



Universidad de Valladolid

ESCUELA DE INGENIERÍA INFORMÁTICA (SG)

Grado en Ingeniería Informática de Servicios y
Aplicaciones

Plataforma de Gestión de Itinerarios GPSSM

Alumno: David Burgos Perela

Tutores: Aníbal Bregón Bregón

Miguel Á. Martínez Prieto

“He fallado una y otra vez en mi vida. Esa es la razón principal de mi éxito”.

- Michael Jordan

RESUMEN

La popularización del turismo ha permitido que todo tipo de personas realicen viajes que desean recordar y compartir con sus amigos y seres queridos. A su vez, la revolución de los dispositivos móviles e Internet ha permitido la creación de contenidos en cualquier momento y lugar. El presente proyecto quiere aprovechar la necesidad de dar un formato a toda la información sobre los viajes capturados en nuestros teléfonos móviles, y gestionarla en una plataforma que nos permita clasificar los distintos viajes realizados, así como como consultarlos y compartirlos, para que otras personas puedan seguirlos o reinventarlos.

La plataforma empresarial desarrollada implementa múltiples servicios REST que permiten las operaciones de creación, consulta, modificación y eliminación de información sobre usuarios, relaciones de amistad, itinerarios turísticos y su contenido multimedia asociado, y puntos de interés turístico. Estos servicios permiten la comunicación con otras plataformas como dispositivos Android para la gestión del contenido. Además, la plataforma proporciona una interfaz web donde los usuarios pueden consultar y gestionar sus itinerarios creados desde un dispositivo móvil.

Índice general

CAPÍTULO 1	1
1.1 MOTIVACIÓN	2
1.2 OBJETIVOS	5
1.3 DESCRIPCIÓN DEL PROYECTO	5
1.4 ESTRUCTURA DEL DOCUMENTO	8
CAPÍTULO 2	9
2.1 ANÁLISIS DE COMPETENCIA	9
2.1.1 APLICACIONES DEL GRUPO I	10
2.1.2 APLICACIONES DEL GRUPO II	18
2.1.3 APLICACIONES DEL GRUPO III	29
2.2 CONCLUSIONES	33
CAPÍTULO 3	35
3.1 METODOLOGÍA DE DESARROLLO	35
3.2 DETALLE DE PLANIFICACIÓN	37
3.3 ESTIMACIONES TEMPORALES	39
3.3.1 ESTIMACIÓN POR PUNTOS DE FUNCIÓN	39
3.3.2 ESTIMACIÓN DE ESFUERZO POR COCOMO	42
3.4 PRESUPUESTO	44
3.4.1 PRESUPUESTO HARDWARE	45
3.4.2 PRESUPUESTO SOFTWARE	45
3.4.3 PRESUPUESTO DE RECURSOS HUMANOS	46
3.4.4 PRESUPUESTO TOTAL DEL PROYECTO	46
3.5 ESFUERZO REAL DEL PROYECTO	46
CAPÍTULO 4	49
4.1 ACTORES DEL SISTEMA	49
4.2 REQUISITOS DE USUARIO	50
4.2.1 LISTADO DE CASOS DE USO	50
4.2.2 ESPECIFICACIÓN DE CASOS DE USO	53
4.3 REGLAS DE NEGOCIO	56

4.4 REQUISITOS FUNCIONALES.....	57
4.5 REQUISITOS NO FUNCIONALES	58
4.5.1 ATRIBUTOS DE CALIDAD.....	58
4.5.2 RESTRICCIONES DE IMPLEMENTACIÓN Y ARQUITECTURA.....	60
4.5.3 REQUISITOS DE INTERFACES EXTERNAS	60
4.6 MODELO CONCEPTUAL DE DATOS	60
4.6.1 MODELO ENTIDAD-RELACIÓN	60
4.6.2 DICCIONARIO DE DATOS	62
CAPÍTULO 5	67
5.1 ARQUITECTURA LÓGICA	67
5.2 ARQUITECTURA FÍSICA	68
5.3 MODELO LÓGICO DE DATOS	69
5.4 DIAGRAMA DE CLASES	72
5.5 DIAGRAMAS DE SECUENCIA.....	81
5.6 DISEÑO DE INTERFACES	87
CAPÍTULO 6	95
6.1 SERVIDOR DE APLICACIONES	95
6.2 SERVIDOR DE BASES DE DATOS.....	95
6.3 PRINCIPALES TECNOLOGÍAS UTILIZADAS.....	96
7.4 DIFICULTADES TÉCNICAS SOLVENTADAS.....	97
7.5 HERRAMIENTAS UTILIZADAS	101
CAPÍTULO 7	103
7.1 PRUEBAS DE LA ITERACIÓN I.....	103
7.2 PRUEBAS DE LA ITERACIÓN II.....	104
7.3 PRUEBAS DE LA ITERACIÓN III.....	106
7.4. PRUEBAS DE LA ITERACIÓN IV	107
CAPÍTULO 8	109
8.1 MANUAL DE INSTALACIÓN Y DESPLIEGUE.....	109
8.1.1 PREPARACIÓN DEL SERVIDOR.....	109
8.1.2 CONFIGURACIÓN DE GLASSFISH.....	111
8.1.3 DESPLIEGUE DE LA APLICACIÓN	113
8.2 MANUAL DE USUARIO	114
8.2.1 MANUAL DEL USUARIO SIN IDENTIFICAR.....	115
8.2.2 MANUAL DEL USUARIO NORMAL.....	116

8.2.3 MANUAL DEL USUARIO GESTOR	121
CAPÍTULO 9	123
9.1 CONOCIMIENTOS ADQUIRIDOS	123
9.2 LÍNEAS DE TRABAJO PARA EL FUTURO.....	124
BIBLIOGRAFÍA.....	127
ANEXO I	131
ANEXO II	133
ANEXO III	147
ANEXO IV	155

Índice de Figuras

FIGURA 1. EVOLUCIÓN DEL GSM AL LTE. FUENTE: CNMC BLOG	2
FIGURA 2. TURISMO AL EXTERIOR Y SU IMPACTO ECONÓMICO EN 2018. FUENTE: BARÓMETRO UNWTO	3
FIGURA 3. INFORME TECNOLOGÍA DIGITAL EN ESPAÑA. FUENTE: INFORME HOOTSUITE.	4
FIGURA 4. DIAGRAMA DE CARACTERÍSTICAS DE LA PLATAFORMA DE GESTIÓN DE ITINERARIOS GPSSM (PLATAFORMA SOFTWARE EMPRESARIAL).....	6
FIGURA 5. PANTALLA INICIAL DE LA APLICACIÓN.	11
FIGURA 6. PANTALLA EN LA QUE VEMOS CÓMO SE VA GUARDANDO NUESTRO RECORRIDO MIENTAS LO REALIZAMOS.	11
FIGURA 7. TIPOS DE PUNTOS DE INTERÉS QUE PODEMOS AÑADIR.	11
FIGURA 8. INCORPORACIÓN DE FOTOGRAFÍA AL PUNTO DE INTERÉS.	11
FIGURA 9. CONFIRMACIÓN DE UNA FOTOGRAFÍA UNA VEZ TOMADA.	11
FIGURA 10. LISTA DE TUS RUTAS.	11
FIGURA 11. ESTADÍSTICAS SOBRE EL RECORRIDO DE UNA RUTA.	11
FIGURA 12. PANTALLA INICIAL DE LA APLICACIÓN.	12
FIGURA 13. GRABACIÓN DE UN RECORRIDO.	12
FIGURA 14. CONTROL DE REANUDACIÓN/FINALIZACIÓN DE UN RECORRIDO.	12
FIGURA 15. DATOS PARA GUARDAR UNA RUTA FINALIZADA.	12
FIGURA 16. MENÚ DE LA APLICACIÓN.	13
FIGURA 17. PANTALLA DE AMIGOS EN LA APLICACIÓN.	13
FIGURA 18. ESTADÍSTICAS SOBRE UN RECORRIDO.	13
FIGURA 19. MENÚ DE LA APLICACIÓN.	14
FIGURA 20. PANTALLA DE GRABACIÓN DE RECORRIDO.	14
FIGURA 21. RESULTADO DE UN RECORRIDO.....	14
FIGURA 22. ESTADÍSTICAS OBTENIDAS EN UN RECORRIDO.	14
FIGURA 23. MENÚ DE RUTAS.....	15
FIGURA 24. CONTROL DEL TRANCURSO DE RUTA.	15
FIGURA 25. SEGUIMIENTO DE RUTAS.	15
FIGURA 26. CREACIÓN DE CONTENIDO MULTIMEDIA.....	15
FIGURA 27. MENÚ DE LA APLICACIÓN.	16
FIGURA 28. MENÚ DE CREACIÓN DE RUTA.	16
FIGURA 29. CREACIÓN DE PUNTO DE INTERÉS.....	16
FIGURA 30. VISUALIZACIÓN DE RUTA.	16
FIGURA 31. UBICACIÓN DEL DISPOSITIVO EN EL MAPA.....	17
FIGURA 32. MENÚ DE LA APLICACIÓN.	17
FIGURA 33. PROGRAMACIÓN DE LA NAVEGACIÓN.	17
FIGURA 34. EXPLORACIÓN DE ESTABLECIMIENTOS CERCANOS.....	17
FIGURA 35. PUNTOS DE INTERÉS Y NEGOCIOS EN EL MAPA.	17
FIGURA 36. PUNTOS DE INTERÉS Y NEGOCIOS EN EL MAPA.	17
FIGURA 37. PUNTOS DE INTERÉS Y NEGOCIOS EN EL MAPA.	17

FIGURA 38. MAPAS SIN CONEXIÓN.	17
FIGURA 39. PANTALLA DE INICIO DE LA APLICACIÓN.	19
FIGURA 40. NOTIFICACIÓN DE PUNTO DE INTERÉS CERCANO.	19
FIGURA 41. LISTA DE PUNTOS DE INTERÉS NOTIFICADOS.	19
FIGURA 42. MAPA SOBRE EL QUE SE REPRESENTA LA POSICIÓN DEL PUNTO DE INTERÉS NOTIFICADO.	19
FIGURA 43. MENSAJE AL ACERCARSE A UN PUNTO DE INTERÉS.	19
FIGURA 44. PANTALLA DE INICIO DE LA APLICACIÓN.	20
FIGURA 45. LECTOR DE CÓDIGOS QR.	20
FIGURA 46. MAPA PARA EXPLORAR LOS PUNTOS DE INTERÉS.	20
FIGURA 47. IMAGEN E INFORMACIÓN DE PUNTOS DE INTERÉS.	20
FIGURA 48. RECOPIACIÓN DE IMÁGENES DE UN PUNTO.	21
FIGURA 49. PLANO ESQUEMÁTICO DE DESPLAZAMIENTOS A PIE.	22
FIGURA 50. MENÚ DE LA APLICACIÓN.	22
FIGURA 51. INFORMACIÓN PRINCIPAL DE UN PUNTO DE INTERÉS.	22
FIGURA 52. ENLACE A GOOGLE MAPS PARA NAVEGACIÓN.	22
FIGURA 53. PANTALLA INICIAL DE LA APLICACIÓN.	23
FIGURA 54. LISTA DE GUÍAS DISPONIBLES PARA DESCARGAR.	23
FIGURA 55. MENÚ DE LA APLICACIÓN.	23
FIGURA 56. INFORMACIÓN DE DESCARGA DE LA GUÍA.	23
FIGURA 57. LISTA DE PUNTOS DE INTERÉS DE LA GUÍA DE UNA CIUDAD.	23
FIGURA 58. PUNTOS DE INTERÉS REPRESENTADOS EN UN MAPA.	23
FIGURA 59. INFORMACIÓN SOBRE UN PUNTO DE INTERÉS. TEXTO, IMAGEN Y AUDIO.	23
FIGURA 60. OPCIONES SOBRE UN PUNTO DE INTERÉS.	23
FIGURA 61. LISTADO DE GUÍAS DESTACADAS DISPONIBLES.	25
FIGURA 62. MENÚ DE OPCIONES DE LA APLICACIÓN.	25
FIGURA 63. PLANIFICADOR PARA LA VISITA A LA CIUDAD, ORGANIZADO EN DÍAS DE VISTA.	25
FIGURA 64. EJEMPLO DE GUÍA.	25
FIGURA 65. OPCIONES DE LA GUÍA SOBRE ACTIVIDADES Y MOVILIDAD.	25
FIGURA 66. CLASIFICACIÓN DENTRO DE LA OPCIÓN DE VISITAS Y ACTIVIDADES.	25
FIGURA 67. FICHA DE UN LUGAR DE INTERÉS.	25
FIGURA 68. PANTALLA INICIAL, CON NUESTROS VIAJES Y LA OPCIÓN DE CREAR NUEVOS.	26
FIGURA 69. LOS VIAJES SE IDENTIFICAN POR EL LUGAR DE PARTIDA.	26
FIGURA 70. PODEMOS MARCAR UN VIAJE COMO PÚBLICO O PRIVADO, INVITAR COMPAÑEROS, COMPARTIRLO EN REDES SOCIALES, PERMITIR COMENTARIOS, ETC.	26
FIGURA 71. EN LOS VIAJES PODEMOS AÑADIR NOTAS, FOTOS, DATOS DE CLIMA, UBICACIONES, ETC.	26
FIGURA 72. MENÚ DE LA APLICACIÓN.	27
FIGURA 73. BÚSQUEDA DE MAPAS PARA DESCARGAR.	27
FIGURA 74. VISTA PRINCIPAL DE UNA CIUDAD.	27
FIGURA 75. MAPA DESCARGADO DE UNA CIUDAD.	27
FIGURA 76. VISUALIZACIÓN DE UN PUNTO DE INTERÉS.	28
FIGURA 77. FOTOS Y COMENTARIOS DE UN PUNTO DE INTERÉS.	28
FIGURA 78. PANTALLA INICIAL QUE NOS PERMITE BUSCAR DESTINO.	29
FIGURA 79. DEBAJO DE LA PANTALLA PRINCIPAL TENEMOS SUGERENCIAS DE DESTINOS.	29
FIGURA 80. DESTINO Y SUS LUGARES DE INTERÉS.	29
FIGURA 81. ESTABLECIMIENTOS HOTELEROS.	29

FIGURA 82. DEBLOQUEO DE RETOS.....	29
FIGURA 83. SUGERENCIAS DE DESTINOS.	29
FIGURA 84. CATEGORÍAS DE LA APLICACIÓN.	30
FIGURA 85. LISTADO DE GUÍAS DE LOS MUNICIPIOS DE MADRID.	30
FIGURA 86. PANTALLA DE INFORMACIÓN Y DESCARGA DE GUÍA.	30
FIGURA 87. MAPA CON PUNTOS DE INTERÉS DE UNA GUÍA.	30
FIGURA 88. OPCIONES SOBRE GPS Y SIMULACIÓN.	31
FIGURA 89. CATEGORÍAS DE INFORMACIÓN.	31
FIGURA 90. ELECCIÓN DE ETAPAS DEL CAMINO.	32
FIGURA 91. COMIENZO DE LA PRIMERA ETAPA CON PERSONAJES VIRTUALES.	32
FIGURA 92. ELECCIÓN ENTRE TÁCTIL O GAFAS DE REALIDAD VIRTUAL PARA USAR LA APP.	32
FIGURA 93. VISITA A UN MONUMENTO E INFORMACIÓN SOBRE ESTE.....	32
FIGURA 94. MOMENTO DE UNA ETAPA DONDE SE PUEDE ELEGIR PUNTOS DE INTERÉS A VISITAR.....	32
FIGURA 95. MAPA DE UNA ETAPA CON SUS LOCALIDADES Y PUNTOS DE INTERÉS.	32
FIGURA 96. LISTA DE PUNTOS DE INTERÉS CERCANOS.....	33
FIGURA 97. PUNTOS DE INTERÉS DIBUJADOS EN UN MAPA.	33
FIGURA 98. EXPLORACIÓN DEL ENTORNO CON LA CÁMARA.	33
FIGURA 99. INFORMACIÓN DE UN PUNTO DE INTERÉS.	33
FIGURA 100. DIAGRAMA DE INCREMENTOS Y FASES.	36
FIGURA 101. DETALLE DE PLANIFICACIÓN DE ETAPAS.	37
FIGURA 102. GANTT DE PLANIFICACIÓN DE LA PRIMERA ITERACIÓN.....	38
FIGURA 103. GANTT DE PLANIFICACIÓN DE LA SEGUNDA ITERACIÓN.....	38
FIGURA 104. GANTT DE PLANIFICACIÓN DE LA TERCERA ITERACIÓN.	38
FIGURA 105. GANTT DE PLANIFICACIÓN DE LA CUARTA ITERACIÓN.	39
FIGURA 106. DETALLE TIEMPOS REALES EMPLEADOS EN LAS DISTINTAS ETAPAS.....	47
FIGURA 107. JERARQUÍA ENTRE ACTORES DEL SISTEMA.	50
FIGURA 108. EJEMPLO DE RELACIONES ENTRE CASOS DE USO.....	52
FIGURA 109. DIAGRAMA DE CASOS DE USO.	52
FIGURA 110. DIAGRAMA ENTIDAD-RELACIÓN.	61
FIGURA 111. ARQUITECTURA LÓGICA.	68
FIGURA 112. ARQUITECTURA FÍSICA.	69
FIGURA 113. MODELO RELACIONAL.	70
FIGURA 114. ENTIDADES AMISTADPK, AMISTAD Y UUSARIO.	72
FIGURA 115. ENTIDADES ITINERARIO Y POSICIONES.	73
FIGURA 116. ENTIDADES USERS Y USERGROUPS	73
FIGURA 117. ENTIDAD POI.....	74
FIGURA 118. CLASES ABSTRACTFACADE, APPLICATIONCONFIG Y USUARIOFACADEREST.....	75
FIGURA 119. CLASES AMISTADFACADEREST Y POSICIONESFACADEREST.....	75
FIGURA 120. CLASES ITINERARIOFACADEREST Y POIFACADEREST.....	76
FIGURA 121. CLASES ITINERARIOBACKINGBEAN, POSICIONESBACKINGBEAN Y POIBACKINGBEAN.	77
FIGURA 122. CLASES AMISTADBACKINGBEAN Y USUARIOBACKINGBEAN.....	77
FIGURA 123. AMISTADCLIENTBEAN Y USUARIOCLIENTBEAN.	78
FIGURA 124. CLASES POIClientBEAN Y IMAGECONTROLLER.....	78
FIGURA 125. CLASE ITINERARIOCLIENTBEAN.....	79
FIGURA 126. CLASES AMISTADREADER, AMISTADWRITER, ITINERARIOREADER E ITINERARIO WRITER.....	80

FIGURA 127. CLASES POSICIONESREADER, POIWRITER, POIREADER, USUARIO WRITER. Y USUARIOREADER.	80
FIGURA 128. LOGINVIEW, AUTHENTICATIONUTILS Y USEREJB.	81
FIGURA 129. DIAGRAMA DE SECUENCIA SOBRE REGISTRO DE ITINERARIO.	82
FIGURA 130. DIAGRAMA DE SECUENCIA SOBRE ELIMINAR ITINERARIO.	82
FIGURA 131. DIAGRAMA DE SECUENCIA SOBRE EDITAR ITINERARIO.	83
FIGURA 132. DIAGRAMA DE SECUENCIA SOBRE GESTIONAR ITINERARIOS.	83
FIGURA 133. DIAGRAMA DE SECUENCIA SOBRE BUSCAR ITINERARIO.	84
FIGURA 134. DIAGRAMA DE SECUENCIA DE ELIMINAR UNA AMISTAD.	84
FIGURA 135. DIAGRAMA DE SECUENCIA DE RESOLVER SOLICITUD.	85
FIGURA 136. DIAGRAMA DE SECUENCIA SOBRE LOGIN.	86
FIGURA 137. DIAGRAMA DE SECUENCIA SOBRE REGISTRO.	86
FIGURA 138. DIAGRAMA DE SECUENCIA DE CIERRE DE SESIÓN.	87
FIGURA 139. INTERFAZ GENERAL DE LA APLICACIÓN.	88
FIGURA 140. MENÚS DE LA APLICACIÓN.	88
FIGURA 141. FRAGMENTO DE CÓDIGO CON CRITERIAQUERY.	98
FIGURA 142. FRAGMENTO DE CÓDIGO CON UNITOFWORK.	99
FIGURA 143. FRAGMENTO DE CÓDIGO CON LIBRERÍAS DE JACKSON OBJECTMAPPER.	99
FIGURA 144. FRAGMENTO DE CÓDIGO DE VISTA ITINERARIO.XHTML	101
FIGURA 145. PANTALLA DE ADMINISTRACIÓN DE GLASSFISH.	110
FIGURA 146. CONFIGURACIÓN DE PUERTO DE SALIDA HTTP.	111
FIGURA 147. CONFIGURACIÓN DE 'POLL' DE CONEXIONES.	112
FIGURA 148. RECURSOS JDBC AÑADIDOS A GLASSFISH.	112
FIGURA 149. RECURSOS JDBC AÑADIDOS A GLASSFISH.	113
FIGURA 150. DESPLIEGUE DE LA APLICACIÓN EN GLASSFISH.	114
FIGURA 151. MENÚS DE LA APLICACIÓN.	115
FIGURA 152. PANTALLA DE REGISTRO.	116
FIGURA 153. PANTALLA DE LOGIN.	116
FIGURA 154. PANTALLA PRINCIPAL O DE MIS ITINERARIOS.	117
FIGURA 155. PANTALLA DE CONSULTA DE ITINERARIO.	117
FIGURA 156. PANTALLA CON MAPA DEL ITINERARIO.	118
FIGURA 157. PANTALLA DE EDICIÓN DE ITINERARIO.	118
FIGURA 158. PANTALLA DE BÚSQUEDA DE ITINERARIOS.	119
FIGURA 159. PANTALLA DE AMISTADES.	119
FIGURA 160. PANTALLA PERFIL DE USUARIO.	119
FIGURA 161. PANTALLA DE BÚSQUEDA DE USUARIOS.	120
FIGURA 162. PANTALLA DE VER PERFIL DE USUARIO BUSCADO.	120
FIGURA 163. PANTALLA DE GESTIÓN DE PUNTOS DE INTERÉS.	121
FIGURA 164. PANTALLA DE EDICIÓN DE PUNTOS DE INTERÉS.	121
FIGURA 165. PANTALLA DE CREACIÓN DE UN PUNTO DE INTERÉS.	122
FIGURA 166. PANTALLA DE CREACIÓN DE UN PUNTO DE INTERÉS II.	122

