4.6	Lápices	2	50,00%	0,60 €	0,60 €
4.7	Impresora	1	2,00%	80,00 €	1,60 €
4.8	Tóner	1	50,00%	37,00 €	18,50 €
<i>Subtotal</i>					12.303,37 €
<i>Beneficio (10%)</i>					1.230,34 €
Total					13.533,71 €

Tabla 11.1. Presupuesto interno final

A pesar de que el número de horas para realizar este proyecto se asemeja al número de horas planificadas, estas horas se tuvieron que repartir en distintas fases. Por lo que existe un sobrecoste de 253,79€.

11.2.2 Presupuesto para el cliente

Al igual que para el presupuesto interno final, el presupuesto para el cliente final se realizará siguiendo los pasos descritos durante la realización del presupuesto para el cliente inicial. A continuación, se muestra este presupuesto.

Concepto		Coste Total Concepto
<i>Analisis y diseño del sistema</i>		3.722,18 €
<i>Implementación</i>		5.102,18 €
<i>Pruebas</i>		1.787,18 €
<i>Documentación y presentación del proyecto</i>		2.922,18 €
	<i>Subtotal</i>	13.533,71 €
	<i>IVA (21%)</i>	2.842,08 €
	TOTAL	16.375,79 €

Tabla 11.2. Presupuesto cliente final

Con esta información se puede concluir que existe un sobrecoste de 307,09€. Esto ha sido debido a que no se planificó de forma exacta el número de horas necesarias para cada tarea. Sin embargo, esta información será muy útil para saber definir con mayor exactitud los recursos necesarios en proyectos futuros.

Capítulo 12. Referencias Bibliográficas

12.1 Libros y Artículos

[Cambridge04] Cambridge University Press. “The Object Primer 3rd Edition. Agile Model Driven Development with UML 2”. ISBN#: 0-521-54018-6

12.2 Referencias en Internet

[42ID14] 42ID. “ANNE Stack – AngularJS, Node, Neo4j and Express”. <http://www.42id.com/articles/anne-stack-angular-js-node-neo4j-and-express/>. [Accedido por última vez el 11/02/17]

[Álvarez15] Álvarez, Cecilio. “Angular.js módulos y arquitectura”. <https://www.genbetadev.com/desarrollo-web/angular-js-modulos-y-arquitectura>. [Accedido por última vez el 11/02/17]

[Damphousee15] Damphousse, Robert. “Token Based Authentication for Single Page Apps”. <https://stormpath.com/blog/token-auth-spa>. [Accedido por última vez el 26/03/17]

[Dutton13] Dutton, Sam. “WebRTC in the real world: STUN, TURN and signaling”. <https://www.html5rocks.com/en/tutorials/webrtc/infrastructure/>. [Accedido por última vez el 18/05/17]

[Express] Express. “Utilización del middleware”. <http://expressjs.com/es/guide/using-middleware.html>. [Accedido por última vez el 26/03/17]

[Google] Google. <https://material.angularjs.org/latest/demo/dialog>. [Accedido por última vez el 08/04/17]

[Guildernstern15] Guildernstern. “Web-RTC: Usando PeerConnection en modo local”. <https://ferweblab.wordpress.com/2015/04/26/web-rtc-usando-peerconnection-en-modo-local/>. [Accedido por última vez el 11/05/17]

[Hassan08] Hassan Montero, Y. “Guía de Evaluación Heurística de Sitios Web”. <http://www.nosolousabilidad.com/articulos/heuristica.htm>. [Accedido por última vez el 18/06/17]

[Herman15] Herman, Michael. “Testing Node.js with Mocha and Chai”. <http://mherman.org/blog/2015/09/10/testing-node-js-with-mocha-and-chai/#.WVfohtPyi1t>. [Accedido por última vez el 02/04/17]

[IsWebRTCReadyYet] “Is WebRTC ready yet?”. <http://iswebrtcreadyyet.com/>. [Accedido por última vez el 24/06/17]

[Jaketrent13] JakeTrent. “HTTPS Redirect for Node on Heroku”. <https://jaketrent.com/post/https-redirect-node-heroku/>. [Accedido por última vez el 01/04/17]

[Lewis15] Lewis, Ryan. “Node Web frameworks comparison: Express, hapi and Sails”.
<https://www.pluralsight.com/blog/software-development/node-web-frameworks>.