Índice de tablas

TABLA 1. CLASIFICACIÓN DE LAS APPS A DESCRIBIR.	9
TABLA 2. CARACTERÍSTICAS DE WIKILOC.	10
TABLA 3. CARACTERÍSTICAS DE STRAVA.	12
TABLA 4. CARACTERÍSTICAS DE ENDOMONDO.	13
TABLA 5. CARACTERÍSTICAS DE ASWINGS.	14
TABLA 6. CARACTERÍSTICAS DE ORUXMAPS.	15
TABLA 7. CARACTERÍSTICAS DE GOOGLE MAPS.	16
TABLA 8. CARACTERÍSTICAS DE TOURSNAP.	18
TABLA 9. CARACTERÍSTICAS DE EL HIERRO.	20
TABLA 10. CARACTERÍSTICAS DE METROMINUTO.	21
TABLA 11. CARACTERÍSTICAS DE PAT CITY GUIDES.	22
TABLA 12. CARACTERÍSTICAS DE TRIPWOLF.	24
TABLA 13. CARACTERÍSTICAS DE JOURNI.	26
TABLA 14. CARACTERÍSTICAS DE CITYMAPS2GO.	27
TABLA 15. CARACTERÍSTICAS DE MINUBE.	28
TABLA 16. CARACTERÍSTICAS DE GUÍA MADRID 5D.	30
TABLA 17. CARACTERÍSTICAS DE SANTIAGO 360º.	31
TABLA 18. CARACTERÍSTICAS DE SENDITUR.	33
TABLA 19. PUNTOS DE FUNCIÓN SIN AJUSTAR.	41
TABLA 20. FACTORES DE AJUSTE Y PESOS.	42
TABLA 21. EQUIVALENCIA ENTRE PUNTO DE FUNCIÓN Y LÍNEAS DE CÓDIGO. FUENTE QSM.	42
TABLA 22. LÍNEAS DE CÓDIGO Y ESTIMACIONES FINALES.	42
TABLA 23. COEFICIENTES COCOMO. FUENTE WIKIPEDIA.	43
TABLA 24. FACTORES DE ESFUERZO COCOMO.	44
TABLA 25. PRESUPUESTO DE HARDWARE.	45
TABLA 26. PRESUPUESTO DE SOFTWARE.	45
TABLA 27. PRESUPUESTO TOTAL.	46
TABLA 28. LISTA DE ACTORES DEL SISTEMA.	49
TABLA 29. LISTA DE CASOS DE USO.	51
TABLA 30. ESPECIFICACIÓN CASO DE USO GS1.	53
TABLA 31. ESPECIFICACIÓN CASO DE USO GS2.	54
TABLA 32. ESPECIFICACIÓN CASO DE USO GS3.	55
TABLA 33. ESPECIFICACIÓN CASO DE USO GS4.	55
TABLA 34. REGLAS DE NEGOCIO.	56
TABLA 35. REQUISITOS FUNCIONALES ASOCIADOS AL CASO DE USO GS1.	57
TABLA 36. REQUISITOS FUNCIONALES ASOCIADOS AL CASO DE USO GS2.	57
TABLA 37. REQUISITOS FUNCIONALES ASOCIADOS AL CASO DE USO GS3.	58
TABLA 38. REQUISITOS FUNCIONALES ASOCIADOS AL CASO DE USO GS4.	58

TABLA 39. ATRIBUTOS DE CALIDAD DE USABILIDAD.	59
TABLA 40. ATRIBUTOS DE CALIDAD DE RENDIMIENTO	59
TABLA 41. ATRIBUTOS DE CALIDAD DE SEGURIDAD.	59
TABLA 42. ATRIBUTOS DE CALIDAD DE DISPONIBILIDAD.....	59
TABLA 43. ATRIBUTOS DE CALIDAD DE ESCALABILIDAD.	59
TABLA 44. RESTRICCIONES DE IMPLEMENTACIÓN.	60
TABLA 45. REQUISITOS SOBRE INTERFACES DE INTERFACES EXTERNAS.	60
TABLA 46. ATRIBUTOS ENTIDAD USUARIO.	62
TABLA 47. ATRIBUTOS ENTIDAD ITINERARIO.....	63
TABLA 48. ATRIBUTOS ENTIDAD POSICIONES.....	63
TABLA 49. ATRIBUTOS ENTIDAD PUNTOSMULTIMEDIA.	63
TABLA 50. ATRIBUTOS ENTIDAD POI.....	64
TABLA 51. ATRIBUTOS ENTIDAD PRA.....	64
TABLA 52. ATRIBUTOS ENTIDAD AMISTAD.	66
TABLA 53. TRIGGERS DE LA BASE DE DATOS.....	71
TABLA 54. DISEÑO DE INTERFAZ DE LOGIN.	89
TABLA 55. DISEÑO DE INTERFAZ DE REGISTRO.	89
TABLA 56. DISEÑO DE INTERFAZ PANEL DE USUARIO.....	90
TABLA 57. DISEÑO DE INTERFAZ VER ITINERARIO.	90
TABLA 58. DISEÑO INTERFAZ BUSCAR USUARIO.	91
TABLA 59. DISEÑO INTERFAZ AMIGOS.....	91
TABLA 60. DISEÑO INTERFAZ VER PERFIL.	92
TABLA 61. DISEÑO INTERFAZ BUSCAR USUARIO.	92
TABLA 62. DISEÑO INTERFAZ BUSCAR GESTIÓN DE PUNTOS DE INTERÉS.	93
TABLA 63. DISEÑO INTERFAZ BUSCAR AÑADIR PUNTO DE INTERÉS.	93
TABLA 64. PRUEBA DEL SISTEMA P01.	103
TABLA 65. PRUEBA DEL SISTEMA P02.	103
TABLA 66. PRUEBA DEL SISTEMA P03.	103
TABLA 67. PRUEBA DEL SISTEMA P03B.	104
TABLA 68. PRUEBA DEL SISTEMA P04.	104
TABLA 69. PRUEBA DEL SISTEMA P05.	104
TABLA 70. PRUEBA DEL SISTEMA P06.	104
TABLA 71. PRUEBA DEL SISTEMA P07.	104
TABLA 72. PRUEBA DEL SISTEMA P08.	105
TABLA 73. PRUEBA DEL SISTEMA P08B.	105
TABLA 74. PRUEBA DEL SISTEMA P08C.	105
TABLA 75. PRUEBA DEL SISTEMA P09.	105
TABLA 76. PRUEBA DEL SISTEMA P10.	105
TABLA 77. PRUEBA DEL SISTEMA P10B.	106
TABLA 78. PRUEBA DEL SISTEMA P11.	106
TABLA 79. PRUEBA DEL SISTEMA P12.	106
TABLA 80. PRUEBA DEL SISTEMA P13.	106
TABLA 81. PRUEBA DEL SISTEMA P14.	107
TABLA 82. PRUEBA DEL SISTEMA P15.	107
TABLA 83. PRUEBA DEL SISTEMA P16.	107

TABLA 84. PRUEBA DEL SISTEMA P17.	107
TABLA 85. PRUEBA DEL SISTEMA P18.	107
TABLA 86. PRUEBA DEL SISTEMA P19.	108
TABLA 87PRUEBA DEL SISTEMA P19B.	108
TABLA 88. CASO DE USO UGI1.	133
TABLA 89. CASO DE USO UGI2.	134
TABLA 90. CASO DE USO UGR1.	135
TABLA 91. CASO DE USO UGR2.	135
TABLA 92. CASO DE USO UGR3.	136
TABLA 93. CASO DE USO UGR4.	136
TABLA 94. CASO DE USO UGR5.	137
TABLA 95. CASO DE USO UGR6.	137
TABLA 96. CASO DE USO UGR7.	138
TABLA 97. CASO DE USO UGR8.	138
TABLA 98. CASO DE USO UGR9.	139
TABLA 99. CASO DE USO UGR10.	139
TABLA 100. CASO DE USO UGR11.	140
TABLA 101. CASO DE USO UGR12.	140
TABLA 102. CASO DE USO UGR13.	141
TABLA 103. CASO DE USO UGR14.	142
TABLA 104. CASO DE USO UWR1.	142
TABLA 105. CASO DE USO PM1.	143
TABLA 106. CASO DE USO PM2.	143
TABLA 107. CASO DE USO PM3.	144
TABLA 108. CASO DE USO PM4.	144
TABLA 109. CASO DE USO PM5.	145
TABLA 110. CASO DE USO PM6.	145
TABLA 111. CASO DE USO PM7.	146
TABLA 112. REQUISITOS FUNCIONALES EN CU UGI1.	147
TABLA 113. REQUISITOS FUNCIONALES EN CU UGI2.	148
TABLA 114. REQUISITOS FUNCIONALES EN CU UGR1.	148
TABLA 115. REQUISITOS FUNCIONALES EN CU UGR2.	148
TABLA 116. REQUISITOS FUNCIONALES EN CU UGR3.	149
TABLA 117. REQUISITOS FUNCIONALES EN CU UGR4.	149
TABLA 118. REQUISITOS FUNCIONALES EN CU UGR5.	149
TABLA 119. REQUISITOS FUNCIONALES EN CU UGR6.	149
TABLA 120. REQUISITOS FUNCIONALES EN CU UGR7.	150
TABLA 121. REQUISITOS FUNCIONALES EN CU UGR8.	150
TABLA 122. REQUISITOS FUNCIONALES EN CU UGR9.	150
TABLA 123. REQUISITOS FUNCIONALES EN CU UGR10.	151
TABLA 124. REQUISITOS FUNCIONALES EN CU UGR11.	151
TABLA 125. REQUISITOS FUNCIONALES EN CU UGR12.	151
TABLA 126. REQUISITOS FUNCIONALES EN CU UGR13.	152
TABLA 127. REQUISITOS FUNCIONALES EN CU UGR14.	152
TABLA 128. REQUISITOS FUNCIONALES EN CU UWR1.	152

TABLA 129. REQUISITOS FUNCIONALES EN CU PM1.....	153
TABLA 130. REQUISITOS FUNCIONALES EN CU PM2.....	153
TABLA 131. REQUISITOS FUNCIONALES EN CU PM3.....	153
TABLA 132. REQUISITOS FUNCIONALES EN CU PM4.....	153
TABLA 133. REQUISITOS FUNCIONALES EN CU PM5.....	154
TABLA 134. REQUISITOS FUNCIONALES EN CU PM6.....	154
TABLA 135. REQUISITOS FUNCIONALES EN CU PM7.....	154
TABLA 136. CONTENIDO DEL CD.....	155

Capítulo 1

Introducción

Hace 45 años Motorola lanzaba, como un producto revolucionario, el primer teléfono móvil. Hoy en día, aquel primer dispositivo que produjo una revolución de las comunicaciones al permitir comunicarse telefónicamente de forma inalámbrica, es un producto totalmente obsoleto. La actualidad tecnológica ha sobrepasado completamente el concepto de teléfono móvil usado para comunicaciones telefónicas, y estamos en una consolidada revolución de los dispositivos móviles. Esta revolución surge apoyada en dos pilares fundamentales, en el nacimiento del concepto de 'Smartphone' y la mejora de las redes de telefonía para dar soporte a la navegación de banda ancha para los dispositivos móviles.

El diccionario de Google, define el término 'Smartphone' como: 'Teléfono celular con pantalla táctil, que permite al usuario conectarse a Internet, gestionar cuentas de correo electrónico e instalar otras aplicaciones y recursos a modo de pequeño computador'. Aunque es una definición correcta, podríamos añadir que el Smartphone es la herramienta que en los últimos años ha potenciado un acceso (casi) universal a Internet. En conclusión, un Smartphone es una ventana abierta permanentemente a cualquier tipo de información, procedente de cualquier parte del mundo, y desde cualquier lugar, gracias a una conexión a Internet.

La popularización y crecimiento de los Smartphones no hubiera sido posible si no hubiera sido acompañado de una evolución tecnológica de las redes de telefonía. El comienzo de la comunicación móvil de forma digital se produjo con la tecnología GSM o 2G, que mediante ondas de radio permitía transmitir información de un dispositivo a otro, identificándolos de forma única en la red a través de una tarjeta 'SIM'. El siguiente paso importante fue GPRS, donde se incluyó la conmutación de paquetes. Cada nueva actualización de las tecnologías que sustentan la comunicación entre dispositivos móviles (*ver Figura 1*) ha ido incluyendo sobre todo mejoras en la velocidad de transferencia de datos, favoreciendo cada vez un mayor intercambio de volúmenes de información.

Como consecuencia de la oportunidad de intercambiar información desde cualquier lugar, con la facilidad ofrecida por los dispositivos móviles en la actualidad, se ha popularizado el intercambio de contenidos multimedia y otra información personal entre los usuarios, a través de herramientas como las redes sociales, y otras aplicaciones dedicadas al ocio y el entretenimiento de los usuarios.

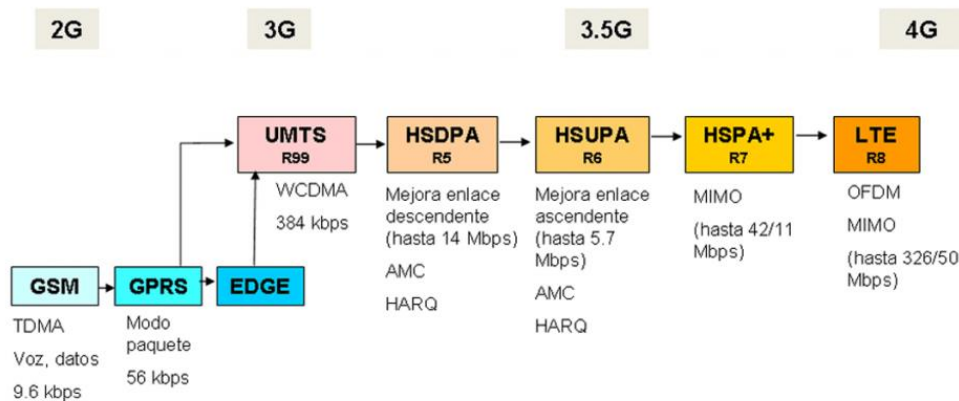


Figura 1. Evolución del GSM al LTE. Fuente: [CNMC Blog¹](#)

Es en nuestro tiempo de ocio, cuando las personas nos hemos contagiado más de esta forma de relación con el entorno. Sí antes realizábamos fotografías de nuestros viajes para guardarlas en un álbum, y quizás enseñárselas a algún visitante en nuestra casa, ahora publicamos en internet los contenidos de lo que hemos hecho recientemente para compartirlo con nuestras redes de contactos, y también para poder disponer de este contenido desde cualquier lugar.

Esta nueva forma de compartir nuestro tiempo de ocio ha abierto importantes oportunidades en el mercado turístico, ya que este ofrece un contexto perfecto para la generación de contenidos que los usuarios desean guardar y compartir, así como el deseo de proporcionar contenidos a los usuarios para guiarlos hacia ubicaciones determinadas por un interés comercial o cultural.

Si bien la importancia de los dispositivos móviles para cumplir esta misión es evidente, es importante señalar que para permitir la centralización y disponibilidad de la información que creamos y compartimos con nuestros Smartphones, son necesarias tecnologías en el lado servidor que nos permitan realizar todas las operaciones asociadas a la gestión de información.

1.1 Motivación

La motivación del proyecto GPSSM (GPS Sightseeing Moments) es desarrollar una plataforma que permita la gestión contenidos creados a través de dispositivos móviles. Concretamente, el proyecto busca gestionar la información que los usuarios generan durante sus viajes turísticos.

En la actualidad el turismo es un gran motor económico mundial gracias a la globalización, el abaratamiento de los costes de viaje (aerolíneas de bajo coste, plataformas para compartir

¹ Evolución del GSM al LTE según CNMC Blog: <https://blog.cnmc.es/2010/05/21/conceptos-basicos-del-telecom-evolucion-de-las-comunicaciones-moviles-del-gsm-al-lte/>

transporte...) y alojamiento (albergues, Hostels, Airbandb, Bed&Breakfast...) y la apertura de espacios de libre circulación de ciudadanos como Schengen. El turismo llega a representar un alto porcentaje del PIB en algunos países como España, en cuya serie contable 1995-2003, supuso una media del 10,92% según el Instituto Nacional de Estadística². En la *Figura 2*, puede observarse el impacto económico (en billones de euros) que dejan los turistas de algunos de los países que más turismo exterior realizan.

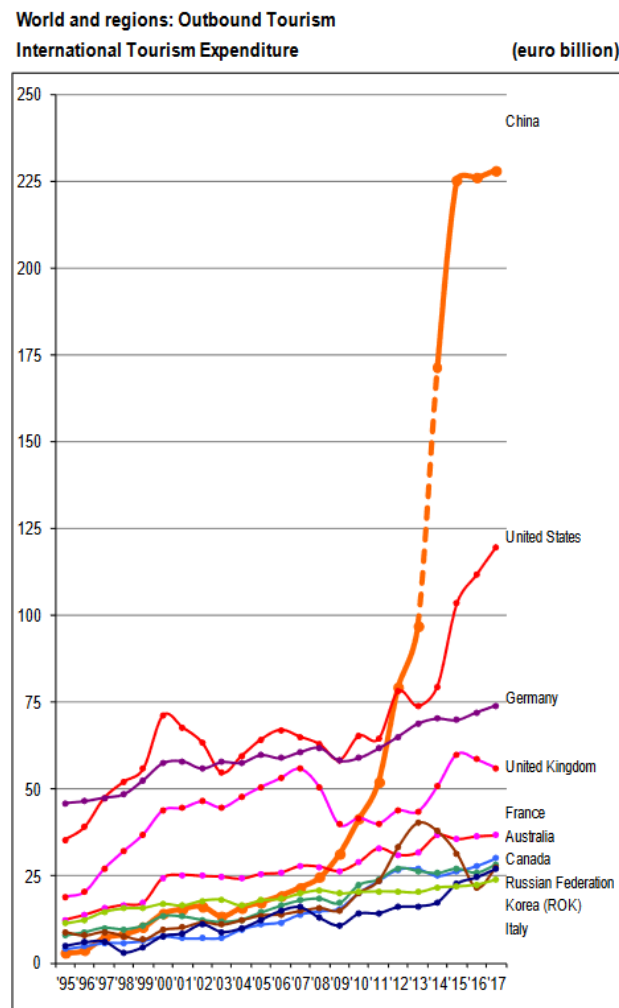


Figura 2. Turismo al exterior y su impacto económico en 2018. Fuente: [Barómetro UNWTO](#)

Es un hecho demostrado por los datos la popularización del turismo en nuestros días, al igual que lo es que la creación de contenidos multimedia desde dispositivos móviles allí donde se encuentran los usuarios, principalmente compartidos mediante redes sociales. La penetración de estas plataformas supera la mitad del total de la población en países desarrollados como España (ver *Figura 3*).

² Dato obtenido de la web de Instituto Nacional de Estadística:
<http://www.ine.es/jaxi/Datos.htm?path=/t35/p011/9503/l0/&file=01001.px>

Un dato muy relevante (obtenido del informe 2018 de Hootsuite para la penetración digital en el mundo³) es que en España 23 millones de usuarios móviles usan las redes sociales en sus dispositivos.



Figura 3. Informe tecnología digital en España. Fuente: [Informe Hootsuite](#).

Aprovechando la oportunidad que suponen la explosión tecnológica de los dispositivos móviles y las redes sociales, sumado al creciente impulso del turismo, el proyecto 'GPSSM' busca crear una plataforma que de soporte a la creación y gestión de información durante los recorridos turísticos de los usuarios de plataformas móviles.

La concepción de GPSSM se sustenta sobre el concepto de 'Itinerario', entendido como una recolección de las posiciones que ha seguido el usuario a lo largo de su recorrido turístico, junto con los contenidos multimedia creados con su dispositivo móvil en este trayecto.

El proyecto GPSSM, está dividido en dos partes claramente diferenciadas. Una aplicación empresarial que disponga y mantenga toda la información sobre rutas turísticas, con los servicios necesarios para tratar esta información a través de la propia plataforma u a través de dispositivos. La otra parte se trata de una aplicación móvil para dispositivos Android, desde la cual se realizará la creación de los contenidos turísticos del usuario. Esta parte constituye otro TFG desarrollado por el alumno Pablo Hernando García.

El presente documento está correspondiente a la parte de la aplicación empresarial, denominada 'Plataforma de gestión de Itinerarios GPSSM'.

³ Informe de Hootsuite sobre penetración digital en el mundo en 2018: <https://hootsuite.com/es/pages/digital-in-2018>

1.2 Objetivos

El objetivo de la Plataforma de Gestión de Itinerarios GPSSM es digitalizar y almacenar toda la información generada por un usuario durante sus recorridos turísticos para poder mejorar y compartir su experiencia.

Para alcanzar este objetivo, se pretenden alcanzar cada una de estas metas:

- Permitir a los usuarios mantener centralizada y organizada la información sobre sus recorridos.
- Proporcionar servicios de almacenamiento de la información de itinerarios, soportando las operaciones de creación, consulta, actualización y borrado, así como la ejecución de estas operaciones desde dispositivos móviles.
- Propiciar la creación y gestión de puntos de interés que puedan resultar interesantes al usuario para enriquecer su experiencia turística.
- Ofrecer una interfaz donde poder obtener una visualización su información turística accesible desde un navegador web.

1.3 Descripción del proyecto

La Plataforma de gestión de Itinerarios GPSSM, es un software empresarial que permite a los usuarios persistir la información de sus recorridos o itinerarios turísticos, pudiendo consultarlos, retocarlos o eliminarlos siempre que deseen. La plataforma permite estas acciones desde un dispositivo móvil, gracias a los servicios que esta proporciona, o desde el entorno web de la propia plataforma. Aunque la creación de contenido queda restringido a los dispositivos móviles, debido a las necesidades de servicios de ubicación (GPS) o de creación de contenido multimedia (cámara de fotos) que poseen los Smartphones.

La plataforma a su vez proporciona un complemento a la experiencia turística del usuario, proporcionando una serie de coordenadas o 'Puntos de Interés'. Estos puntos serán lugares de especial interés cultural o turístico, y serán gestionados desde la plataforma por un administrador. Además, la plataforma permitirá establecer amistades, de forma que los usuarios puedan acceder fácilmente a los contenidos creados por otros usuarios.

Para que los usuarios puedan mantener el control de su información, los Itinerarios podrán recibir una privacidad establecida en tres niveles. 'Público', si cualquiera puede acceder a su visualización, 'Amistad', si solo quiere que sea visible a sus amigos en la plataforma, y 'Privado', si no desea que nadie más pueda visualizar este recorrido.

La materialización de esta descripción, y de los objetivos descritos en el apartado anterior, pretende obtenerse mediante la implementación de una serie de características y subcaracterísticas, que pueden verse resumidas en el diagrama que se incluye en este mismo apartado (*ver figura 4*).