[Accedido por última vez el 12/02/17]

[MDN] Mozilla Developer Network. <https://developer.mozilla.org>. 2017.

[Millner08] Millner, Rachel. “Four regular expressions to check email addresses”.
https://www.wired.com/2008/08/four_regular_expressions_to_check_email_addresses/.

[Accedido por última vez el 27/03/17]

[Pixabay] Fotografías con licencia Creative Commons sin derechos de propiedad intelectual. <https://pixabay.com>. 2017

[Pukey22] Pukey22. “Login and Register tabbed form”.
<https://bootsnipp.com/snippets/featured/login-and-register-tabbed-form>. [Accedido por

última vez el 25/03/17]

[Sevilleja15] Sevilleja, Chris. “Authenticate a Node.js API with JSON Web Tokens”.
<https://scotch.io/tutorials/authenticate-a-node-js-api-with-json-web-tokens>. [Accedido

por última vez el 27/03/17]

[StackOverflow] Stack Overflow. <https://stackoverflow.com/>. 2017.

[Uechi11] Uechi, Fábio. “JMeter – POSTing / Sending JSON data”.
<http://www.fabiouechi.tk/2011/03/jmeter-posting-json-data-to-web-service.html>.

[Accedido por última vez el 18/06/17]

[W3Schools] W3Schools. <http://www.w3schools.com/>. 2017.

[Wikipedia] Wikipedia. <https://www.wikipedia.org/>. 2017

Capítulo 13. Apéndices

13.1 Glosario y Diccionario de Datos

Descripción breve de algunos términos importantes utilizados en esta memoria.

- **ANNE:** Son las siglas de AngularJS, Node.js, Neo4j y Express, y se refiere a aquellas aplicaciones desarrolladas utilizando esas tecnologías.
- **Servicio:** Aquella herramienta proporcionada en las salas que puede ser abierta, desplazada, redimensionada o cerrada. Permite realizar algún tipo de comunicación entre los usuarios de una sala como, por ejemplo, videollamadas o dibujos colaborativos.
- **Servidor de WebSockets:** Estos servidores permiten el envío bidireccional de datos entre el cliente y el servidor sobre un único socket TCP.
- **WebRTC:** colección de protocolos de comunicaciones y APIs dirigidos a simplificar el desarrollo de aplicaciones web basadas en comunicaciones en tiempo real.

13.2 Contenido Entregado en el Archivo adjunto

En este apartado se explicará el contenido de los directorios que podemos encontrar en el archivo adjunto proporcionado.

13.2.1 Contenidos

Directorio	Contenido
<i>./ Directorio raíz del Archivo adjunto</i>	Contiene un fichero leeme.txt explicando toda la organización en ficheros y carpetas. También contiene una imagen con el logo de la aplicación.
<i>./aplicacion</i>	Contiene toda la estructura de directorios de la aplicación desarrollada.
<i>./documentacion</i>	Contiene toda la documentación asociada al proyecto.
<i>./documentacion/diagramas</i>	Imágenes con los diagramas desarrollados.

13.2.1.1 Estructura del directorio “aplicacion”

En esta sección se explicación los directorios contenidos en la carpeta “aplicacion” en la cual se almacenan todos aquellos ficheros y directorios que conforman la aplicación desarrollada.

Directorio	Contenido
<i>./ Directorio raíz de “aplicacion”</i>	Contiene los ficheros bower.json, package.json y package-lock.json que son necesarios para la instalación de las dependencias de la aplicación. Además, encontramos el fichero app.js el cual prepara la aplicación y ejecuta los servidores, y también un fichero oculto .env, el cual contiene las variables de entorno necesarias para la correcta ejecución de la aplicación.
<i>./api</i>	Contiene todos aquellos ficheros que permiten el correcto funcionamiento de la API REST.

<code>./public</code>	Contiene todo el código fuente que forma la parte del cliente. También almacena fotografías y hojas de estilos.
<code>./server</code>	Contiene el código fuente para la configuración de los servidores y para gestionar las peticiones de vistas al servidor HTTP.
<code>./test</code>	Contiene un fichero con las pruebas unitarias realizadas.
<code>./views</code>	Contiene los ficheros EJS que constituyen las interfaces de la aplicación.

13.2.2 Código Ejecutable e Instalación

Se explicará una breve descripción para la instalación y ejecución de la aplicación. Para más detalles dirigirse a los apartados [9.1 Manual de Instalación](#) y [9.2 Manual de Ejecución](#).