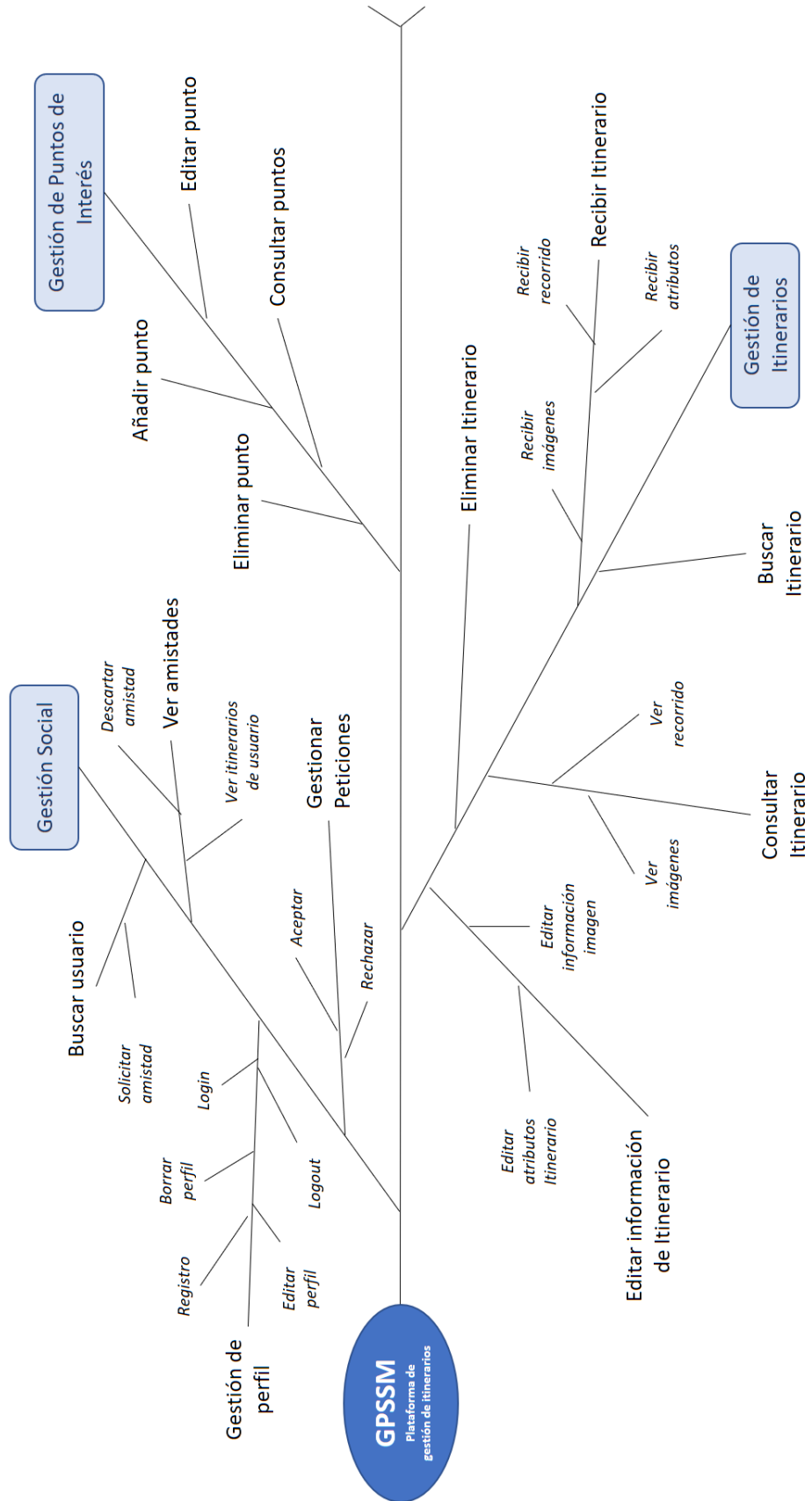


Figura 4. Diagrama de características de la Plataforma de Gestión de Itinerarios GPSSM (Plataforma Software Empresarial).

Estas características se organizan en tres ramas, que a su vez se dividen en varias subramas. Son las siguientes:

- **GESTIÓN DE ITINERARIOS:** esta rama del árbol comprende todo lo relacionado con los itinerarios turísticos de los usuarios. Contiene las siguientes subcaracterísticas.
 - Recibir Itinerario: permite recibir desde un dispositivo móvil toda la información de un itinerario, desde los atributos de este, a todas las posiciones que forman parte del recorrido y el contenido multimedia recogido en determinadas posiciones.
 - Ver Itinerario: permite la consulta de toda la información recogida en el Itinerario de un usuario, visualizando su recorrido sobre un mapa y donde han sido recogidos los contenidos multimedia.
 - Editar información: permite modificar algunos atributos descriptivos del itinerario, así como la información descriptiva de un contenido multimedia.
 - Eliminar itinerario: permite a un usuario ordenar que un itinerario deje de estar disponible en la plataforma.
 - Buscar Itinerario: permite al usuario buscar itinerarios de otros usuarios.

- **GESTIÓN DE PUNTOS DE INTERÉS:** esta rama engloba toda la gestión que realizará el administrador sobre los puntos de interés. Contiene las siguientes subcaracterísticas.
 - Consultar puntos: permite consultar todos los puntos de interés dados de alta en la plataforma, ya sea en la propia plataforma o proporcionándolos para su explotación.
 - Crear: permite dar de alta un punto de interés.
 - Editar: permite modificar los atributos informativos de un punto de interés.
 - Eliminar: permite borrar del sistema un punto de interés previamente dado de alta.

- **GESTIÓN SOCIAL:** esta rama comprende todo lo relacionado con los usuarios de la plataforma. Contiene las siguientes subcaracterísticas.
 - Gestión de perfil: engloba todas las operaciones asociadas a la cuenta de usuario que son; registro, login, logout, modificar perfil y borrar perfil.
 - Buscar usuario: permite la búsqueda de un usuario en la plataforma y solicitar su amistad si así se desea.
 - Ver amigos: permite acceder a un listado de amistades activas, así como ver los itinerarios de cada amigo, o revocar la relación de amistad.
 - Gestionar peticiones: permite recibir la notificación de que se ha producido una solicitud de relación de amistad, y recoger la respuesta del usuario a esta.

1.4 Estructura del documento

En esta sección se describe las distintas partes en las que se divide la presente memoria.

- **Capítulo 1:** Introducción. Este capítulo contiene la descripción y características principales del proyecto.
- **Capítulo 2:** Estado del arte. Contiene un profundo análisis de la tecnología similar presente en el mercado para medir la fortaleza del concepto de nuestra herramienta.
- **Capítulo 3:** Planificación del proyecto. Capítulo dedicado a las previsiones de costes temporales y económicos para el desarrollo del proyecto.
- **Capítulo 4:** Análisis. Detalla la funcionalidad que deberá cumplir el sistema a través de la especificación de requisitos.
- **Capítulo 5:** Diseño del sistema. En este capítulo se realiza una descripción de la arquitectura y los componentes especificados para la plataforma.
- **Capítulo 6:** Implementación. El capítulo trata sobre las tecnologías y librerías utilizadas, así como el entorno en que va a ser desplegado.
- **Capítulo 7:** Pruebas. Recoge las pruebas llevadas a cabo para comprobar la funcionalidad del sistema.
- **Capítulo 8:** Manuales. Contiene guías tanto para el despliegue de la aplicación como para su uso.
- **Capítulo 9:** Conclusiones. Recoge las reflexiones finales tras el desarrollo del proyecto y las líneas futuras a seguir para su ampliación y mejora.
- **BIBLIOGRAFÍA.** Libros, documentos y sitios web consultados durante el desarrollo del proyecto.
- **ANEXOS:**
 - **ANEXO I** – Glosario
 - **ANEXO II** – Casos de uso
 - **ANEXO III** – Requisitos funcionales
 - **ANEXO IV** – Estructura del CD

Capítulo 2

Estado del Arte

En esta sección se realiza un análisis sobre los productos y tecnologías, que existen actualmente, relacionados con la idea del proyecto. Se realiza un análisis de aplicaciones que pueden considerarse competencia, clasificadas según sus características.

Debido a que este proyecto está dividido en dos plataformas complementarias desarrolladas en dos trabajos de fin de grado distintos, se ha realizado un estudio conjunto y más completo del habitual sobre el estado del arte. Esto se debe a que la mayoría de herramientas están formadas por una plataforma empresarial y una aplicación móvil, por lo que se realizó un primer análisis conjuntamente sobre las herramientas previamente seleccionadas, procediendo después a plasmar los resultados y conclusiones de forma independiente en cada trabajo.

2.1 Análisis de competencia

Se proceden a analizar una serie de aplicaciones relacionadas con rutas turísticas o tracks GPS para aprender de ellas. Posteriormente se realizará una reflexión para obtener conclusiones.

Vamos a clasificar las distintas aplicaciones en 3 grupos según su objetivo:

- GRUPO I: aplicaciones con gestión de rutas.
- GRUPO II: aplicaciones de información turística o de interés.
- GRUPO III: aplicaciones con realidad aumentada, o realidad virtual.

GRUPO I	GRUPO II	GRUPO III
Wikiloc STRAVA Endomodo Aswings Oruxmaps Google maps	TourSnap El hierro Metrominuto PAT City Guides Tripwolf Journi CityMaps2Go Minube	Guía Madrid 5D Camino 360º Senditur

Tabla 1. Clasificación de las apps a describir.

Se procede al análisis de las aplicaciones de los distintos grupos, para obtener una comparativa y conclusiones.

Todas las aplicaciones móviles han sido probadas en dispositivos Android. Concretamente en tres modelos distintos: Xiaomi Redmi 4X pro versión EU (Android 6.0.1 MIUI), Motorola Moto G5 Plus (Android 7.0) o Huawei Honor 8 (Android 7.0).

2.1.1 Aplicaciones del Grupo I

En este apartado se describen las aplicaciones del grupo I, donde se engloban las aplicaciones que tienen características para gestionar rutas.

Wikiloc

Wikiloc es una aplicación que permite la grabación y el seguimiento de rutas, así como la definición y consulta de punto de interés dentro de la ruta.

Información técnica			
Coste	Gratis versión limitada	Opiniones	NO
Ámbito	Global	Red social	NO
SO	Android, IOS,	Puntos de interés	SI
Tamaño	22,9 MB	Alojamientos	NO
Idiomas	Español	Transportes	NO
GPS	SI	Tiempo	NO
Mapas	SI	Contenido multimedia	Fotos
Crear rutas	SI	Creación de multimedia	Fotos
Seguir rutas	SI (de pago)	Realidad aumentada	NO
Brújula	NO	Otros servicios	NO

Tabla 2. Características de Wikiloc.

Cuando se graba una ruta se indica el tipo de actividad a la que corresponde según se realiza (a pie, ciclismo, cicloturismo, quad, a caballo, etc), e indicamos si queremos activar el seguimiento en vivo (que otras personas sigan nuestra posición en tiempo real).

La grabación de rutas (*ver figura 6*) permite registrar nuestros movimientos sobre un mapa mientras nos desplazamos, permitiéndonos además realizar fotos y marcar puntos de interés (panorámica, monumento, etc) en nuestra posición.

Las rutas almacenadas pueden subirse al repositorio de Wikiloc o guardarse en local. También pueden consultarse todas las rutas del repositorio buscando por diferentes filtros (localización, tipo de actividad, etc). Además de consultarse estas rutas pueden ser recorridas mediante navegador GPS.

También proporciona una característica de destinos turísticos promocionados y otra de seguimiento de mapas fuera de línea (sin conexión a internet).

La aplicación fácil de utilizar, y posee un indicador de nuestra cobertura GPS, mostrando avisos de que no se registrará nuestra posición si esta es muy escasa.

Es destacable la posibilidad de poder añadir puntos de interés (ver figuras 7, 8 y 9), con varias fotografías en un mismo punto, aunque no se resalta de ninguna manera especial que el punto marcado de interés posea las fotografías.

El principal atractivo de la aplicación es poder seguir una ruta registrada y seguirla, mientras en tiempo real vemos como se dibuja nuestro propio recorrido.

Aunque la funcionalidad de la aplicación encaja en la propuesta del proyecto, esta parece estar centrada más en rutas de deporte, que en un uso para el turismo. El modo de navegación GPS conlleva un coste adicional (suscripción anual de 4,99€).



Figura 5. Pantalla inicial de la aplicación.



Figura 6. Pantalla en la que vemos cómo se va guardando nuestro recorrido mientras lo realizamos.

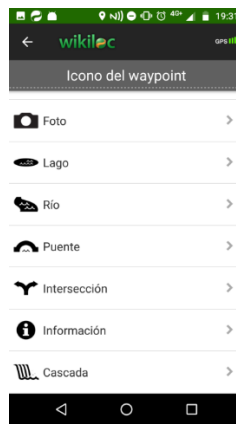


Figura 7. Tipos de puntos de interés que podemos añadir.



Figura 8. Incorporación de fotografía al punto de interés.

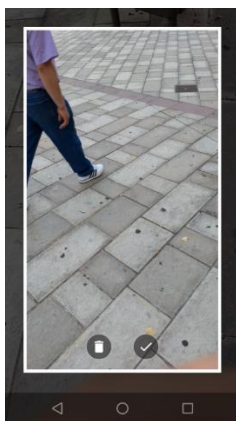


Figura 9. Confirmación de una fotografía una vez tomada.

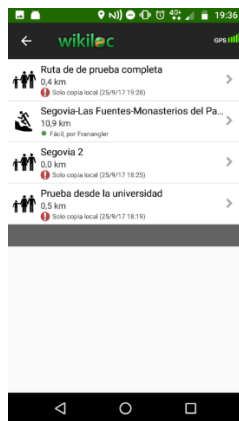


Figura 10. Lista de tus rutas.



Figura 11. Estadísticas sobre el recorrido de una ruta.

STRAVA

Se trata de una aplicación de ámbito global, enfocada a la monitorización en tiempo real de actividades deportivas como correr o ciclismo (entre otras).

Esta aplicación tiene una versión limitada gratuita, que puede ampliarse con una suscripción mensual (7,99/mes) o anual (59,99€/año).

Información técnica			
Coste	Gratis versión limitada	Opiniones	SI
Ámbito	Global	Red social	SI
SO	Android, IOS,	Puntos de interés	NO
Tamaño	52,74 MB	Alojamientos	NO
Idiomas	Español, Inglés	Transportes	NO
GPS	SI	Tiempo	NO
Mapas	SI	Contenido multimedia	Fotos
Crear rutas	SI	Creación de multimedia	Fotos
Seguir rutas	SI	Realidad aumentada	NO
Brújula	NO	Otros servicios	NO

Tabla 3. Características de Strava.

Nos permite la creación de rutas propias, basándose en la posición GPS de nuestro dispositivo móvil. Con cada ruta permite asociar una imagen, descripción, tipo de deporte, etc (figuras 13, 14 y 15).

Tiene funcionalidades de red social, ya que nos permite conectarnos con amigos, con los que podemos compartir nuestra actividad y ver la suya. Además, nos permite participar en grupos o clubs para obtener nuevos objetivos y logros.

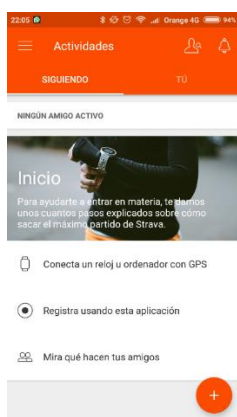


Figura 12. Pantalla inicial de la aplicación.

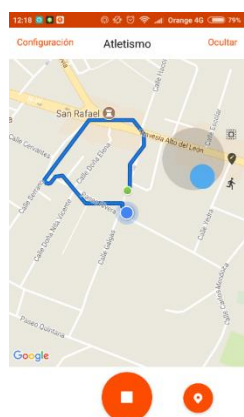


Figura 13. Grabación de un recorrido.

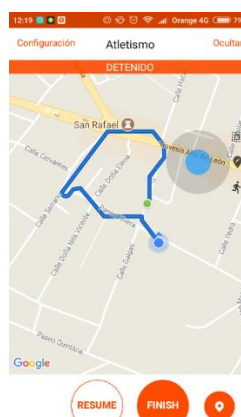


Figura 14. Control de reanudación/finalización de un recorrido.

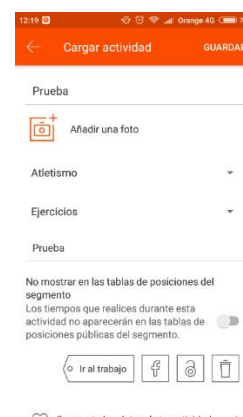


Figura 15. Datos para guardar una ruta finalizada.

Podemos obtener estadísticas globales, o sobre un recorrido al finalizar este, como calorías, tiempo, velocidad, etc (ver figura 18).

Destaca la posibilidad de crear rutas a través de una web, para posteriormente seguirlas desde el dispositivo móvil. También nos permite publicar datos de la actividad física realizada en Google Fit.

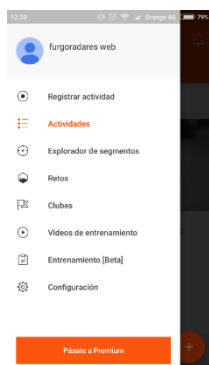


Figura 16. Menú de la aplicación.

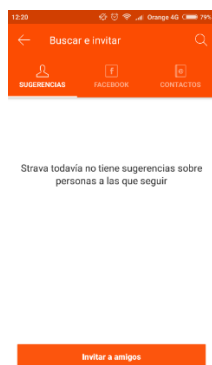


Figura 17. Pantalla de amigos en la aplicación.

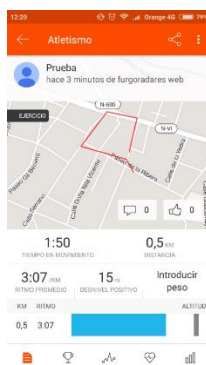


Figura 18. Estadísticas sobre un recorrido.

Endomondo

Se trata de otra aplicación de rutas deportivas. Concretamente está enfocada al ciclismo y la carrera a pie, aunque contempla más variedad de deportes.

Permite la creación de rutas mediante el GPS del dispositivo y obtener un resultado final con datos estadísticos sobre distancia, velocidad, etc.

Información técnica			
Coste	Gratis versión limitada	Opiniones	SI
Ámbito	Global	Red social	NO
SO	Android, IOS,	Puntos de interés	NO
Tamaño	36,50MB	Alojamientos	NO
Idiomas	Español, Inglés, Alemán, Italiano, Polaco, Portugués, Francés, Danés.	Transportes	NO
GPS	SI	Tiempo	NO
Mapas	SI	Contenido multimedia	Fotos
Crear rutas	SI	Creación de multimedia	Fotos
Seguir rutas	SI	Realidad aumentada	NO
Brújula	NO	Otros servicios	NO

Tabla 4. Características de Endomondo.

Tiene una versión gratuita que cuenta principalmente con las siguientes características: seguimiento de entrenamientos en mapa (ver figura 20), análisis de rendimiento (ver figura 22), conexión con otras apps y dispositivo y fijar metas y desafío.

Pueden obtenerse más características con la suscripción Premium (de pago), como estadísticas avanzadas, planes personalizados de entrenamiento y ocultar la publicidad. Puede seguirse un periodo de prueba gratuito de la suscripción Premium de 30 días.

No soporta la creación de contenidos multimedia.

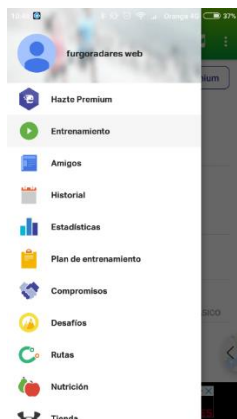


Figura 19. Menú de la aplicación.

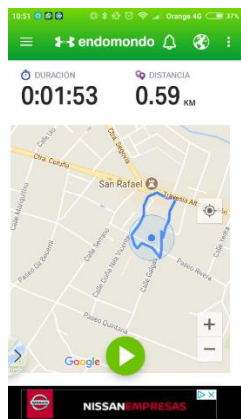


Figura 20. Pantalla de grabación de recorrido.

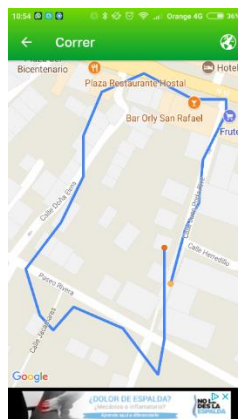


Figura 21. Resultado de un recorrido.



Figura 22. Estadísticas obtenidas en un recorrido.

AsWings

Es una aplicación centrada en rutas de patinaje sobre núcleos urbanos. Permite grabar rutas, registrar tiempos, añadir 'spots' (puntos de interés) notas, imágenes, vídeos o clips de sonido. Permite recorrer rutas propias o las de otros usuarios, avisando si te desvías o de acercas a un spot.

Información técnica			
Coste	Gratis versión limitada	Opiniones	SI
Ámbito	Global	Red social	NO
SO	Android, IOS,	Puntos de interés	NO
Tamaño	36,50MB	Alojamientos	NO
Idiomas	Español, Inglés, Alemán, Italiano, Polaco, Portugués, Francés, Danés.	Transportes	NO
GPS	SI	Tiempo	NO
Mapas	SI	Contenido multimedia	Fotos
Crear rutas	SI	Creación de multimedia	Fotos
Seguir rutas	SI	Realidad aumentada	NO
Brújula	NO	Otros servicios	NO

Tabla 5. Características de AsWings.

Permite establecer relaciones de amistad y compartir las rutas mediante estas relaciones. También permite generar rutas a través de GPX o KML para su seguimiento (ver figura 25). Por el contrario, no permite seguir las rutas generadas por otros usuarios.

Durante el transcurso de las rutas, permite añadir 'sitios' con varios tipos de contenido multimedia: fotografía, vídeo y audio (ver figura 26).

Las características más destacables de esta App son la posibilidad de realizar rutas en grupo, y por otra parte, la notificación de proximidad a eventos de patinaje que pueden registrarse a través de la propia aplicación.



Figura 23. Menú de rutas.



Figura 24. Control del transcurso de ruta.



Figura 25. Seguimiento de rutas.



Figura 26. Creación de contenido multimedia.

Oruxmaps

Se trata de una aplicación que permite generar mapas para su uso offline. Permite la utilización de mapas relieve y 3D, y muestra información de orientación como coordenadas, rumbo, velocidad, altitud, distancia, etc.

Información técnica			
Coste	Gratis	Opiniones	NO
Ámbito	Global	Red social	NO
SO	Android	Puntos de interés	SI
Tamaño	41MB	Alojamientos	NO
Idiomas	Español	Transportes	NO
GPS	SI	Tiempo	NO
Mapas	SI	Contenido multimedia	Fotos
Crear rutas	SI	Creación de multimedia	Fotos
Seguir rutas	SI	Realidad aumentada	NO
Brújula	NO	Otros servicios	NO

Tabla 6. Características de Oruxmaps.

Permite grabar nuestras propias rutas a través de nuestro posicionamiento GPS, así como exportar las rutas generadas en formatos GPX, KML o KMZ (ver figura 27). También permite el seguimiento de estas.

Aunque incluye la característica de los puntos de interés (figuras 28 y 29), estos están enfocados a ser indicaciones para que un caminante siga la ruta, que para destacar sitios o recuerdos del creador de la ruta.

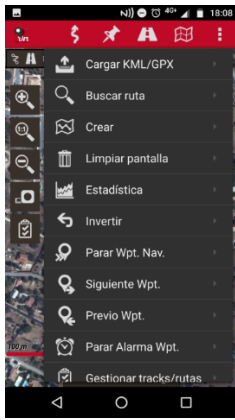


Figura 27. Menú de la aplicación.



Figura 28. Menú de creación de ruta.



Figura 29. Creación de punto de interés.

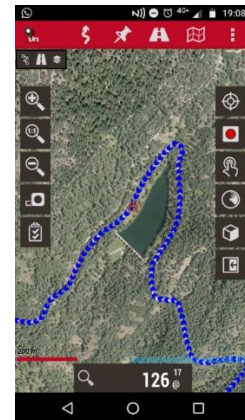


Figura 30. Visualización de ruta.

Google Maps

Se trata de la aplicación oficial de Google para mapas y navegación. Dispone de mapas detallados de 220 países, horarios de transporte público de más de 15.000 ciudades e información detallada sobre establecimientos empresariales de más de 100 millones de lugares.

Información técnica			
Coste	Gratis	Opiniones	SI
Ámbito	Global	Red social	NO
SO	Android, iOS, SmartWach	Puntos de interés	SI
Tamaño	10,13MB	Alojamientos	SI
Idiomas	74 de todo el mundo	Transportes	SI
GPS	SI	Tiempo	NO
Mapas	SI	Contenido multimedia	Fotos
Crear rutas	NO	Creación de multimedia	Fotos
Seguir rutas	SI	Realidad aumentada	NO
Brújula	SI	Otros servicios	SI

Tabla 7. Características de Google Maps.

En Maps podemos explorar los alrededores de cualquier lugar (ver figura 31), ya sea en mapas, vista satélite o visión mixta. Además, podemos iniciar la navegación a cualquier lugar desde cualquier ubicación. Esta navegación puede ser andando, en transporte público o en transporte privado, ofreciendo diferentes alternativas para cada ruta, así como tiempos estimados de duración del recorrido e información sobre el tráfico.



Figura 31. Ubicación del dispositivo en el mapa.

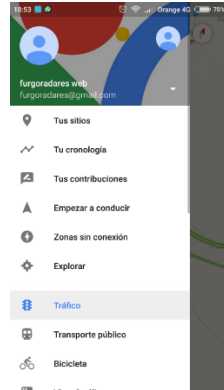


Figura 32. Menú de la aplicación.

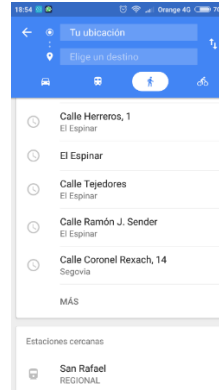


Figura 33. Programación de la navegación.



Figura 34. Exploración de establecimientos cercanos.

También permite importar rutas en formatos como KML, o un modo llamado 'StreetView', que permite una navegación con imágenes 3D a través de las calles de las ciudades y poblaciones.

Es muy importante toda la información sobre establecimientos y puntos de interés que posee Google Maps, permitiendo además sobre estos valoraciones y opiniones de usuario (ver figuras 35 y 36). Así como un montón de información como horarios de apertura o fluencia de gente según la hora.



Figura 35. Puntos de interés y negocios en el mapa.



Figura 36. Puntos de interés y negocios en el mapa.

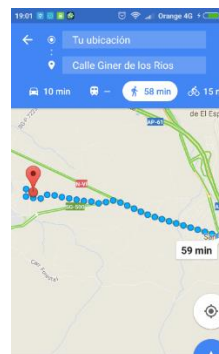


Figura 37. Puntos de interés y negocios en el mapa.

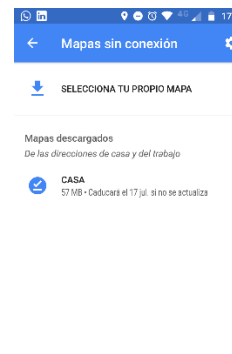


Figura 38. Mapas sin conexión.

Por otra parte, también permite la descarga de mapas para su uso sin conexión, aunque en este modo no permite la navegación.

Permite su integración con multitud de plataformas, gracias a sus APIs (Android, iOS, JavaScript). Esta integración se realiza gracias a librerías proporcionadas por Google y un 'API Key', que es una clave que identifica al desarrollador.

Aunque el uso de Google Maps como usuario es totalmente gratuito, su uso como desarrollador es limitado. A partir de ciertos niveles de tráfico (medido en impresiones), Google establece unas tarifas.

2.1.2 Aplicaciones del Grupo II

En este apartado se describen las aplicaciones del grupo II, donde se engloban las aplicaciones que no tienen características relacionadas con las rutas, pero sí con puntos de interés, geolocalización e información turística.

TourSnap

TourSnapp es una aplicación turística que te avisa de puntos de interés cercanos a tu ubicación, y te proporciona información relevante en el momento. La información proporcionada proviene de la oficina de turismo de la ciudad, por lo que es información contrastada.

La forma de avisar de la aplicación es utilizar un "sonar" que debemos activar, la aplicación trabaja en segundo plano y te avisa de puntos cercanos (a menos de 5 minutos andando) mediante notificaciones (*figura 41*), y te muestra una descripción del punto de interés cuando te encuentres a una distancia aproximada de 1 minuto andando.

En la actualidad, cuenta con 9 ciudades implementadas, todas ellas europeas, siendo 7 de estas españolas.

Información técnica			
Coste	Gratuita	Opiniones	NO
Ámbito	Local	Red social	NO
SO	Android	Puntos de interés	SI
Tamaño	13 MB	Alojamientos	NO
Idiomas	Español, Inglés	Transportes	NO
GPS	SI	Tiempo	NO
Mapas	SI	Contenido multimedia	SI
Crear rutas	NO	Creación de multimedia	NO
Seguir rutas	NO	Realidad aumentada	NO
Brújula	NO	Otros servicios	NO

Tabla 8. Características de TourSnap.

Durante su prueba se ha comprobado que la información del punto de interés, que la aplicación debe de mostrar al estar a corta distancia del punto (*ver figura 40*), no suele funcionar correctamente. Esto puede ser porque no se han controlado los errores que podemos obtener a veces al obtener la posición GPS en Android.



Figura 39. Pantalla de inicio de la aplicación.

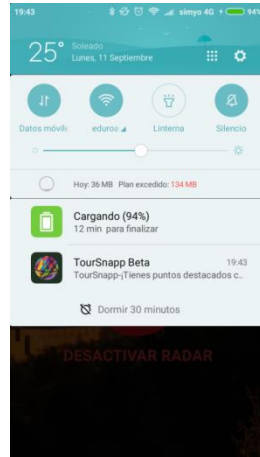


Figura 40. Notificación de punto de interés cercano.



Figura 41. Lista de puntos de interés notificados.



Figura 42. Mapa sobre el que se representa la posición del punto de interés notificado.

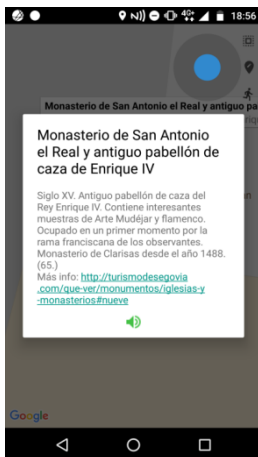


Figura 43. Mensaje al acercarse a un punto de interés.

El hierro

Se trata de una aplicación centrada en el turismo de la propia isla, que sirve para profundizar en el conocimiento de esta y acceder a contenidos y puntos de interés.

Cuenta con un lector de códigos QR (*ver figura 45*), para leer los códigos que están repartidos por la isla para interactuar con la aplicación, y obtener información de las localizaciones (*ver figura 48*) donde se encuentran dichos códigos.

También permite descubrir puntos de la Isla sin estar presente en ella, a través de un mapa (ver figura 47) con todos los puntos de interés recopilados. De todos los puntos se dispone de una galería de imágenes que complementa a la foto principal del punto de interés.

Información técnica			
Coste	Gratuita	Opiniones	NO
Ámbito	Local	Red social	NO
SO	Android, iOS	Puntos de interés	SI
Tamaño	26 MB	Alojamientos	NO
Idiomas	Español, Inglés	Transportes	NO
GPS	SI	Tiempo	NO
Mapas	SI	Contenido multimedia	Fotos
Crear rutas	NO	Creación de multimedia	NO
Seguir rutas	NO	Realidad aumentada	NO
Brújula	NO	Otros servicios	NO

Tabla 9. Características de El Hierro.

Probando la aplicación, se ha comprobado que la funcionalidad del mapa no funciona correctamente desde una ubicación fuera de la Isla. Esta complicación puede salvarse accediendo a una lista de todos los puntos de interés disponibles, pulsando en el botón de menú de la parte superior-derecha de la pantalla de 'explorar' (ver figura 46).



Figura 44. Pantalla de inicio de la aplicación.

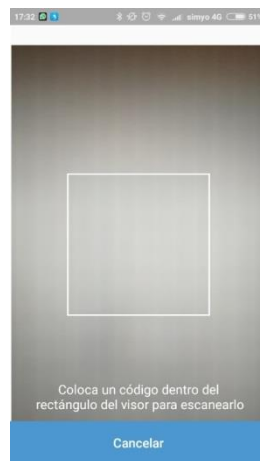


Figura 45. Lector de códigos QR.



Figura 46. Mapa para explorar los puntos de interés.



Figura 47. Imagen e información de puntos de interés.

También es destacable la posibilidad de descargar toda la información e imágenes, para el uso de la aplicación sin conexión a internet, gracias a la última opción del menú de la pantalla principal (ver figura 44).

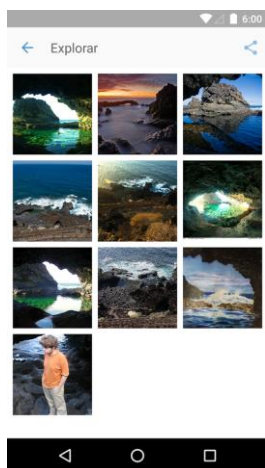


Figura 48. Recopilación de imágenes de un punto.

Metrominuto

Es una aplicación basada en un plano esquemático (simulando una red de transporte público) de todos los lugares turísticos o de interés de la ciudad. Estos están representados por puntos, unidos por líneas rectas que representan la distancia y tiempo que se emplea en recorrer el trayecto entre los dos puntos.

Indica la situación de los principales elementos de movilidad: estaciones, aparcamientos gratuitos y los principales de pago, y los clasifica por distritos junto con los lugares de interés. También nos da información sobre el clima de la ciudad de Pontevedra.

Para calcular el tiempo que se tarda en cubrir las distancias se toma una velocidad de referencia de entre 4 y 5 km/h.

Información técnica			
Coste	Gratuita	Opiniones	NO
Ámbito	Local	Red social	NO
SO	Android, iOS	Puntos de interés	SI
Tamaño	50 MB	Alojamientos	NO
Idiomas	Español, Gallego	Transportes	NO
GPS	NO	Tiempo	SI
Mapas	SI	Contenido multimedia	Fotos
Crear rutas	NO	Creación de multimedia	NO
Seguir rutas	A través de Maps	Realidad aumentada	NO
Brújula	NO	Otros servicios	NO

Tabla 10. Características de Metrominuto.

En su mayor parte es informativa: tiempos andando entre puntos destacados, tiempo en la ciudad, lugares de interés por distritos, lugares añadidos previamente como favoritos y vista satélite con los puntos destacados.

El tema de la navegación entre la posición actual y el lugar que hemos elegido es externo a la aplicación, se realiza por Google Maps externamente (sin integración con su API). Para realizarla tenemos que ampliar la información del punto de interés (ver figura 51), y bajar a la parte inferior pulsando sobre 'Ir allí' (ver figura 52). Esto nos abre la aplicación de Google Maps (ver en aplicaciones del Grupo II) con la ruta trazada entre nuestra posición y el punto de interés.



Figura 49. Plano esquemático de desplazamientos a pie.

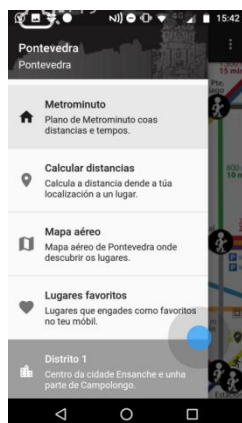


Figura 50. Menú de la aplicación.

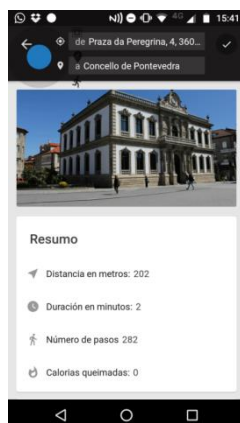


Figura 51. Información principal de un punto de interés.

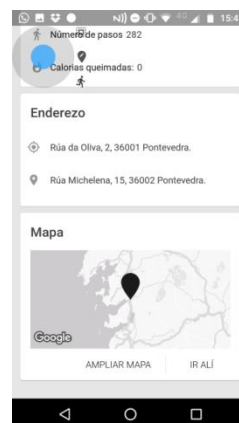


Figura 52. Enlace a Google Maps para navegación.

PAT City Guides

Se trata de una aplicación de información turística, a la cual se van añadiendo ciudades progresivamente, y que contiene audio-guías sobre estas, permitiendo explorar las ciudades como si fueran un museo.

Información técnica			
Coste	Gratuita	Opiniones	SI
Ámbito	Global	Red social	NO
SO	Android, iOS	Puntos de interés	SI
Tamaño	14 MB (sin guías)	Alojamientos	NO
Idiomas	Español, Inglés	Transportes	NO
GPS	SI	Tiempo	NO
Mapas	SI	Contenido multimedia	Fotos, audios
Crear rutas	NO	Creación de multimedia	Fotos
Seguir rutas	NO	Realidad aumentada	NO
Brújula	NO	Otros servicios	NO

Tabla 11. Características de PAT City Guides.

La aplicación tiene un diseño muy limpio e intuitivo. Nos permite acceder con cuenta de usuario o de forma anónima, aunque usarla como usuario registrado e identificado en el sistema te proporciona ciertos privilegios, como por ejemplo descuentos en la compra de entradas a través de la aplicación.

Las guías de las ciudades son descargables (ver figura 54), por lo que pueden visualizarse con o sin Internet. Con la guía de una ciudad descargada, podemos visualizar todos los puntos de interés que posee dicha ciudad, tanto en formato de lista (ver figura 57) como representados en un mapa (ver figura 58). Pulsando sobre un ítem de la lista o del mapa entramos en la información del punto de interés, dónde podemos encontrar foto, reproducir la audio guía sobre el punto de interés y acceder a un menú de opciones (ver figura 60). En este menú podemos encontrar la información en texto, añadir imágenes, compartir por redes sociales o marcar como favorito el punto.

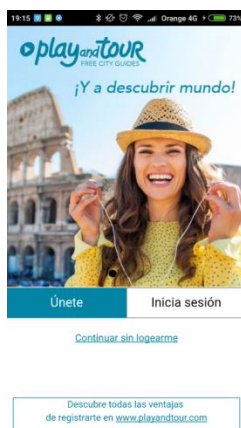


Figura 53. Pantalla inicial de la aplicación.



Figura 54. Lista de guías disponibles para descargar.

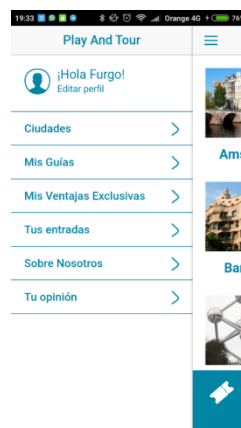


Figura 55. Menú de la aplicación.

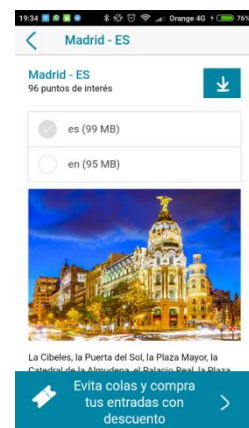


Figura 56. Información de descarga de la guía.

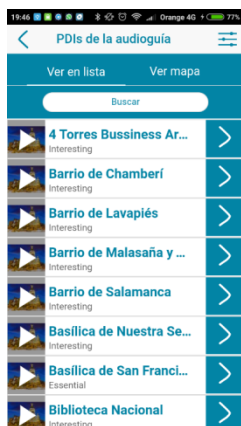


Figura 57. Lista de puntos de interés de la guía de una ciudad.

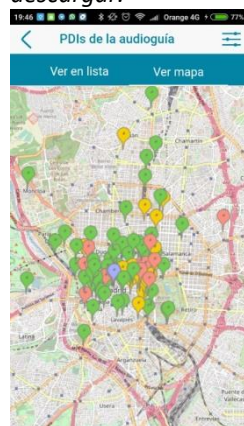


Figura 58. Puntos de interés representados en un mapa.

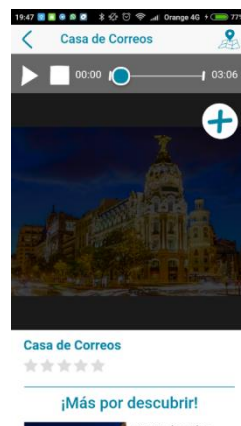


Figura 59. Información sobre un punto de interés. Texto, imagen y audio.

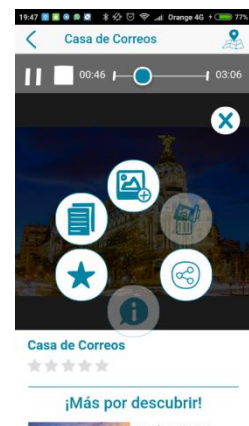


Figura 60. Opciones sobre un punto de interés.

Tripwolf

Es una aplicación de guías turísticas (dispone de más de 600 ciudades de todo el mundo). Los contenidos de sus guías. Gran parte del contenido es información proporcionada por la guía de viajes “Marco Polo”, una de las más populares en ventas.

Dispone de una versión gratuita con funcionalidades reducidas y una versión de pago con todas las características. Independientemente de la versión todas las guías tienen algún coste. En su versión completa, cuenta con mapas offline, navegación, y un planificador de puntos a visitar en el viaje.

Información técnica			
Coste	Según versión	Opiniones	SI
Ámbito	Global	Red social	NO
SO	Android, iOS	Puntos de interés	SI
Tamaño	98 MB (sin guías)	Alojamientos	SI
Idiomas	Español, Inglés, Alemán, Francés e Italiano	Transportes	SI
GPS	SI	Tiempo	NO
Mapas	SI	Contenido multimedia	Fotos
Crear rutas	NO	Creación de multimedia	NO
Seguir rutas	NO	Realidad aumentada	NO
Brújula	NO	Otros servicios	Planificador viaje

Tabla 12. Características de TripWolf.

Cuando abrimos la aplicación nos encontramos tres pestañas principales (*ver figura 61*), una de bienvenida con una selección de puntos de interés de las distintas guías, una con todas las guías destacadas que contiene la aplicación, y una de ofertas (para comprar varias guías a menor precio que por separado).

Cuando nos encontramos consultando la guía de una ciudad (*figuras 65 y 66*) encontramos una detallada descripción de la ciudad, y debajo una serie de recuadros que organizan las opciones y puntos de interés disponibles. Destacando el planificador de viaje (*ver imagen x*), los lugares destacados, atracciones, actividades, establecimientos hosteleros, transporte, etc.

Cuando seleccionamos un punto de interés (*ver figura 67*), además de leer información sobre él, podemos obtener información sobre recorridos que lo contengan, añadirlo al plan de viaje, obtener navegación para llegar y añadir o leer reseñas sobre él. Si el punto requiere algún tipo de reserva o entrada también se puede adquirir.

Como aplicación de apoyo para recorrer una ciudad es muy útil y está muy bien valorada, pero no nos permite guardar ni compartir experiencias sobre nuestro viaje. Además, el contenido gratuito es muy limitado.

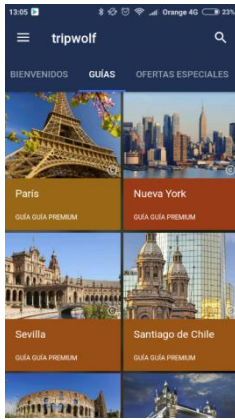


Figura 61. Listado de guías destacadas disponibles.

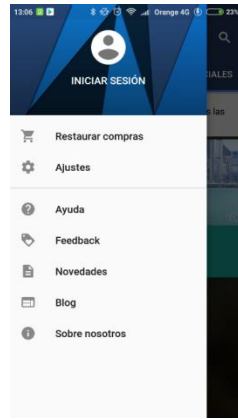


Figura 62. Menú de opciones de la aplicación.

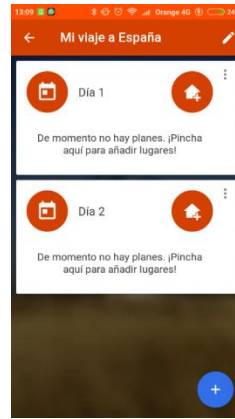


Figura 63. Planificador para la visita a la ciudad, organizado en días de vista.

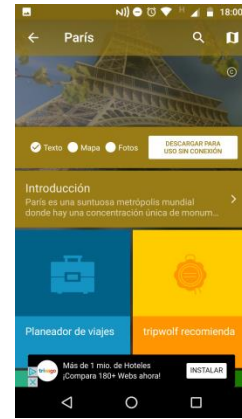


Figura 64. Ejemplo de guía.



Figura 65. Opciones de la guía sobre actividades y movilidad.



Figura 66. Clasificación dentro de la opción de visitas y actividades.



Figura 67. Ficha de un lugar de interés.

Journi

Se trata de una app que funciona como una red social de viajes, en la que puedes publicar fotos, texto, mapas... de cada uno de tus viajes, con la posibilidad de compartir este contenido en tu perfil y recibir feedback de amigos.

Permite crear y compartir contenido multimedia como cualquier otra red social, con la diferencia de que nos permite agrupar la información por viajes y está enfocada en estos, sin embargo, no permite la creación de rutas ni su seguimiento, ni nos ofrece información o interacción con los lugares en los que nos encontramos.

Es destacable que nos permite trabajar off-line para poder crear contenido sin conexión a Internet (se sube posteriormente).

En la aplicación tenemos agrupados nuestros viajes. Estos pueden ser vistos por nuestros amigos en la red social según los tengamos configurados. Cada vez que iniciamos un viaje lo

creamos (ver figuras 69 y 70), y después podemos irle añadiendo todo tipo de contenido (ver figura 71).

Información técnica			
Coste	Gratuita limitada	Opiniones	SI
Ámbito	Global	Red social	SI
SO	Android, iOS	Puntos de interés	SI
Tamaño	80MB	Alojamientos	SI
Idiomas	Español, Inglés	Transportes	SI
GPS	SI	Tiempo	SI
Mapas	SI	Contenido multimedia	Fotos, audio, vídeo
Crear rutas	NO	Creación de multimedia	Fotos, audio, vídeo
Seguir rutas	NO	Realidad aumentada	NO
Brújula	NO	Otros servicios	NO

Tabla 13. Características de Journi.

Aunque nos permite organizar de una forma vistosa todos los recuerdos de nuestros viajes, no está enfocada al seguimiento de estos. También es un modelo totalmente diferente el concepto de tener una red social solo para viajes, frente a la idea de una herramienta centrada en características específicas sobre creación y seguimiento de viajes, que posteriormente pueda exportar contenido a redes sociales. Por otra parte, la aplicación si contempla compartir contenido con otras redes sociales.



Figura 68. Pantalla inicial, con nuestros viajes y la opción de crear nuevos.



Figura 69. Los viajes se identifican por el lugar de partida.