13.2.2.1 Instalación

1. Descargar Node.js (se utilizó la versión 6.11.0 LTS): <https://nodejs.org/es/>
2. Descargar Git (se utilizó la versión 2.13.2): <https://git-scm.com/downloads>
3. Descargar Neo4j (se utilizó la versión Community Edition 3.2.1): <https://neo4j.com/download/community-edition/>
4. Ejecutar la base de datos Neo4j y establecer la contraseña “thesis” sin comillas.
5. Actualizar npm con comando “npm install npm -g”.
6. Abrir terminal/símbolo de sistema en carpeta “Copernicus/aplicacion” y ejecutar el comando “npm install”.

13.2.2.2 Ejecución

1. Arrancar la base de datos Neo4j (Si no se está ejecutando tras los pasos anteriores).
2. Abrir terminal/símbolo de sistema en carpeta “Copernicus/aplicacion” y ejecutar el comando “npm start”. La aplicación estará desplegada cuando aparezca el mensaje “Servidor de WebSockets escuchando en el puerto: 8080”.
3. Dirigirse con un navegador con soporte para WebRTC, como Firefox o Google Chrome, a la dirección localhost:8080.

Para detener la ejecución del sistema, únicamente deberemos de parar el proceso que se está ejecutando en la terminal o símbolo del sistema.

Es importante tener en cuenta que, aunque la aplicación se ejecuta en local, es necesario disponer de conexión a internet. Esto es debido a que se utilizó un servidor remoto para almacenar y recuperar fotografías.

También es posible probar la aplicación dirigiéndose a la dirección <https://copernicus-platform.herokuapp.com/>. Cabe destacar que se utiliza un servidor gratuito de Heroku por lo que el rendimiento y respuesta de la aplicación puede no ser óptimo.

13.2.3 Ficheros de Configuración

Para el correcto funcionamiento de la aplicación es necesario disponer de un fichero oculto llamado `.env` en el directorio “Copernicus/aplicacion”. Este fichero contiene variables de entorno necesarias para la aplicación. A continuación, se muestra la información que debe contener este fichero:

```
JWT_SECRET=  
LOCAL_DB_USER=  
LOCAL_DB_PASS=  
PASS_TURN=  
API_SECRET_IMAGE_HOSTING=  
CLOUD_NAME=  
CLOUD_API_KEY=  
CLOUD_API_SECRET=
```

13.3 Índice Alfabético

A

administrador · 67, 69, 73, 74, 80, 83, 84, 85, 95, 97, 98, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 121, 122, 123, 124, 141, 143, 169, 172, 191, 195, 197, 198, 199, 200, 202, 204, 211, 227, 229, 230, 231, 232, 236, 238, 245, 249, 254, 286, 288

Administrador · 67, 74, 96, 97, 200, 232

AngularJS · 34, 39, 40, 52, 54, 66, 87, 149, 152, 153, 154, 157, 160, 218, 219, 221, 311, 313

ANNE · 34, 220, 311, 313

Ch

Chat de texto · 4, 27, 292, 293

C

Comunicaciones · 5, 67

contactos · 27, 28, 32, 35, 60, 66, 68, 69, 70, 77, 78, 79, 89, 90, 95, 98, 105, 108, 109, 121, 129, 130, 141, 142, 143, 157, 168, 171, 172, 176, 179, 181, 184, 203, 204, 210, 211, 212, 215, 236, 237, 238, 244, 245, 249, 250, 253, 254, 258, 270, 280, 283, 284, 285, 286, 287, 301, 302

D

Dibujos · 4, 27, 33, 212, 213, 246, 250, 251, 255, 295

E

Express · 34, 36, 37, 40, 88, 152, 154, 167, 220, 311, 312, 313

F

framework · 34, 36, 39, 40, 48, 49, 139, 149, 152, 154, 157, 194, 219, 220, 223, 234

G

grafos · 36, 41, 60, 172, 220

Gridstack.js · 49, 221, 224, 305, 306

I

interfaces · 4, 6, 26, 27, 45, 49, 52, 55, 60, 119, 120, 121, 127, 175, 182, 184, 188, 191, 265, 266, 303, 315