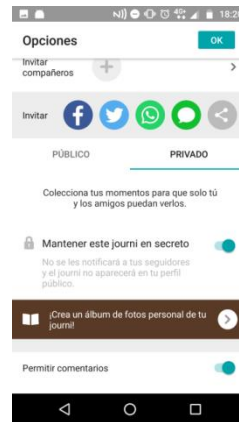


Figura 70. Podemos marcar un viaje como público o privado, invitar compañeros, compartirlo en redes sociales, permitir comentarios, etc.

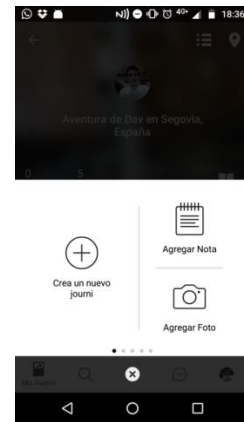


Figura 71. En los viajes podemos añadir notas, fotos, datos de clima, ubicaciones, etc.

CityMaps2Go

Es una aplicación de mapas y puntos de interés para visitar ciudades. Combina contenido informativo de Wikipedia con imágenes y lugares de interés de las ciudades. Además de contar

con información de guías de Wikipedia (<https://wikitravel.org>), también cuenta con información adicional y puntos proporcionados por “OpenStreetMap” y “Foursquare”. También permite la compra de entradas, y sugiere alojamientos cercanos donde alojarse.

Permite la descarga de los mapas para su visualización sin conexión a internet.

Información técnica			
Coste	Según versión	Opiniones	SI
Ámbito	Global	Red social	SI
SO	Android, iOS	Puntos de interés	SI
Tamaño	115MB	Alojamientos	SI
Idiomas	Español, Inglés	Transportes	NO
GPS	SI	Tiempo	SI
Mapas	SI	Contenido multimedia	Fotos
Crear rutas	NO	Creación de multimedia	NO
Seguir rutas	NO	Realidad aumentada	NO
Brújula	NO	Otros servicios	NO

Tabla 14. Características de CityMaps2Go.

La aplicación nos permite descubrir destinos (*primera opción en figura 72*), o bien buscar por palabra (*tercera opción en figura 72*), o bien por país (*ver figura 73*). Descargado el mapa disponemos de la guía (*ver figura 74*), y el mapa (*ver figura 75*) con multitud de puntos de interés, que podemos seleccionar.

Al seleccionar un punto, vemos su información (*ver figuras 76 y 77*) y fotos. Así como ver y dejar comentarios sobre el lugar.

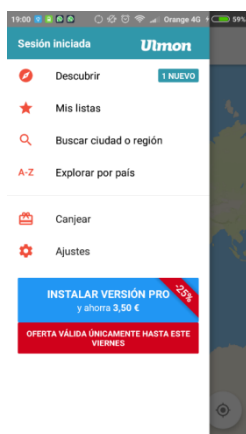


Figura 72. Menú de la aplicación.

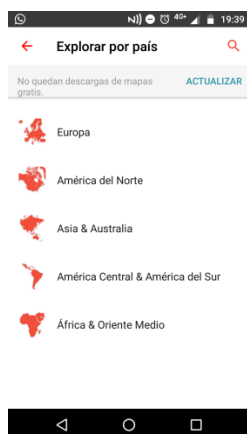


Figura 73. Búsqueda de mapas para descargar.

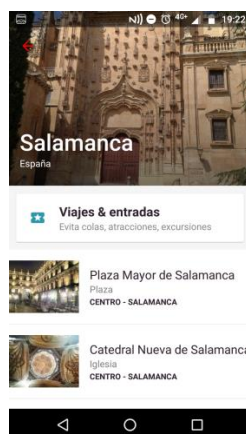


Figura 74. Vista principal de una ciudad.

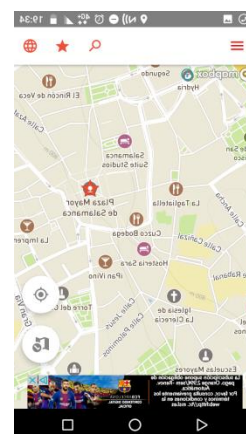


Figura 75. Mapa descargado de una ciudad.

En el mapa podemos ver la posición de los puntos, y situar nuestra posición, pero no podemos realizar una navegación o seguimiento hasta los puntos de interés.

También en la versión PRO (la cual no es gratuita) podemos compartir una lista de los puntos que hemos visitado.



Figura 76. Visualización de un punto de interés.



Figura 77. Fotos y comentarios de un punto de interés.

Minube

Es una aplicación con información sobre todo lo que podemos encontrar en más de dos millones de lugares en todo el mundo. Nos sugiere destinos según nuestras preferencias y muestra lugares para visitar, lugares donde comer, y establecimientos para alojarse.

Información técnica			
Coste	GRATIS	Opiniones	SI
Ámbito	Global	Red social	SI
SO	Android, IOS, Smartwatch	Puntos de interés	SI
Tamaño	58,4MB	Alojamientos	SI
Idiomas	Español, Inglés, Alemán, Francés, Italiano, polaco y Portugués.	Transportes	NO
GPS	SI	Tiempo	NO
Mapas	SI	Contenido multimedia	Fotos
Crear rutas	NO	Creación de multimedia	Fotos
Seguir rutas	NO	Realidad aumentada	NO
Brújula	NO	Otros servicios	Retos

Tabla 15. Características de Minube.

La aplicación nada más abrirse nos permite buscar destinos, o encontrar sugerencias según el tipo de escapada o viaje que queremos realizar. Cuando seleccionamos un destino (ver figura 80) podemos ver todos los lugares destacados que nos ofrece, mostrándonos fotos y comentarios del lugar cuando lo seleccionamos para acceder a él. También nos enlaza a la posibilidad de realizar reservas (mediante otra plataforma, Booking) y consultar vuelos para llegar. Podemos filtrar los lugares de interés en sitios para comer o dormir (ver figura 81).

Mi nube además nos plantea retos para valorar los puntos de interés de los lugares en los que nos encontramos.



Figura 78. Pantalla inicial que nos permite buscar destino.

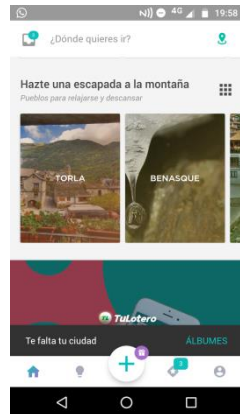


Figura 79. Debajo de la pantalla principal tenemos sugerencias de destinos.

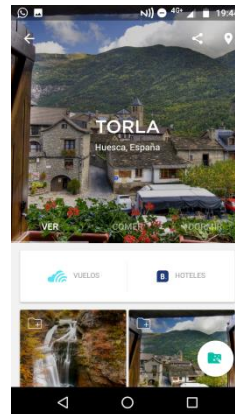


Figura 80. Destino y sus lugares de interés.

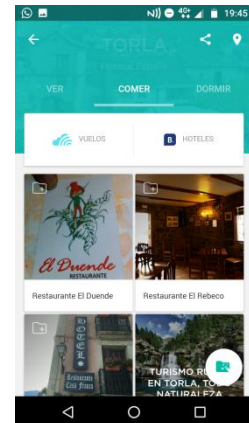


Figura 81. Establecimientos hoteleros.

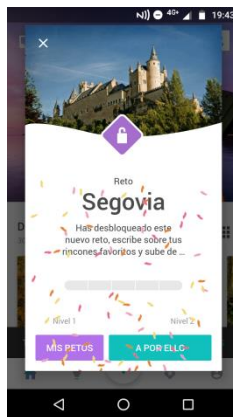


Figura 82. Desbloqueo de retos.

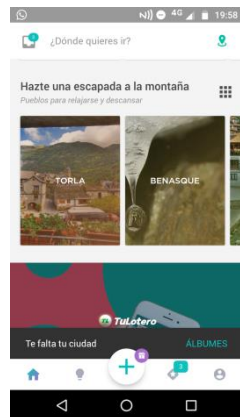


Figura 83. Sugerencias de destinos.

2.1.3 Aplicaciones del Grupo III

En este apartado se describen las aplicaciones del grupo III, donde se engloban las aplicaciones con características de realidad aumentada, o realidad virtual.

Guía Madrid 5D

Es una aplicación desarrollada para proponer una experiencia dinámica en el turismo de la Comunidad Autónoma de Madrid. Posee contenidos sobre patrimonio y servicios de la comunidad, así como gran cantidad de contenido multimedia.

Puede usarse offline si se realiza una descarga de contenidos, así como una realización de los recorridos virtualmente gracias a una característica de GPS simulado.

Información técnica			
Coste	Gratis	Opiniones	NO
Ámbito	Regional	Red social	NO
SO	Android, iOS	Puntos de interés	SI
Tamaño	13,93 MB (sin guías)	Alojamientos	SI
Idiomas	Inglés y Español	Transportes	NO
GPS	SI	Tiempo	NO
Mapas	SI	Contenido multimedia	Fotos, Vídeo
Crear rutas	NO	Creación de multimedia	NO
Seguir rutas	SI	Realidad aumentada	SI
Brújula	SI	Otros servicios	SI

Tabla 16. Características de Guía Madrid 5D.

La aplicación organiza todas sus guías virtuales en 5 secciones, y una adicional con la red de oficinas turísticas de la comunidad. Estas 5 secciones en las que se organiza la aplicación son: 'Patrimonio Mundial en Madrid', 'Villas de Madrid', 'Rutas temáticas', 'Sierra Oeste' y 'Municipios' (ver figura 84).



Figura 84. Categorías de la aplicación.



Figura 85. Listado de guías de los municipios de Madrid.



Figura 86. Pantalla de información y descarga de guía.



Figura 87. Mapa con puntos de interés de una guía.

Dentro de estas categorías encontramos multitud de guías virtuales que puede descargarse para su seguimiento de forma online, o para ser usada como herramienta de realidad aumentada, de forma que según nos acerquemos a los puntos de interés, obtenemos información sobre ellos (figuras 85, 86 y 87).

También dispone de una amplia información sobre establecimientos gastronómicos y hoteleros (ver figura 89).



Figura 88. Opciones sobre GPS y simulación.



Figura 89. Categorías de información.

Camino de Santiago 360º

Esta aplicación nos permite recorrer el Camino de Santiago desde cualquier lugar gracias a la realidad virtual. Junta un recorrido con personajes ficticios, con información, para crear una guía interactiva que acerque a la experiencia de recorrer el camino. Permite descubrir paisajes, y entrar en monumentos para descubrir cómo son por dentro (*figuras 91 y 93*).

Información técnica			
Coste	0€ instalación+1 capítulo 2,99€ disponer de los 9 capítulos 0,99€ adquirir un capítulo suelto (Android)	Opiniones	NO
Ámbito	Camino de Santiago	Red social	NO
SO	Android, iOS	Puntos de interés	SI
Tamaño	580MB	Alojamientos	SI
Idiomas	Inglés y Español	Transportes	NO
GPS	NO	Tiempo	NO
Mapas	SI	Contenido multimedia	Fotos, Vídeo
Crear rutas	NO	Creación de multimedia	NO
Seguir rutas	NO	Realidad Virtual	SI
Brújula	NO	Otros servicios	NO

Tabla 17. Características de Santiago 360º.

Una característica muy destacable de esta aplicación es que puede ser usada con gafas de realidad virtual, o mediante interacción táctil en caso de no disponer de estas (*ver figura 92*).

Aunque plantea el seguimiento virtual de una ruta, a través de etapas y puntos de interés (figuras 90 y 95), no se trata de una herramienta que nos permita seguir como recorreremos nosotros mismo una ruta.



Figura 90. Elección de etapas del camino.



Figura 91. Comienzo de la primera etapa con personajes virtuales.

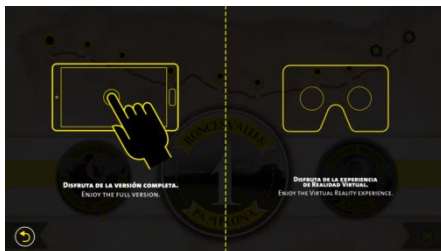


Figura 92. Elección entre táctil o gafas de realidad virtual para usar la app.



Figura 93. Visita a un monumento e información sobre este.



Figura 94. Momento de una etapa donde se puede elegir puntos de interés a visitar.



Figura 95. Mapa de una etapa con sus localidades y puntos de interés.

Senditur

Se trata de una aplicación inicialmente enfocada hacia el camino de Santiago, pero que posteriormente ha aumentado su alcance a todo el territorio español, para la exploración de sitios en el recorrido turístico.

La aplicación nos permite explorar puntos de interés de nuestro entorno, gracias al uso de Realidad Aumentada con la cámara del dispositivo móvil. También nos ofrece los puntos de interés cercanos en lista (ver figura 96) o representados en un mapa (ver figura 97), pero su atractivo está en su uso con la cámara. Según enfocamos el horizonte con la cámara y nos movemos, o giramos, mediante iconos nos muestra en qué dirección se encuentran los puntos de interés próximos.

Es importante reseñar que, para poder usar esta aplicación con realidad aumentada, es imprescindible tener un dispositivo móvil con brújula.

Información técnica			
Coste	Gratis	Opiniones	NO
Ámbito	Nacional	Red social	NO
SO	Android, iOS	Puntos de interés	SI
Tamaño	48 MB	Alojamientos	SI
Idiomas	Español	Transportes	NO
GPS	SI	Tiempo	NO
Mapas	SI	Contenido multimedia	Fotos
Crear rutas	NO	Creación de multimedia	NO
Seguir rutas	NO	Realidad aumentada	SI
Brújula	NO	Otros servicios	NO

Tabla 18. Características de Senditur.

En cualquiera de los modos, podemos acceder a la información de cada punto de interés. Dentro de esta información, podemos solicitar indicaciones para llegar al punto desde nuestra posición, pero para ofrecérselas la app usará de forma externa la aplicación 'Google Maps' (anteriormente mencionada en el punto 3.1.2), sin utilizar integración con el API proporcionado por Google.

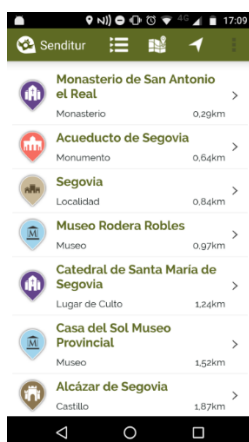


Figura 96. Lista de puntos de interés cercanos.

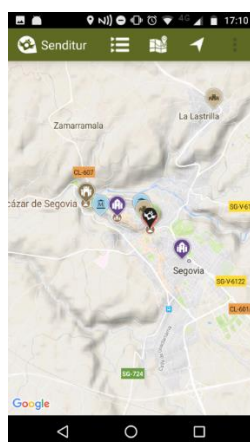


Figura 97. Puntos de interés dibujados en un mapa.



Figura 98. Exploración del entorno con la cámara.



Figura 99. Información de un punto de interés.

2.2 Conclusiones

En este apartado se pretende reflexionar sobre lo aprendido a través del análisis de las aplicaciones existentes en el mercado, para buscar un factor diferenciador y completar las ideas iniciales sobre el proyecto.

Hemos podido comprobar que existe gran variedad de aplicaciones sobre información turística y puntos de interés y, por otro lado, multitud de aplicaciones que permiten crear (y en menor medida) seguir rutas. Sin embargo, no son tantas las que, combinando la aparición de puntos

de interés y creación de rutas, dan una alta importancia a los puntos de interés, y menos aún a los puntos donde el usuario decide recordar como interesantes para él, tomando una foto. Podemos encontrar multitud de aplicaciones enfocadas al deporte, que nos ofrecen estadísticas, pero no una atención sobre el recuerdo de un recorrido.

Este proyecto busca la posibilidad de crear rutas, y que otros usuarios puedan reinventarlas a la vez que están siguiendo la ruta de otro usuario. Es un concepto que hemos podido ver en 'Wikiloc' pero, aunque esta aplicación nos permite establecer puntos personalizados, están poco destacados, y parece tener un enfoque de indicaciones para seguir un camino. El presente proyecto busca dar más importancia a las imágenes obtenidas con el dispositivo móvil mientras realizamos un recorrido, de forma que la visualización de nuestro itinerario represente el camino que seguimos hasta los puntos que nos impresionaron, y que decidimos recordar en una fotografía. Estos puntos pueden ser más importantes para el usuario que los puntos de interés predefinidos por una aplicación, que pueden coincidir o no con los gustos de este. El espíritu del proyecto es descubrir, guardar para el recuerdo y compartir conocimiento de nuestra experiencia personal al recorrer un lugar, no hacer kilómetros o ser una tradicional guía de viaje.

Si bien es cierto que el proyecto pretende centrarse en que el usuario recorra lugares para descubrirlos, también pretende ofrecerle al usuario sugerencias sobre puntos de interés. Son interesantes los enfoques de las aplicaciones TourSanp, con notificaciones por proximidad, o de Senditur con la exploración realidad aumentada. Ambas aplicaciones implementan dos soluciones distintas, pero lo que tiene en común es que no contemplan ninguna característica relacionada con rutas o recorridos, por lo que no sirven para cumplir nuestra idea de negocio.

Capítulo 3

Planificación

En este capítulo se realiza una descripción detallada de los pasos previstos a seguir para el desarrollo del proyecto, así como estimaciones temporales y económicas que se esperan en el desarrollo.

3.1 Metodología de desarrollo

El primer paso para iniciar el desarrollo de software es establecer el modelo que se va a seguir en este proceso, buscando la mejor herramienta para desarrollador y cliente, que permita un producto final satisfactorio.

En el caso de este proyecto, que se trata de un trabajo de autoaprendizaje, se va a utilizar una metodología 'Incremental'. Esta metodología divide el desarrollo del proyecto en varios fragmentos que se denominan 'incrementos', que son iteraciones que van añadiendo funcionalidad al desarrollo.

Mediante estos incrementos funcionales, se pretende que el obtener un continuo aprendizaje, gracias al desarrollo en de cada iteración, y el uso de la funcionalidad de estas cuando están terminadas.

En cada iteración se busca complementar la funcionalidad del producto, obteniendo en cada entrega un producto completamente funcional, hasta llegar a la solución completa que se pretendía obtener. En la primera iteración se desarrolla una arquitectura completa del sistema, seguido de sucesivos incrementos, que se realizaran en un orden determinado, buscando realizar primero aquellos cuya funcionalidad mayor valor y utilidad aporten al conjunto del proyecto.

Cada incremento, comprenderá una serie de etapas, que son las siguientes:

1. **Análisis:** Comprende inicialmente la especificación de las características principales. En los sucesivos incrementos, se contemplará si es necesario cambiar o añadir requisitos, en base a lo aprendido en el anterior incremento.
2. **Diseño:** Una vez analizadas las funcionalidades se procede al diseño de los nuevos componentes del sistema descritos en la etapa anterior.
3. **Implementación:** Comprende las labores de programación necesarias para desarrollar el prototipo funcional que ha de obtenerse en el incremento.

4. **Pruebas:** Finalizado el desarrollo, es necesario proceder a comprobar el correcto funcionamiento del incremento.
5. **Documentación:** Para finalizar, debe de documentarse todo lo añadido, modificado o probado durante el incremento.

Estas serán repetidas en cada uno hasta obtener el producto final.

Para establecer las iteraciones que se van a realizar en este proyecto, vamos a tener en cuenta qué características aportan más valor

Para la primera iteración a seguir, hay que tener en cuenta que este proyecto esta complementado por otra plataforma, y que esta debe dar un soporte de servicios a la parte de dispositivos móviles. Podemos considerar que estos servicios son la parte que más valor aportará al resultado final del proyecto, ya que estos servicios que controlarán el flujo de la información, y lo harán tanto para la interfaz de esta plataforma, como para el dispositivo móvil. Junto con estos, la base del proyecto esta complementada por la Seguridad y la Persistencia.

Una vez conseguida la funcionalidad de los servicios, se deben de desarrollar las tres ramas de características vistas en el apartado 1.3 de este documento. De estas tres ramas, podemos considerar la más importante la de 'Gestión de Itinerarios', ya que es en la que está centrada el proyecto, después la de 'Gestión Social', ya que se trata de la que va a permitir que podamos compartir la información entre los distintos usuarios, así como regular a estos, y por último, está la 'Gestión de Puntos de Interés', que pretende complementar la experiencia de los usuarios.

Este planteamiento proporciona un escenario de desarrollo en 4 incrementos, como puede verse en la *Figura 100*.

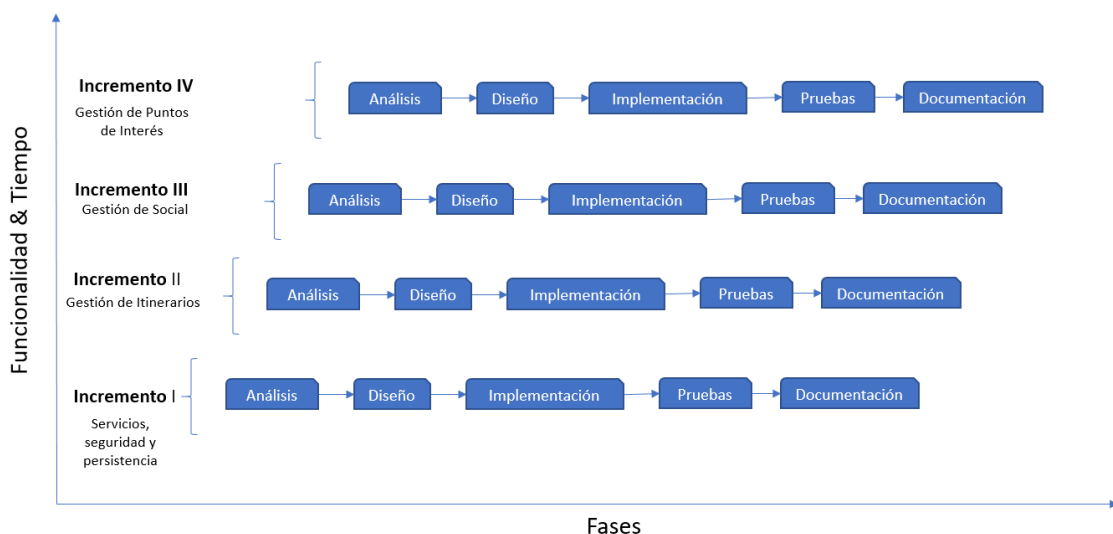


Figura 100. Diagrama de incrementos y fases.

3.2 Detalle de planificación

En este apartado vamos a detallar la planificación realizada para el proyecto, en un Diagrama de Gantt. En él, podremos observar el tiempo estimado para cada iteración, así como las distintas fases de cada iteración.

Definimos la fecha de inicio del proyecto el 08/11/17, y como fecha de fin el 11/06/18. El detalle de comienzos, fin y duración de días, de cada iteración y fase puede observarse en la siguiente figura (*Figura 101*).

Etapas proyecto	Días		Inicio	Final
ITERACIÓN I	42.0	84.0	13-11-17	09-01-18
Análisis	20.0		13-11-17	08-12-17
Diseño	11.0		11-12-17	25-12-17
Implementación	7.0		26-12-17	03-01-18
Pruebas	2.0		04-01-18	05-01-18
Documentación	2.0		08-01-18	09-01-18
ITERACIÓN II	39.0	78.0	10-01-18	05-03-18
Análisis	4.0		10-01-18	15-01-18
Diseño	4.0		16-01-18	19-01-18
Implementación	23.0		22-01-18	21-02-18
Pruebas	4.0		22-02-18	27-02-18
Documentación	4.0		28-02-18	05-03-18
ITERACIÓN III	36.0	72.0	06-03-18	24-04-18
Análisis	5.0		06-03-18	12-03-18
Diseño	3.0		13-03-18	15-03-18
Implementación	18.0		16-03-18	10-04-18
Pruebas	7.0		11-04-18	19-04-18
Documentación	3.0		20-04-18	24-04-18
ITERACIÓN IV	34.0	68.0	25-04-18	11-06-18
Análisis	3.0		25-04-18	27-04-18
Diseño	3.0		30-04-18	02-05-18
Implementación	14.0		03-05-18	22-05-18
Pruebas	8.0		23-05-18	01-06-18
Documentación	6.0		04-06-18	11-06-18

Figura 101. Detalle de planificación de etapas.

En las siguientes figuras (*102, 103, 104 y 105*) se muestra el Diagrama de Gantt para cada iteración del proyecto.

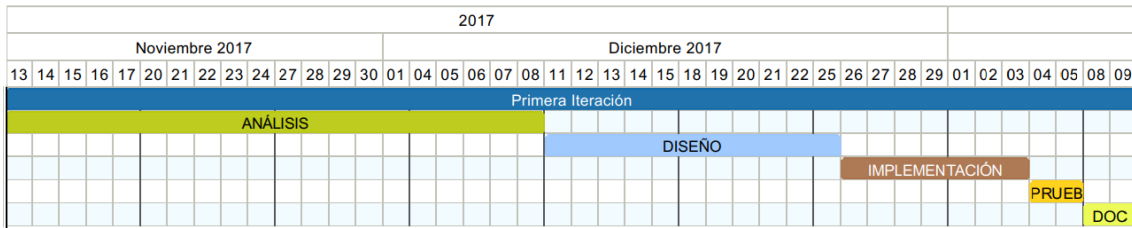


Figura 102. Gantt de planificación de la primera iteración.

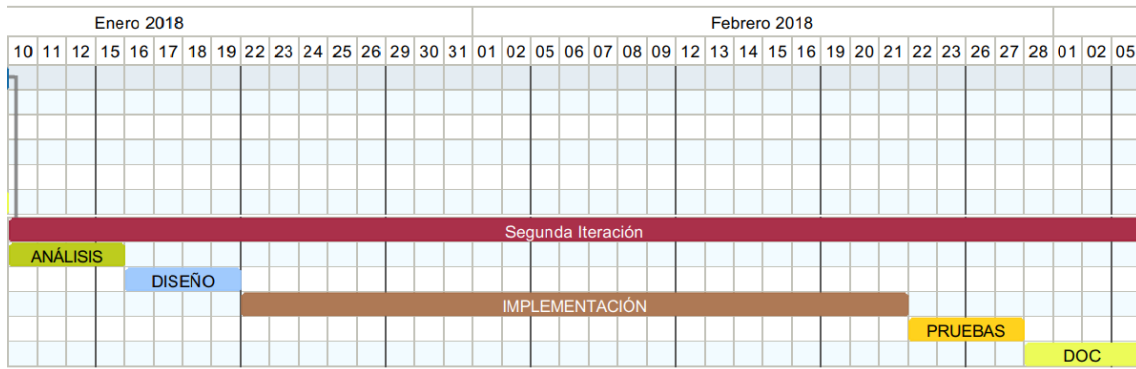


Figura 103. Gantt de planificación de la segunda iteración.

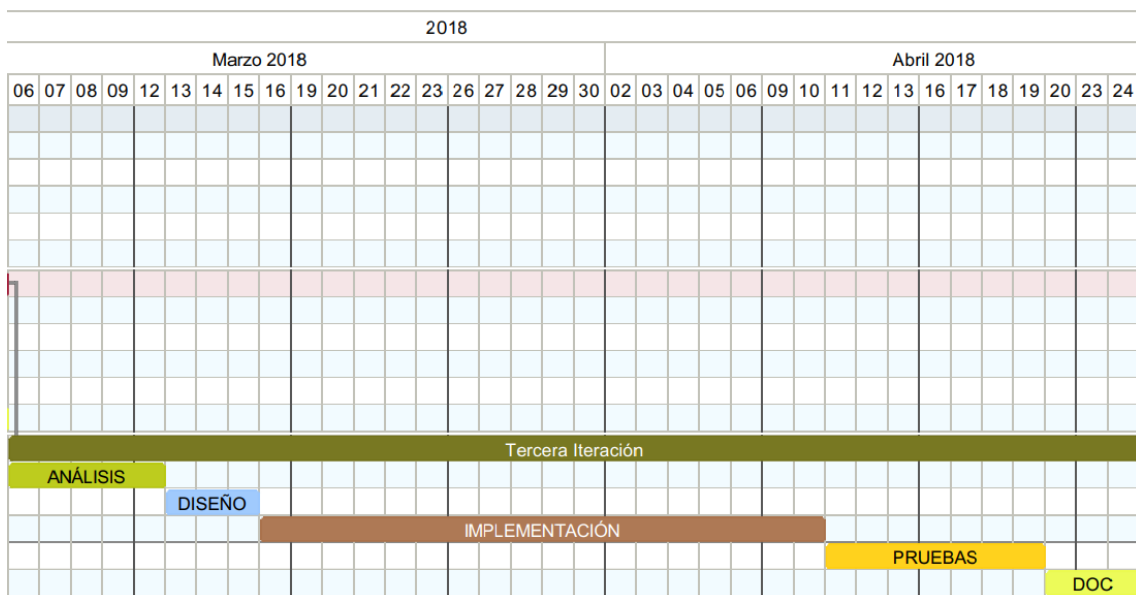


Figura 104. Gantt de planificación de la tercera iteración.

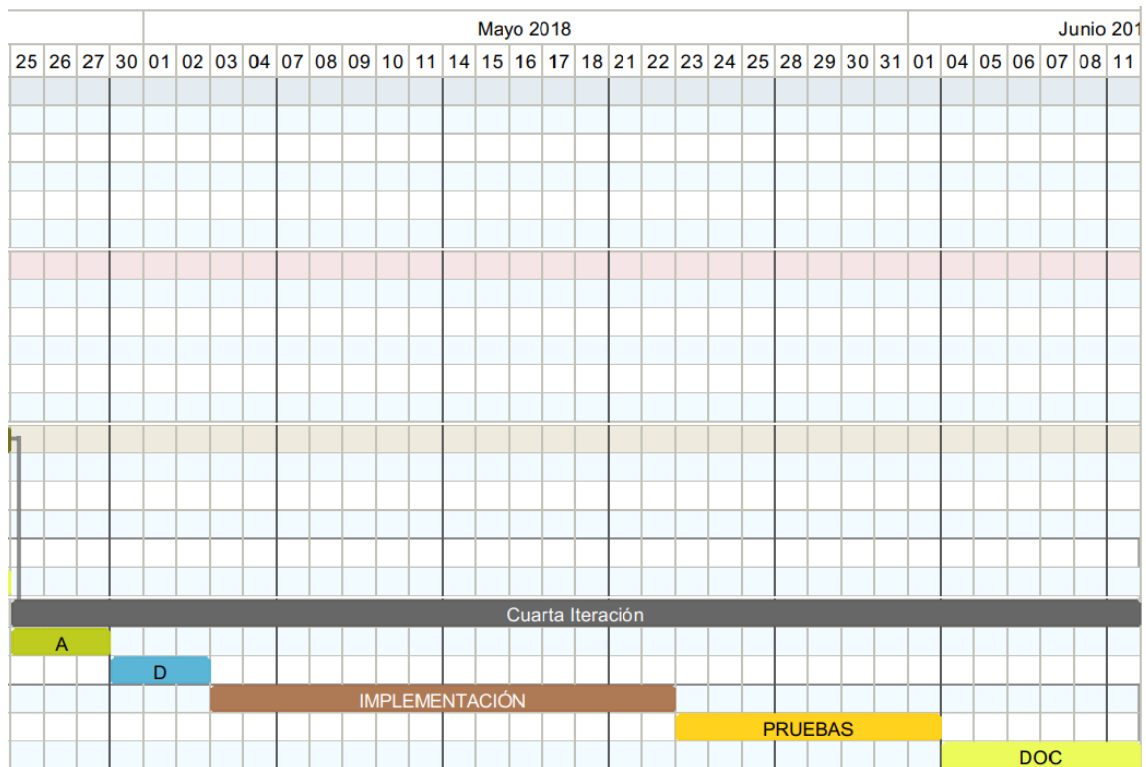


Figura 105. Gantt de planificación de la cuarta iteración.

3.3 Estimaciones temporales

Existen multitud de técnicas para la estimación temporal en desarrollo de Software. En este apartado, se realizarán dos estimaciones distintas para hacernos una idea del tiempo de desarrollo que llevará el proyecto, y así poder hacer un presupuesto.

Las estimaciones que a continuación se van a llevar a cabo son: Estimación por Puntos de Función y Estimación por COCOMO.

3.3.1 Estimación por Puntos de Función

La estimación por puntos de función es un método utilizado en el desarrollo de software, para medir la funcionalidad de un determinado software, independientemente de la tecnología o lenguaje usado para construirlo. Fue definida por Allan Albrecht en 1979.

Para el uso de este método de estimación se siguen los siguientes pasos:

- Definición de flujos de información y sus complejidades:
 - Entradas de usuario: datos que comunica el usuario al sistema.
 - Salidas de usuario: datos que el sistema procesa para mostrar al usuario.
 - Consultas: consultas que realiza el sistema como respuesta inmediata.

- Ficheros lógicos internos: donde el sistema persiste su información.
 - Ficheros externos: ficheros o fuentes de información con las que trabaja el sistema.
2. Recuento de elementos de cada clase y asignación de complejidad a estos (baja, media o alta).
 3. Se obtiene un valor de puntos de función no ajustados (PFNA), a través del cálculo ponderado de pesos (*ver en tabla 19*).
 4. Se ajustan los PFNA, aplicando un factor producto de la suma de 14 factores de complejidad (*ver tabla tal*). El peso que se le otorga a estos factores es un valor entre 0 y 5. Teniendo en cuenta que este valor representará la influencia o importancia del factor en el proceso de desarrollo, siguiendo esta equivalencia: 0 – Sin influencia, 1- Influencia mínima, 3 – Influencia moderada, 4 – Influencia significativa, 5 – Influencia fuerte.
 5. Finalmente, para obtener una estimación de líneas de código se utiliza una equivalencia entre puntos de función y lenguaje de programación. En el presente proyecto se van a combinar distintos lenguajes, por lo que aplicaremos varios valores según la equivalencia obtenida en la web '<http://www.qsm.com/resources/function-point-languages-table>'. Estos han sido reflejados en la *Tabla 21*.

De acuerdo con el punto 1, definimos lo siguientes flujos en el sistema:

ENTRADAS DE USUARIO

- Registro → *Complejidad baja*
- Buscar usuario → *Complejidad baja*
- Solicitar amistad → *Complejidad media*
- Descartar amistad → *Complejidad baja*
- Aceptar petición → *Complejidad baja*
- Descartar amistad → *Complejidad baja*
- Rechazar petición de amistad → *Complejidad baja*
- Modificar itinerario → *Complejidad alta*
- Añadir punto de interés → *Complejidad media*
- Modificar punto de interés → *Complejidad media*
- Eliminar punto de interés → *Complejidad baja*
- Eliminar itinerario → *Complejidad baja*
- Buscar itinerario → *Complejidad baja*

SALIDAS DE USUARIO

- Ver amistades → *Complejidad media*
- Ver itinerarios → *Complejidad baja*
- Ver peticiones → *Complejidad media*
- Ver puntos → *Complejidad baja*
- Ver itinerario → *Complejidad baja*

CONSULTAS

- Login → *Complejidad baja*
- Comprobar existencia de usuario → *Complejidad baja*

FICHEROS LÓGICOS INTERNOS

- Base de Datos → *Complejidad media*

FICHEROS LÓGICOS EXTERNOS

- Plataforma Software móvil → *Complejidad alta*

Una vez identificados estos flujos y otorgada su complejidad, obtenemos los Puntos de Función No Ajustados en la *Tabla 19*.

Puntos de función sin ajustar			
Elemento	Complejidad	Número	Total
Entradas	Baja (x3)	8	24
	Media (x4)	3	12
	Alta (x6)	1	6
Salidas	Baja (x4)	2	8
	Media (x5)	2	10
	Alta (x7)	1	7
Consultas	Baja (x3)	2	6
	Media (x4)	0	0
	Alta (x6)	0	0
Fichero Log. Interno	Baja (x7)	0	0
	Media (x10)	0	0
	Alta (x15)	1	15
Fichero Log. Externo	Baja (x5)	0	0
	Media (x7)	0	0
	Alta (x10)	1	10
TOTAL PFSA			98

Tabla 19. Puntos de función sin ajustar.

De acuerdo con el paso número 4, asignamos un peso a los 14 factores según su influencia en la *Tabla 20*.

Con el factor de ajuste, y teniendo en cuenta que conocemos que el proyecto va a ser desarrollado en JavaEE (aproximadamente un 65% del código), Html & Facelets (aproximadamente un 34% del código) y JavaScript (aproximadamente un 1%). Podemos determinar el total de líneas de código del proyecto, gracias a las equivalencias que podemos ver en la *Tabla 21*.

Para obtener el número de líneas de código de cada lenguaje, multiplicamos su equivalencia de líneas de código por los Puntos de Función Ajustados (PFA), y después aplicamos el

porcentaje de código, respecto del total del proyecto, que tiene ese lenguaje. El total de líneas de código será la suma del resultado de esta operación en los tres lenguajes utilizados en el proyecto. Estos cálculos, y el resultado total de Líneas De Código se pueden observar en la *Tabla 22*.

Factor	Valor
1. Comunicación de datos	3
2. Procesamiento distribuido	3
3. Rendimiento	4
4. Desempeño Crítico	4
5. Frecuencia de transacciones	4
6. Entrada de Datos en Línea	5
7. Transacción de Entrada sobre Pantallas Múltiples	2
8. Actualización en Línea	4
9. Procesos complejos	3
10. Facilidad de mantenimiento	2
11. Código Diseñado para Reutilización	2
12. Conversión/ Instalación en Diseño	2
13. Instalaciones Múltiples	2
14. Facilidad de cambios	3
Factor de Ajuste de Valor (0.01 * ΣFC)	0,43

Tabla 20. Factores de ajuste y pesos.

Lenguaje	LDC/PF
HTML	34
J2EE	46
JavaScript	47

Tabla 21. Equivalencia entre Punto de Función y Líneas de Código. Fuente [QSM](#).

Líneas DE CÓDIGO Y ESTIMACIONES FINALES	Total
PFA = (PFSA*(FA+0,65))	105,84
Líneas de código 1PF = 46LDC en javaEE *65%	3165
Líneas de código 1PF = 34LDC en HTML/Facelets *34%	1224
Líneas de código 1PF = 47LDC en JS *1%	50
Total LDC	4439

Tabla 22. Líneas de código y estimaciones finales.

3.3.2 Estimación de Esfuerzo por COCOMO

Se trata de una estimación para medir el tiempo y esfuerzo que será necesario emplear para llevar a cabo el desarrollo de un sistema. Este método fue desarrollado por B. W. Boehm a finales de los años setenta. Para su utilización es necesario el conocimiento del número de

líneas de código que se prevé que tendrá el futuro proyecto (este valor le hemos obtenido en el apartado anterior).

Se pueden aplicar tres modelos distintos de COCOMO según las características del proyecto y los desarrolladores que participaran en su desarrollo:

- **Orgánico:** se utiliza cuando un grupo de programadores experimentados o familiarizados con una determinada tecnología, llevan a cabo un desarrollo no muy extenso y con requisitos poco rígidos.
- **Empotrado:** se trata de un desarrollo con requisitos muy estrictos, siendo un problema no trivial o frecuente y con dificultades para resolverlo desde la experiencia.
- **Semi-empotrado:** se trata de un modelo intermedio entre los dos anteriores, que puede contar con un equipo de desarrollo con una mezcla de personas experimentadas y noveles.

Dependiendo del modelo elegido se usan unos coeficientes u otros para los cálculos, siendo estos los siguientes de la *Tabla 23*:

Modelo COCOMO	Coeficientes			
	a	b	c	d
Orgánico	2,4	1,05	2,5	0,38
Semi-empotrado	3	1,12	2,5	0,35
Empotrado	3,6	1,2	2,5	0,33

Tabla 23. Coeficientes COCOMO. Fuente [Wikipedia](#).

En este proyecto se va a considerar un desarrollo Orgánico, debido al pequeño número de desarrolladores y pocas líneas de código, a pesar de que la complejidad técnica no sea trivial debido a la poca experiencia de desarrollo.

Para aplicar COCOMO además es necesario aplicar 15 valores distintos relacionados con atributos del proyecto, el entorno operativo y el equipo de desarrollo. Estos pueden verse en la *Tabla 24*. Puede observarse que los factores elegidos para cada atributo han sido destacados en color gris en la tabla anterior. La multiplicación entre sí de todos estos factores nos dará un valor necesario para obtener 'm(x)', una variable necesaria para realizar los cálculos.

Los cálculos que vamos a desarrollar nos permitirán obtener el 'Esfuerzo' (**E**), que se trata del número de personas por mes que requerirá el desarrollo del proyecto, el 'Tiempo de Desarrollo' (**Tdev**), que es el tiempo que requerirá desarrollar el proyecto, y por último el 'Esfuerzo Nominal' (**N**), que es el número de desarrolladores necesarios para realizar el proyecto en el tiempo estimado.

$$E = a \cdot (KLDC)^b \cdot m(x) = 2,4 \cdot (4,439)^{1,05} \cdot 1,32934432995 = 15,258 \text{ personas/mes.}$$

$$Tdev = c \cdot E^d = 2,5 \cdot (15,258)^{0,38} = 7,0415 \approx \mathbf{7 \text{ meses}}$$

$$N = E/Tdev = 15,258/7,0415 = 2,167 \approx \mathbf{3 \text{ personas}}$$

De acuerdo con los resultados obtenidos, podemos afirmar que la estimación de este proyecto por el método de COCOMO será de un desarrollo de tres personas durante 7 meses y algunos días extra.

Factor	Valor					
	Muy bajo	Bajo	Medio	Alto	Muy alto	Extra
Fiabilidad requerida	0,75	0,88	1,00	1,15	1,4	
Tamaño de la base de datos		0,94	1,00	1,08	1,16	
Complejidad del software	0,7	0,85	1,00	1,15	1,16	
Restricciones de tiempo de ejecución			1,00	1,11	1,30	1,65
Restricciones de memoria			1,00	1,06	1,21	1,66
Volatilidad del hardware		0,87	1,00	1,15	1,30	
Restricciones de tiempo de respuesta		0,87	1,00	1,07		
Calidad de los analistas	1,46	1,19	1,00	0,86	0,71	
Experiencia con el tipo de aplicación	1,29	1,13	1,00	0,91	0,82	
Experiencia con el hardware	1,21	1,10	1,00	0,90		
Experiencia con el lenguaje de programación	1,14	1,07	1,00	0,95		
Calidad de los programadores	1,42	1,17	1,00	0,86	0,70	
Técnicas modernas de programación	1,24	1,10	1,00	0,91	0,82	
Empleo de herramientas	1,24	1,10	1,00	0,91	0,83	
Restricciones a la duración del proyecto	1,23	1,08	1,00	1,04	1,10	

Tabla 24. Factores de esfuerzo COCOMO.

3.4 Presupuesto

En este apartado se pretende hacer una estimación de los costes que supondría el desarrollo de este proyecto de forma profesional. Para ello será necesario realizar un cálculo de los costes de Hardware, Software y recursos humanos.

3.4.1 Presupuesto Hardware

Para la elaboración del presupuesto de Hardware, vamos a tener en cuenta que los equipos adquiridos tienen una amortización estimada, por lo que al presupuesto de este proyecto solo se cargará el coste de amortización para el periodo de duración de este.

El coste de amortización por año se obtiene mediante la siguiente fórmula:

$$CA = \frac{\text{Coste total del producto (€)}}{\text{Duración del producto (años)}}$$

Como la duración del proyecto será de medio año, el resultado obtenido será dividido entre dos, para conocer la amortización durante 6 meses.

Componente	Coste Total	Amortización estimada	Coste en proyecto
Xiaomi Notebook Pro 15.6 - Intel i7 - 16GB RAM	1.061 €	8 años	66,32€
Xiaomi Mi Mouse 2	15€	10 años	0,75€
HP ProLiant MicroServer Gen10 AMD Opteron X3216/8GB	392€	4	49€
Coste total de Hardware			67.07€

Tabla 25. Presupuesto de Hardware.

3.4.2 Presupuesto Software

Al igual que en el presupuesto anterior, vamos a tener en cuenta la amortización estimada para el periodo de duración del proyecto.

Componente	Coste Total	Amortización estimada	Coste en proyecto
Ubuntu Server 16.04 LTS	0 €	5 años	0€
NetBeans 8.2	0€	-	0€
Glassfish Server 4.1	0€	-	0€
Windows 10 (Incluido en Xiaomi Notebook)	0€	-	0€
Microsoft Office 2016 (Hogar & Estudiante - 1 licencia)	149€	4 años	18,63€
Coste total de Software			18,63€

Tabla 26. Presupuesto de Software.

3.4.3 Presupuesto de Recursos Humanos

Según lo estimado en el *punto 3.2.2* de este documento, el desarrollo del proyecto requiere de 3 desarrolladores durante el periodo de 7 meses.

Según el convenio colectivo de consultoría⁴, un Analista-Programador (Categoría CIII), obtiene una remuneración anual de 20.486,70€. Como el proyecto tiene una duración de 7 meses, debemos de tener en cuenta que corresponde pagar una paga extra.

$$20.486,70/14 = 1.463'34 \text{ € brutos mensuales}$$

Para obtener el gasto por un empleado, debemos de multiplicar el coste mensual por 8 meses (7 de proyecto + paga extra).

$$1.463'34 \cdot 8 = 11.706,72\text{€}$$

Por lo que el coste laboral de tres desarrolladores para la realización del proyecto será de **35.120,16€**.

3.4.4 Presupuesto total del proyecto

Una vez calculados los presupuestos se Hardware, Software y recursos humanos los sumamos para obtener el presupuesto total del proyecto.

Concepto	Total
Hardware	67,07€
Software	18,63€
Recursos humanos	35.120,16
Presupuesto TOTAL	35.205,86€

Tabla 27. Presupuesto total.

Obtenemos un presupuesto total para el proyecto de **35.205,86€**.

3.5 Esfuerzo real del proyecto

En este apartado se va a comentar lo que ha supuesto, una vez finalizado, el esfuerzo de desarrollo del proyecto.