J

JavaScript · 34, 35, 38, 39, 44, 47, 54, 86, 149, 157, 216, 217, 218, 219

jQuery · 49, 54, 175, 224, 299

M

manager · 154, 158, 170, 221, 300

miembro · 30, 72, 73, 74, 80, 82, 83, 84, 85, 97, 110, 111, 112, 113, 114, 115, 116, 117, 118, 121, 124, 143, 144, 173, 197, 198, 199, 200, 205, 206, 211, 229, 230, 231, 232, 233, 239, 240, 245, 249, 254

moderador · 72, 74, 80, 82, 83, 84, 85, 97, 98, 110, 111, 112, 113, 114, 115, 116, 117, 118, 121, 123, 124, 143, 144, 173, 191, 197, 198, 199, 200, 205, 211, 229, 230, 231, 232, 239, 245, 249, 254

modular · 4, 6, 27, 119, 124, 303

módulos · 50, 86, 139, 153, 224, 269, 311

N

Node.js · 5, 7, 33, 34, 36, 37, 38, 39, 40, 50, 54, 139, 154, 194, 218, 219, 220, 221, 225, 276, 277, 312, 313, 315

P

Planificación · 26, 58, 304

plataforma · 4, 25, 27, 30, 33, 38, 39, 42, 43, 50, 68, 150, 156, 221, 294, 298, 299, 301, 303

Presentaciones · 4, 28, 212, 246, 250, 255, 293
presupuesto · 26, 59, 63, 64, 65, 304, 308, 309

R

Radio · 4, 28, 212, 213, 246, 247, 250, 251, 255, 296
REST · 5, 7, 46, 60, 66, 86, 91, 139, 153, 154, 155, 156,
163, 169, 194, 201, 272, 302, 314

S

salas · 27, 29, 33, 35, 50, 52, 60, 66, 67, 69, 70, 72, 73,
78, 83, 84, 88, 89, 95, 96, 97, 100, 101, 104, 107,
108, 109, 110, 115, 116, 117, 118, 119, 121, 122,
123, 124, 125, 126, 131, 132, 142, 143, 144, 149,
150, 153, 154, 155, 156, 157, 159, 171, 172,
176, 179, 180, 181, 182, 184, 189, 196, 197, 203,
204, 205, 211, 212, 213, 215, 220, 222, 224, 225,
228, 229, 230, 234, 237, 238, 240, 245, 246, 247,
249, 250, 251, 252, 254, 256, 258, 266, 270, 271,
278, 285, 286, 287, 288, 289, 298, 301, 302, 313
servicios · 4, 25, 27, 28, 29, 30, 32, 33, 37, 47, 50, 54,
60, 66, 67, 71, 74, 80, 81, 86, 88, 91, 110, 111, 112,
113, 124, 125, 126, 133, 139, 149, 150, 152, 153,
154, 157, 159, 170, 175, 182, 183, 184, 186, 188,
189, 194, 210, 212, 213, 214, 219, 221, 222, 246,

247, 250, 251, 255, 256, 259, 260, 289, 290, 299,
303, 306

solicitud de contacto · 68, 69, 77, 90, 105, 106, 141,
142, 172, 177, 195, 203, 210, 226, 227, 236, 237,
280, 281

solicitud de unión a sala · 78, 84, 106, 107, 115, 116,
142, 144, 196, 197, 199, 203, 205, 228, 229, 232,
237, 239, 240

V

videollamadas · 5, 25, 27, 28, 43, 67, 71, 94, 184, 212,
222, 224, 246, 250, 255, 291, 306, 313

Videos · 4, 27

W

web · 4, 6, 25, 27, 28, 29, 31, 32, 33, 34, 36, 39, 40, 43,
47, 48, 49, 66, 73, 86, 87, 88, 155, 209, 210, 219,
220, 244, 248, 253, 259, 260, 261, 263, 264, 311,
312, 313

WebRTC · 5, 7, 31, 33, 43, 44, 73, 221, 222, 265, 277,
301, 302, 311, 313, 315

WebSockets · 5, 7, 33, 36, 42, 56, 66, 86, 87, 101, 152,
153, 154, 155, 158, 159, 166, 167, 170, 221, 222,
277, 300, 301, 313, 315

13.4 Código Fuente

El código fuente puede ser encontrado en el directorio “Copernicus/aplicacion”. Para una explicación del contenido de este directorio puede leer el apartado [13.2.1.1 Estructura del directorio “aplicacion”](#).