Antes de realizar el cálculo real del proyecto, es importante comentar que las estimaciones no son sensibles al uso de Frameworks de desarrollo que ahorran gran tiempo al desarrollador. En este proyecto se usan tecnologías como 'Java Server Faces' y 'Primefaces', que reducen el

⁴ Convenio colectivo estatal de empresas de consultoría y estudios de mercado y de la opinión pública año 2018: <https://www.boe.es/boe/dias/2018/03/06/pdfs/BOE-A-2018-3156.pdf>

tiempo de desarrollo, además de la gran posibilidad de reutilización de código que permite JavaEE.

Aun así, el proyecto sufre un retraso de **22 días** en su finalización. Para explicar esta mayor duración del desarrollo, cuando debería de haber sido menor por lo anteriormente expuesto, influyen varios factores:

1. El desarrollo no pudo ser realizado a tiempo completo debido a obligaciones laborales ajenas al proyecto.
2. Parón en la etapa de análisis de la primera iteración, desde el día 20/11/17 al 29/11/17 debido a problemas de salud.
3. Retraso de inicio de la etapa de diseño de la primera iteración, entre el 22/12/17 y el 10/01/18, debido a aumento de horas en la jornada laboral.
4. Retraso significativo en la etapa de implementación de la segunda iteración, debido a un parón entre el 19/03/18 al 28/03/18 motivado por un aumento de jornada de trabajo.
5. Aumento de los días previsto en diversas fases de implementación debido a dificultades técnicas no esperadas, en especial en la segunda iteración.

A continuación, se incluye el detalle de tiempos reales empleados en cada iteración y fase (ver Figura 106).

Etapas proyecto	Días		Inicio	Final
ITERACIÓN I	64.0	109.0	13-11-17	08-02-18
Análisis	23.0		13-11-17	22-12-17
Diseño	11.0		10-01-18	24-01-18
Implementación	7.0		25-01-18	02-02-18
Pruebas	2.0		05-02-18	06-02-18
Documentación	2.0		07-02-18	08-02-18
ITERACIÓN II	46.0	86.0	09-02-18	13-04-18
Análisis	4.0		09-02-18	14-02-18
Diseño	4.0		15-02-18	20-02-18
Implementación	24.0		21-02-18	03-04-18
Pruebas	4.0		04-04-18	09-04-18
Documentación	4.0		10-04-18	13-04-18
ITERACIÓN III	40.0	78.0	16-04-18	08-06-18
Análisis	5.0		16-04-18	20-04-18
Diseño	3.0		23-04-18	25-04-18
Implementación	20.0		30-04-18	25-05-18
Pruebas	7.0		28-05-18	05-06-18
Documentación	3.0		06-06-18	08-06-18
ITERACIÓN IV	16.0	32.0	11-06-18	02-07-18
Análisis	2.0		11-06-18	12-06-18
Diseño	2.0		13-06-18	14-06-18
Implementación	5.0		15-06-18	21-06-18
Pruebas	3.0		22-06-18	26-06-18
Documentación	4.0		27-06-18	02-07-18

Figura 106. Detalle tiempos reales empleados en las distintas etapas.

Para obtener el coste económico real que hubiera supuesto el desarrollo del proyecto, debemos de añadir al presupuesto los 22 días que se extendió el desarrollo. El coste mensual de un trabajador se estimó en 1.463'34 € brutos mensuales. Suponiendo un mes de 30 días, el salario bruto por día y trabajador de 48,78€. Por lo que el aumento de presupuesto en recursos humanos para 3 trabajadores en 22 días sería de 3.219'35€.

Sumando este aumento a lo anteriormente presupuestado, obtendríamos un presupuesto final de desarrollo 38.425,21€.

Aunque ese presupuesto abarca la fase de desarrollo del proyecto, para el despliegue de la plataforma sería necesario invertir en una plataforma. Para ello se podría optar por varias soluciones, como adquirir un servidor dedicado, usar virtualización o usar servicios en la nube como Amazon o Azure, aunque esto queda fuera del alcance proyecto. Para probar el despliegue de la herramienta la Universidad de Valladolid ha proporcionado una máquina virtual en un entorno de virtualización Proxmox.

Capítulo 4

Análisis

En esta capítulo se recogen los distintos procesos de estudio del comportamiento que deberá tener el futuro sistema a desarrollar.

4.1 Actores del sistema

En esta sección se nombran y describen todos los actores que participan en Plataforma de gestión de Itinerarios GPSSM. Un actor es un usuario o sistema externo que interactúa con el sistema generando una respuesta.

Los actores que van a intervenir en el sistema son los siguientes (*Tabla 28*):

ACTOR	DESCRIPCIÓN
Plataforma móvil	Se trata del actor que engloba todas las acciones de comunicación con las plataformas software móviles que se conectan al sistema,
Usuario Web Registrado	Se trata de los usuarios que están accediendo a portal web mientras están identificados en el sistema.
Gestor	Se trata de un grupo de usuarios, identificados en el sistema, con permiso para acceder a la gestión de los puntos de interés. Este tipo de usuario no puede ser registrado por si mismo.
Usuario General Invitado	Engloba a los usuarios de la plataforma que están accediendo a ella, sin estar identificados en el sistema, y sin importar el medio de acceso que estén utilizando, ya sea el portal web o un sistema externo.
Usuario General Registrado	Engloba a los usuarios de la plataforma que están accediendo a ella, identificados en el sistema, y sin importar el medio de acceso que estén utilizando, ya sea el portal web o un sistema externo.

Tabla 28. Lista de actores del sistema.

Algunos de estos actores se relacionan jerárquicamente, estableciendo una relación de herencia. Esta relación indica que las acciones de un determinado actor, son asumidas como propias por otro actor con el cuál se encuentra relacionado. La jerarquía de los actores del sistema puede verse en la *Figura 107*.

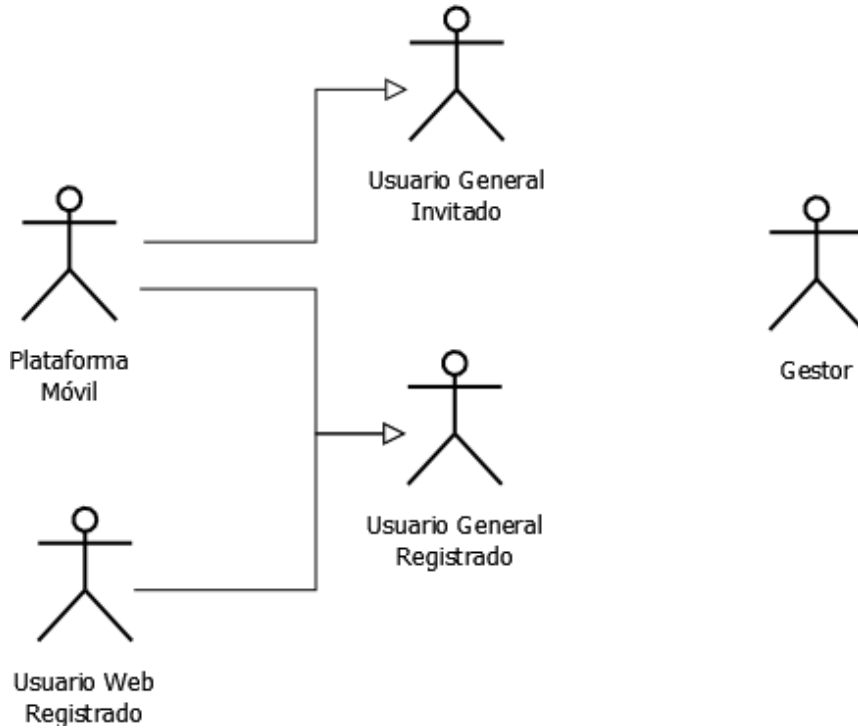


Figura 107. Jerarquía entre actores del sistema.

La descripción de la secuencia de acciones que se producen entre actor y sistema se representa mediante casos de uso. Los casos de uso son un método de especificación de requisitos de usuario, ya que describen el estado del sistema después de cada interacción del usuario.

En la *Figura 107* podemos interpretar que el actor 'Plataforma móvil' heredan los casos de uso del actor 'Usuario General Invitado'. De igual manera podemos interpretar que los casos de uso del actor 'Usuario General Registrado' son heredados por los actores 'Usuario Web Registrado' y 'Plataforma móvil'.

4.2 Requisitos de Usuario

Un requisito de usuario es cada objetivo o tarea que un usuario puede realizar en el sistema. En este apartado se especifican los Requisitos de Usuario a través de Casos de Uso.

4.2.1 Listado de casos de uso

Como paso previo a la especificación de los distintos Casos de Uso, se clasifican estos según el actor que los origina. La clasificación puede consultarse en la *Tabla 29*.

ACTOR	CASOS DE USO
Gestor	GS1 – GESTIONAR PUNTOS DE INTERÉS GS2 – CREAR GS3 – MODIFICAR GS4 – ELIMINAR
Usuario General Invitado	UGI1 – REGISTRARSE UGI2 - LOGIN
Usuario General Registrado	UGR1 – GESTIONAR ITINERARIOS PROPIOS UGR2 – EDITAR ATRIBUTOS UGR3 – EDITAR INFO PUNTO MULTIMEDIA UGR4 – ELIMINAR ITINERARIO UGR5 – BUSCAR USUARIO UGR6 – VER PERFIL UGR7 – SOLICITAR AMISTAD UGR8 – CONSULTAR ITINERARIOS DE USUARIO UGR9 – CONSULTAR LISTADO DE SOLICITUDES UGR10 – RESOLVER SOLICITUD UGR11 – CONSULTAR LISTADO DE AMISTADES UGR12 – ELIMINAR AMISTAD UGR13 – BUSCAR ITINERARIO UGR14 – CONSULTAR ITINERARIO
Usuario Web Registrado	UWR1 - LOGOUT
Plataforma Móvil	PM1 – ENVIAR ITINERARIO CREADO PM2 – ENVIAR CONTENIDO MULTIMEDIA PM3 – CONSULTAR CUENTA PROPIA PM4 – MODIFICAR CUENTA PM5 – ELIMINAR CUENTA PM6 – OBTENER PUNTOS DE INTERÉS PM7 – OBTENER CONTENIDO MULTIMEDIA

Tabla 29. Lista de casos de uso.

Los casos de uso pueden ser ejecutados directamente por el actor, o como consecuencia del flujo de otro caso de uso relacionado. Hay dos tipos de relaciones entre casos de uso:

- **Include:** La ejecución completa de un caso de uso requiere la ejecución obligatoria de otro caso de uso. Usando de ejemplo la *Figura 108*, la relación entre el caso de uso CU1 y el caso de uso CU2, es una relación de este tipo, en la que la ejecución de CU1 produce la ejecución de CU2 de forma obligatoria.
- **Extend:** la ejecución de un caso de uso, según la secuencia de acciones seguida por el actor, puede hacer que se ejecute o no otro caso de uso. Usando de ejemplo la *Figura 108*, la ejecución de CU3 puede provocar la ejecución de CU4 según las acciones realizadas por el usuario en la secuencia de CU3. Se trata de una relación de especialización.

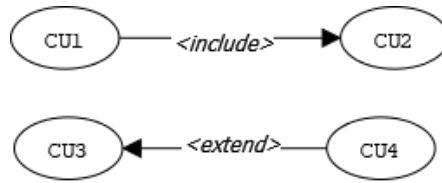


Figura 108. Ejemplo de relaciones entre casos de uso.

El diagrama de casos de uso nos permite representar los casos de uso asociados a un actor, así como las posibles relaciones existentes entre estos casos de uso. Puede verse a continuación el diagrama de casos de uso del proyecto en la Figura 109.

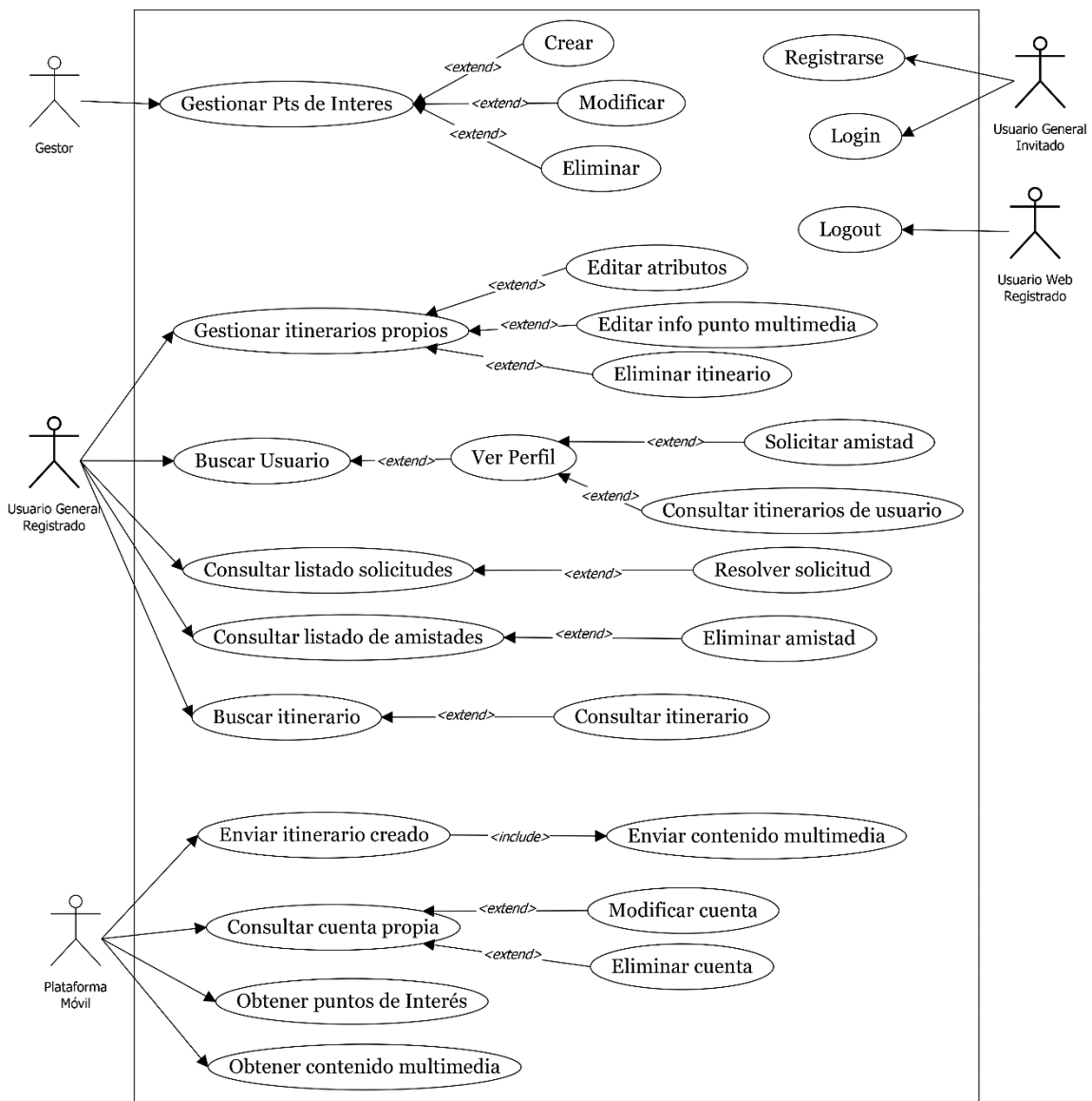


Figura 109. Diagrama de casos de uso.

4.2.2 Especificación de Casos de Uso

En este apartado se incluirán las tablas de especificación de casos del uso del actor ‘Gestor’ a modo de ejemplo. La especificación completa de todos los casos de uso puede encontrarse en el **ANEXO I** de este documento debido a su gran extensión.

ID y NOMBRE:	GS1 – GESTIONAR PUNTOS DE INTERÉS						
Versión	1.0						
Actor principal	Gestor						
Actores secundarios	<i>No contemplado</i>						
Descripción	El gestor desea consultar y gestionar los puntos de realidad aumentada que están registrados en la plataforma.						
Precondición	PRE-1: El gestor se encuentra identificado en el sistema.						
Secuencia normal	1 – El gestor solicita los puntos de interés existentes en el sistema. 2 – El sistema consulta el listado de puntos de interés registrados en el sistema (DEP-2) (EXCP-1). 3 – El usuario puede realizar 3 acciones sobre puntos de RA. Que desea añadir un punto de RA (4a), que desea modificar un punto de RA (4b) o que desea eliminar un punto de RA (4c). O bien, puede finalizar el CU. 4 - El sistema sigue una secuencia alternativa según la elección del usuario.						
Secuencia alternativa	<table border="1"> <tbody> <tr> <td>A</td> <td>4a.1 – El sistema ejecuta el CU “GS2 - CREAR”.</td> </tr> <tr> <td>B</td> <td>4b.1 – El gestor indica que desea modificar un punto de realidad virtual. 4b.2 – El sistema ejecuta el CU “GS2 - MODIFICAR”.</td> </tr> <tr> <td>C</td> <td>4c.1 – El gestor indica que desea eliminar un punto de realidad virtual. 4c.2 – El sistema ejecuta el CU “GS3 - ELIMINAR”.</td> </tr> </tbody> </table>	A	4a.1 – El sistema ejecuta el CU “GS2 - CREAR”.	B	4b.1 – El gestor indica que desea modificar un punto de realidad virtual. 4b.2 – El sistema ejecuta el CU “GS2 - MODIFICAR”.	C	4c.1 – El gestor indica que desea eliminar un punto de realidad virtual. 4c.2 – El sistema ejecuta el CU “GS3 - ELIMINAR”.
A	4a.1 – El sistema ejecuta el CU “GS2 - CREAR”.						
B	4b.1 – El gestor indica que desea modificar un punto de realidad virtual. 4b.2 – El sistema ejecuta el CU “GS2 - MODIFICAR”.						
C	4c.1 – El gestor indica que desea eliminar un punto de realidad virtual. 4c.2 – El sistema ejecuta el CU “GS3 - ELIMINAR”.						
Postcondición	POST-1: El sistema muestra por pantalla el resultado de puntos de interés existentes en la plataforma.						
Excepciones	EXCP-1: Si no hay puntos de interés añadidos, el sistema genera un mensaje indicando que aún no hay elementos añadidos.						
Prioridad	Media						
Dependencias	DEP-1: Regla de negocio RN-26 DEP-2: Requisitos de información RI.EPOI.01, RI.EPOI.02, RI.EPOI.03, RI.EPRA.01, RI.EPRA.02, RI.EPRA.03, RI.EPRA.04, RI.EPRA.05						
Comentarios							

Tabla 30. Especificación caso de uso GS1.

ID y NOMBRE:	GS2 - CREAR
Versión	1.0
Actor principal	Gestor
Actores secundarios	<i>No contemplado</i>
Descripción	El gestor de la plataforma desea registrar un nuevo punto de interés
Precondición	PRE-1: El gestor debe de estar identificado en la plataforma.
Secuencia normal	<p>1 – El gestor indica que quiere añadir un nuevo punto de interés</p> <p>2 – El sistema muestra al gestor un formulario con la información descriptiva del punto de interés.</p> <p>3 – El usuario rellena el formulario ofrecido por el sistema en el paso 2.</p> <p>4 – El sistema almacena la información del formulario ofrecido en el paso 2, y muestra un nuevo formulario en el que el usuario podrá introducir una imagen de referencia del punto de realidad aumentada (EXCP-1) (DEP-3).</p> <p>5 – El usuario envía una imagen.</p> <p>6 – El sistema almacena la imagen y obtiene la ruta donde es guardada en el servidor (EXCP-2).</p>
Secuencia alternativa	<i>No contemplada</i>
Postcondición	<p>POST-1: El sistema registra en la base de datos el nuevo punto de interés.</p> <p>POST-2: El sistema genera un mensaje de confirmación de la creación del nuevo punto.</p>
Excepciones	<p>EXCP-1: Si el usuario no ha rellenado todos los campos, el sistema se lo notifica al usuario y el caso de uso vuelve al paso 3.</p> <p>EXCP-2: Si falla el envío de la imagen, el caso de uso vuelve al paso 5.</p>
Prioridad	Media
Dependencias	<p>DEP-1: Regla de negocio RN-26</p> <p>DEP-2: Requisitos de información RI.EPOI.01, RI.EPOI.02, RI.EPOI.03, RI.EPRA.01, RI.EPRA.02, RI.EPRA.03, RI.EPRA.04, RI.EPRA.05</p>
Comentarios	En la base de datos no se almacena la imagen, si no que se almacena directamente en almacenamiento interno.

Tabla 31. Especificación caso de uso GS2.

ID y NOMBRE:	GS3 - MODIFICAR
Versión	1.0
Actor principal	Gestor
Actores secundarios	<i>No contemplado</i>
Descripción	El gestor de la plataforma desea modificar uno de puntos de interés registrados en el sistema.
Precondición	PRE-1: El gestor debe de estar identificado en la plataforma. PRE-2: Existen puntos de interés registrados en el sistema.
Secuencia normal	1 – El gestor indica que quiere modificar un punto de interés existente. 2 – El sistema recupera la información sobre el punto de interés y se la muestra al usuario. 3 - El usuario indica el campo de información que desea modificar. 4 – El sistema recibe la modificación realizada por el usuario sobre la información original.
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema registra en la base de datos las modificaciones sobre el punto de realidad aumentada modificado. POST-2: El sistema muestra por pantalla los cambios realizados.
Excepciones	<i>No contempladas</i>
Prioridad	Media
Dependencias	DEP-1: Requisitos de información RI.EPOI.02, RI.EPOI.03, RI.EPRA.01, RI.EPRA.02, RI.EPRA.04, RI.EPRA.05
Comentarios	No se permiten modificar imágenes o coordenadas, se considera la información básica del punto.

Tabla 32. Especificación caso de uso GS3.

ID y NOMBRE:	GS4 - ELIMINAR
Versión	1.0
Actor principal	Gestor
Actores secundarios	<i>No contemplado</i>
Descripción	El gestor de la plataforma desea eliminar un punto de interés registrado en la plataforma.
Precondición	PRE-1: El gestor debe de estar identificado en la plataforma. PRE-2: Existen puntos de realidad aumentada registrados en el sistema.
Secuencia normal	1 – El gestor indica que quiere eliminar un punto de interés (DEP-1). 2 – El sistema pide confinación al usuario para proceder a la eliminación del punto de interés. 3 – El usuario confirma la eliminación del caso de uso (EXCP-1). 4 – El sistema finaliza el caso de uso con POST-A .
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-A: El sistema elimina de la base de datos el punto de interés indicado.
Postcondición alternativa	POST-B: El sistema no modifica el estado del punto en la base de datos.
Excepciones	EXCP-1: Si el usuario no confirma la eliminación del punto, el caso de uso finaliza con POST-B .
Prioridad	Media
Dependencias	DEP-1: Requisito de información RI.EPOI.01
Comentarios	

Tabla 33. Especificación caso de uso GS4.

4.3 Reglas de Negocio

Las reglas de negocio son políticas, estándares o regulaciones que rigen el funcionamiento de una organización, empresa o sistema. Son normas establecidas antes del inicio del desarrollo del Software y ayudan a moldear la lógica de negocio.

ID	Descripción
RN.01	Los itinerarios son creados a través de un dispositivo móvil.
RN.03	Los usuarios buscaran itinerarios por título y/o por ubicación.
RN.04	No puede crease más de un itinerario simultáneamente desde
RN.05	No se podrá modificar el contenido multimedia o recorrido de un itinerario.
RN.06	Los atributos descriptivos de un itinerario solo pueden modificarse una vez este ya está publicado.
RN.07	La modificación de atributos de un itinerario puede realizarse desde el portal web, o plataforma móvil. Solo son modificables título, extracto y privacidad.
RN.08	Solo puede modificar o eliminar un itinerario su correspondiente creador.
RN.09	Solo podrán contener una descripción aquellas posiciones que tengan asociadas contenido multimedia
RN.10	Los puntos multimedia no pueden ser eliminados del itinerario
RN.11	Los itinerarios tienen un estado de privacidad asociado que puede ser "PUBLICO", "AMISTAD" o "PRIVADO".
RN.12	Los itinerarios con el estado de privacidad "PRIVADO" solo podrán ser visualizados por su creador.
RN.13	Los itinerarios con el estado de privacidad "AMISTAD" podrán ser visualizados por su creador y sus amigos en la plataforma.
RN.14	Los itinerarios con el estado de privacidad "PUBLICO" podrán ser visualizados por cualquier usuario registrado en la plataforma.
RN.15	Los usuarios serán identificados en el sistema con un nombre de usuario único.
RN.16	Al eliminar un usuario su cuenta, se borrará toda su información personal y todos sus itinerarios propios de la plataforma.
RN.17	Las contraseñas tendrán una longitud mínima de 8 caracteres.
RN.18	Las contraseñas serán codificadas mediante SHA256.
RN.20	Los usuarios accederán al sistema mediante su identificador único y una contraseña de acceso definida en el registro.
RN.21	Las amistades se establecerán en el sistema mediante peticiones de amistad entre usuarios, que podrán ser aceptadas o rechazadas por su destinatario
RN.25	Los puntos de realidad aumentada solo podrán ser añadido y administrados a través de la plataforma software empresarial.
RN.26	Los puntos de realidad aumentada solo podrán ser añadidos y administrados por un usuario 'Gestor'

Tabla 34. Reglas de negocio.

4.4 Requisitos Funcionales

Los requisitos funcionales son una descripción del comportamiento que debe mostrar el sistema bajo condiciones determinadas.

Para facilitar su comprensión y organización, se van a agrupar según los casos de uso listados en el *punto 5.2.1*, clasificándolos a su vez por los actores listados en el *punto 4.1*. En este apartado se incluyen los requisitos funcionales especificados a partir de los casos de uso del actor 'Gestor' a modo de ejemplo. La especificación completa de todos los requisitos funcionales puede encontrarse en el **ANEXO III** de este documento debido a su gran extensión.

Caso de Uso	Descripción
GS1	Gestionar puntos de interés
Requisito Funcional	Descripción
RF.01	El sistema comprueba que el usuario tiene sesión abierta en el sistema y el tipo de usuario es 'Gestor'.
RF.02	El sistema consulta en la base de datos los puntos de interés.
RF.03	El sistema muestra por pantalla los puntos de interés.
RF.04	Si no existen puntos de interés registrados en la plataforma el sistema se lo notifica por pantalla al usuario.

Tabla 35. Requisitos funcionales asociados al caso de uso GS1.

Caso de Uso	Descripción
GS2	Crear
Requisito Funcional	Descripción
RF.01	(Ver en requisitos funcionales de GS1).
RF.05	El sistema muestra por pantalla al usuario un formulario para obtener la información descriptiva del punto.
RF.06	El sistema recoge la información introducida por el usuario en el formulario de información descriptiva.
RF.07	Si el usuario no ha rellenado todos los campos, el sistema se lo notifica al usuario y vuelve a mostrarle el formulario.
RF.08	El sistema ofrece al usuario un formulario para subir una imagen.
RF.09	El sistema recoge la imagen introducida por el usuario en el formulario y la almacena.
RF.10	Si falla la subida de la imagen, el sistema se lo notifica al usuario.
RF.11	El sistema almacena en la base de datos el nuevo punto de interés.
RF.12	El sistema muestra una confirmación al usuario de que el punto ha sido creado.

Tabla 36. Requisitos funcionales asociados al caso de uso GS2.

Caso de Uso	Descripción
GS3	Modificar
Requisito Funcional	Descripción
RF.01	<i>(Ver en requisitos funcionales de GS1).</i>
RF.13	El sistema comprueba que existen puntos de interés en la base de datos.
RF.14	El sistema recoge el identificador del punto de interés que desea modificar el usuario.
RF.15	El sistema recoge la nueva información de los atributos del punto.
RF.16	El sistema registra en la base de datos las modificaciones realizadas por el usuario.
RF.17	El sistema muestra por pantalla al usuario el resultado de los cambios realizados.

Tabla 37. Requisitos funcionales asociados al caso de uso GS3.

Caso de Uso	Descripción
GS4	Eliminar
Requisito Funcional	Descripción
RF.01	<i>(Ver en requisitos funcionales de GS1).</i>
RF.13	<i>(Ver en requisitos funcionales de GS3).</i>
RF.18	El sistema recoge el identificador del punto de interés que desea borrar el usuario.
RF.19	El sistema pide confirmación al usuario para eliminar el punto.
RF.20	Si el usuario confirma, el sistema elimina de la base de datos el punto de interés deseado.

Tabla 38. Requisitos funcionales asociados al caso de uso GS4.

4.5 Requisitos No Funcionales

Los requisitos no funcionales son características y restricciones que debe de seguir el funcionamiento del sistema.

4.5.1 Atributos de calidad

Describen las condiciones en las que el usuario espera que el sistema realice su funcionalidad. Se contemplan varios tipos: usabilidad, rendimiento, seguridad y disponibilidad.

Usabilidad: Son aquellos relacionados con la facilidad de uso y accesibilidad.

ID	Descripción
ATU1	Un usuario que haya leído el manual debe de ser capaz de utilizar cualquier funcionalidad del sistema en menos de 20 min.
ATU2	La gestión de la información, en las operaciones de edición y borrado, se desarrollará siempre en una misma pantalla, recargando los elementos que sean necesarios en esta misma para mostrar los resultados, para una mayor sencillez de uso.
ATU3	La aplicación será perfectamente operativa y correctamente renderizada en los principales navegadores web (Firefox, Chrome, Safari).
ATU4	La subida de imagen de un punto de interés al servidor podrá ser realizada en segundo plano mientras el usuario rellena el formulario de creación de este.

Tabla 39. Atributos de calidad de Usabilidad.

Rendimiento: Son aquellos relacionados con el tiempo de respuesta

ID	Descripción
ATR1	El tiempo de respuesta del sistema no será superior a un segundo para conexión usuario de 20Mbps.
ATR2	El sistema soportará el uso concurrente de datos por parte de al menos 50 usuarios, ya sea a través de la interfaz web o una plataforma móvil.

Tabla 40. Atributos de calidad de Rendimiento

Seguridad: Son aquellos relacionados con la protección de la privacidad del usuario y su información.

ID	Descripción
ATS1	La apertura de sesión en el sistema por parte de los usuarios registrados requerirá la identificación de los mismos a través de la introducción de Nick y contraseña en un formulario.
ATS2	Las contraseñas de los usuarios serán almacenadas de forma cifrada en la base de datos bajo el algoritmo SHA-256 en base 64.
ATS3	Las contraseñas tendrán una longitud mínima de 8 caracteres.

Tabla 41. Atributos de calidad de Seguridad.

Disponibilidad: Son aquellos que miden el tiempo que debe estar operativo el sistema.

ID	Descripción
ATD1	La aplicación deberá tener una disponibilidad de 24 horas al día los 7 días de la semana.

Tabla 42. Atributos de calidad de Disponibilidad

Escalabilidad: Son aquellos que miden el tiempo que debe estar operativo el sistema.

ID	Descripción
ATE1	El sistema deberá soportar modificaciones en su estructura física, sin costes elevados, para poder escalar el sistema ante un aumento de demanda de recursos.

Tabla 43. Atributos de calidad de Escalabilidad.

4.5.2 Restricciones de implementación y arquitectura

Las restricciones de implementación son requisitos que condicionan las tecnologías que serán usadas para el desarrollo del software.

ID	Descripción
RIM1	El diseño de la plataforma seguirá una arquitectura de 3 capas.
RIM2	La plataforma estará desarrollada bajo el lenguaje de programación JavaEE.
RIM3	La parte Web de la plataforma estará desarrollada con la combinación de Frameworks JSF + Primefaces.

Tabla 44. Restricciones de implementación.

4.5.3 Requisitos de Interfaces Externas

Describen la tecnología usada para desarrollar las comunicaciones del sistema con otros sistemas software.

ID	Descripción
RIE1	La comunicación con las plataformas móviles a través de Internet se realizará mediante protocolo HTTP.
RIE2	La plataforma implementará servicios REST para servir y recibir la información.
RIE3	Los servicios REST podrán ofrecer y recibir las entidades de información en formatos XML y Json.
RIE4	La transmisión de datos entre la plataforma empresarial y las plataformas móviles se realizará en formato Json.

Tabla 45. Requisitos sobre interfaces de Interfaces Externas.

4.6 Modelo conceptual de datos

En este apartado se realiza un análisis de la base de datos que soportará la persistencia de la aplicación empresarial.

4.6.1 Modelo Entidad-Relación

Este análisis se realizará mediante un modelo de diseño denominado 'Diagrama Entidad-Relación'. El modelo Entidad-Relación es una herramienta para el modelado de datos, ideado para representar los conjuntos de datos como entidades con propiedades y relaciones.

El modelo conceptual obtenido mediante el análisis del sistema puede verse a continuación en la *Figura 110*.

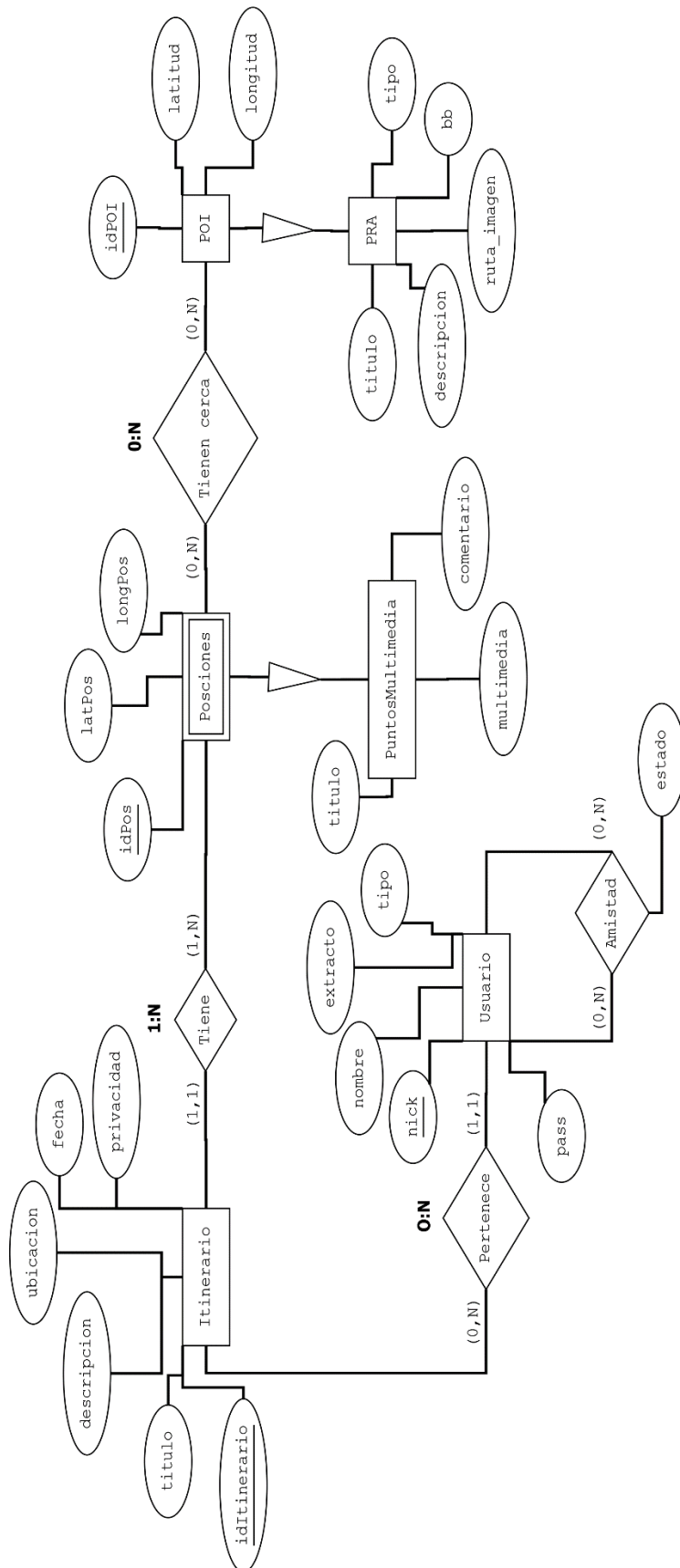


Figura 110. Diagrama Entidad-Relación.

4.6.2 Diccionario de datos

En el diccionario de datos se reflejan con detalle todos los elementos del flujo de datos del sistema, recogidos a través del modelo Entidad-Relación. También se identifican y describen las relaciones entre entidades.

Cada requisito de información se corresponde con un atributo. Estos están organizados por entidades.

Entidad usuario

Entidad que representa a todos los usuarios que van a utilizar la plataforma empresarial, así como la plataforma móvil.

ID	ATRIBUTO	DESCRIPCIÓN	TIPO & LONGITUD	RESTRICCIÓN
RI.EU.01	nick	Identificador que identifica de forma única al usuario en la plataforma.	varchar(12)	primary key
RI.EU.02	nombre	Nombre y apellidos del usuario.	varchar(50)	not null
RI.EU.03	extracto	Descripción sobre el usuario que figurará de forma pública en su perfil.	varchar(100)	not null
RI.EU.04	pass	Contraseña para identificar al usuario en la plataforma. Es una cadena cifrada con MD5	varchar(32)	not null
RI.EU.05	tipo	Define qué tipo de cuenta posee el usuario.	enum('NORMAL', 'GESTOR')	default='NORMAL'

Tabla 46. Atributos entidad Usuario.

Entidad itinerario

Entidad que representa a los recorridos capturados mediante la plataforma web.

ID	ATRIBUTO	DESCRIPCIÓN	TIPO & LONGITUD	RESTRICCIÓN
RI.EI.01	iditinerario	Identificador único del itinerario dentro de la base de datos	int autoincrementet	primary key
RI.EI.02	ubicacion	Ubicación de referencia del itinerario. Indica una localidad en cuyo entorno se localiza el itinerario.	varchar(85)	not null
RI.EI.03	titulo	Nombre representativo para que los usuarios identifiquen el itinerario.	varchar(40)	not null

RI.EI.04	privacidad	Indica el nivel de visibilidad a otros usuarios que desea el creador del itinerario.	emun('PUBLICO', 'AMISTAD', 'PRIVADO')	not null
RI.EI.05	descripcion	Texto que informa sobre que trata el itinerario.	varchar(200)	not null
RI.EI.06	fecha	Fecha del itinerario, proporcionada ya formateada por la plataforma móvil.	String	not null

Tabla 47. Atributos entidad Itinerario.

Entidad Posiciones

Entidad que representa las posiciones, que son las unidades que dan forma al recorrido.

ID	ATRIBUTO	DESCRIPCIÓN	TIPO & LONGITUD	RESTRICCIÓN
RI.EP.01	idPos	Identificador único de la posición.	int autoincrement	primary key
RI.EP.02	latPos	Latitud geográfica de la posición.	float	not null
RI.EP.03	longPos	Longitud geográfica de la posición.	float	not null

Tabla 48. Atributos entidad Posiciones.

Entidad PuntosMultimedia

Entidad que representa a los puntos de un recorrido donde se ha recogido información multimedia. Estos son puntos del recorrido que el usuario desea destacar porque son especiales para él.

ID	ATRIBUTO	DESCRIPCIÓN	TIPO & LONGITUD	RESTRICCIÓN
RI.EPM.01	titulo	Nombre identificativo del punto de interés.	varchar(40)	not null
RI.EPM.02	comentario	Descripción del contenido del punto de interés.	varchar(200)	not null
RI.EPM.03	multimedia	Ruta de almacenamiento del contenido multimedia del punto de interés.	varchar(50)	not null

Tabla 49. Atributos entidad PuntosMultimedia.

Entidad POI

Entidad que representa los puntos de interés, que son puntos sobre ubicaciones específicas, que se presuponen útiles o interesantes para la mayoría de las personas.

ID	ATRIBUTO	DESCRIPCIÓN	TIPO & LONGITUD	RESTRICCIÓN
RI.EPOI.01	idPOI	Identificador único del punto de interés.	int autoincrement	primary key
RI.EPOI.02	latitud	Latitud geográfica del punto de interés.	float	not null
RI.EPOI.03	longitud	Longitud geográfica del punto de interés.	float	not null

Tabla 50. Atributos entidad POI.

Entidad PRA

Entidad que representa un tipo especial de punto de interés. Esta entidad representa la información sobre ubicaciones destacadas que usan para guiar al usuario por lugares destacados de un entorno geográfico.

ID	ATRIBUTO	DESCRIPCIÓN	TIPO & LONGITUD	RESTRICCIÓN
RI.EPRA.01	tipo	Es una palabra descriptiva del contenido de la ubicación que representa el punto de realidad aumentada.	enum ('CULTO', 'PLAZA', 'PANORAMICA', 'CASTILLO', 'MURALLA', 'PAISAJE', 'MONUMENTO', 'ESTATUA', 'MUSEO', 'TEATRO')	not null
RI.EPRA.02	bb	'Bounding box' espacio de búsqueda para encontrar el punto según proximidad.	float	
RI.EPRA.03	ruta_imagen	Ruta de almacenamiento de la imagen del punto de realidad aumentada.	varchar(50)	not null
RI.EPRA.04	descripción	Texto que informa sobre que trata el punto de realidad aumentada.	varchar(200)	not null
RI.EPRA.05	titulo	Nombre representativo para que los usuarios identifiquen punto de realidad aumentada.	varchar(40)	not null

Tabla 51. Atributos entidad PRA.

Relación Pertenece

Esta relación se produce entre la entidad 'Usuario' y la entidad 'Itinerario'. Representa que los itinerarios pertenecen a un usuario.

La cardinalidad es la siguiente: un usuario puede tener varios itinerarios (1,N), y un itinerario solo puede pertenecer a un usuario (1,1).

Relación Tiene

Esta relación se produce entre la entidad 'Itinerario' y la entidad 'Posiciones'. Representa que los itinerarios tienen posiciones que representan su recorrido.

La cardinalidad es la siguiente: un itinerario puede tener varias posiciones (1,N), y una posición solo puede pertenecer a un itinerario (1,1).

Relación Tienen cerca

Esta relación se produce entre la entidad 'Posiciones' y la entidad 'POI'. Representa que algunas de las posiciones de un itinerario están al lado o cercanas a un punto de interés.

La cardinalidad es la siguiente: una posición puede tener cerca ninguno o muchos POI (0,N), y un POI puede estar relacionada con ninguna o muchas posiciones (0,N). Esta relación propagará una nueva tabla (siguiendo un modelo relacional). No se contemplan atributos propios para esta relación.

Aunque esta relación no es necesaria para la funcionalidad actual, es concebida para futuras mejoras y escalabilidad del sistema.

Relación de especialización de Posiciones

Esta relación tiene su origen en la entidad 'Posiciones', esta entidad se divide en la sub-entidad representada por 'PuntosMultimedia'. Esta sub-entidad es necesaria porque define un tipo especial de posiciones.

En esta relación la entidad Posiciones mantiene una relación de 1:1 con su entidad derivada.

Relación de especialización de POI

Esta relación tiene su origen en la entidad 'POI', esta entidad se divide en la sub-entidad representada por 'PRA'. Esta sub-entidad es necesaria porque define un tipo especial de puntos de interés.

En esta relación la entidad POI mantiene una relación de 1:1 con su entidad derivada.

Relación de Amistad

Esta relación se produce sobre la entidad 'Usuario' en sí misma. Representa la relación de amistad entre elementos de la misma entidad.

La cardinalidad es la siguiente: un usuario puede ser amigo de uno o ningún usuarios (0,N). Esta relación propagará una nueva tabla (siguiendo un modelo relacional).

ID	ATRIBUTO	DESCRIPCIÓN	TIPO & LONGITUD	RESTRICCIÓN
RI.EA.01	estado	Indica el estado en el que se encuentra la relación de amistad.	enum ('ACTIVA', 'PENDIENTE')	not null

Tabla 52. Atributos entidad Amistad.

Capítulo 5

Análisis

En esta sección se recoge el proceso de diseño del sistema. La etapa de diseño consiste en determinar los componentes del sistema que dan respuesta a las funcionalidades descritas por los requisitos en la etapa de análisis.

5.1 Arquitectura Lógica

La arquitectura lógica describe los componentes lógicos que componen el sistema y como se relacionan entre sí. Como se ha establecido en el análisis en los requisitos **RMI1**, **RMI2** y **RMI3** (ver punto 4.5.2), el proyecto se desarrollará en la tecnología Java EE con JSF y Primefaces.

La arquitectura de la plataforma empresarial de GPSSSM sigue un diseño en tres capas:

- **Acceso a datos:** Esta capa nos permite acceder al almacenamiento en base de datos (MySQL, ver detalles en *punto 6.3*) gracias al driver de Java para esta base de datos, y la implementación EclipseLink de JPA y el controlador de java para bases de datos JDBC.
- **Lógica de Negocio:** Es la capa encargada del control del flujo de datos del sistema y las operaciones que afectan a estos (creacion, modificación, consulta y borrado). Se divide en 4 grupos eseniales, que son los siguientes:
 - **Clases entidad:** Clases encargadas de realizar el mapeo de datos en objetos para permitir una abstracción sobre la base de datos.
 - **Servicios RESTfull:** Contienen metodos que permiten realizar todas las operaciones de creacion, modificación, consulta y borrado, sobre una o varias entidades, de forma que puedan realizarse dentro de la propia aplicación o desde el exterior (comunicación con plataforma móvil), a través del protocolo HTTP y formatos de intercambio de información como XML y JSON.
 - **JsonPasers:** son un conjunto de clases que se encargan de escribir o leer entidades en JSON, para su manejo a través de los servicios REST.
 - **Enterprise Beans:** Son las clases encargadas de realizar las operaciones necesarias para proporcionar la información a la capa de presentación.
- **Presentación:** Se trata de la capa encargada de gestionar la presentación de la información a los usuarios, que visualizaran a través de un navegador web. Esta desarrollada con las librerías de 'Java Server Faces' (JSF) y el Framework de diseño 'Primefaces'. Está compuesta por el 'Facelets Servlet', que interpreta y sirve las

paginas xhtml, así como por clases 'Backing Bean' que se encargan de guardar el modelo de los datos.

La arquitectura posee también un elemento transversal a todas las capas, encargado de gestionar la seguridad de la aplicación. Esta seguridad está implementada con el API JAAS de JavaEE.

El diagrama que representa la arquitectura lógica puede verse a continuación en la *Figura 111*.

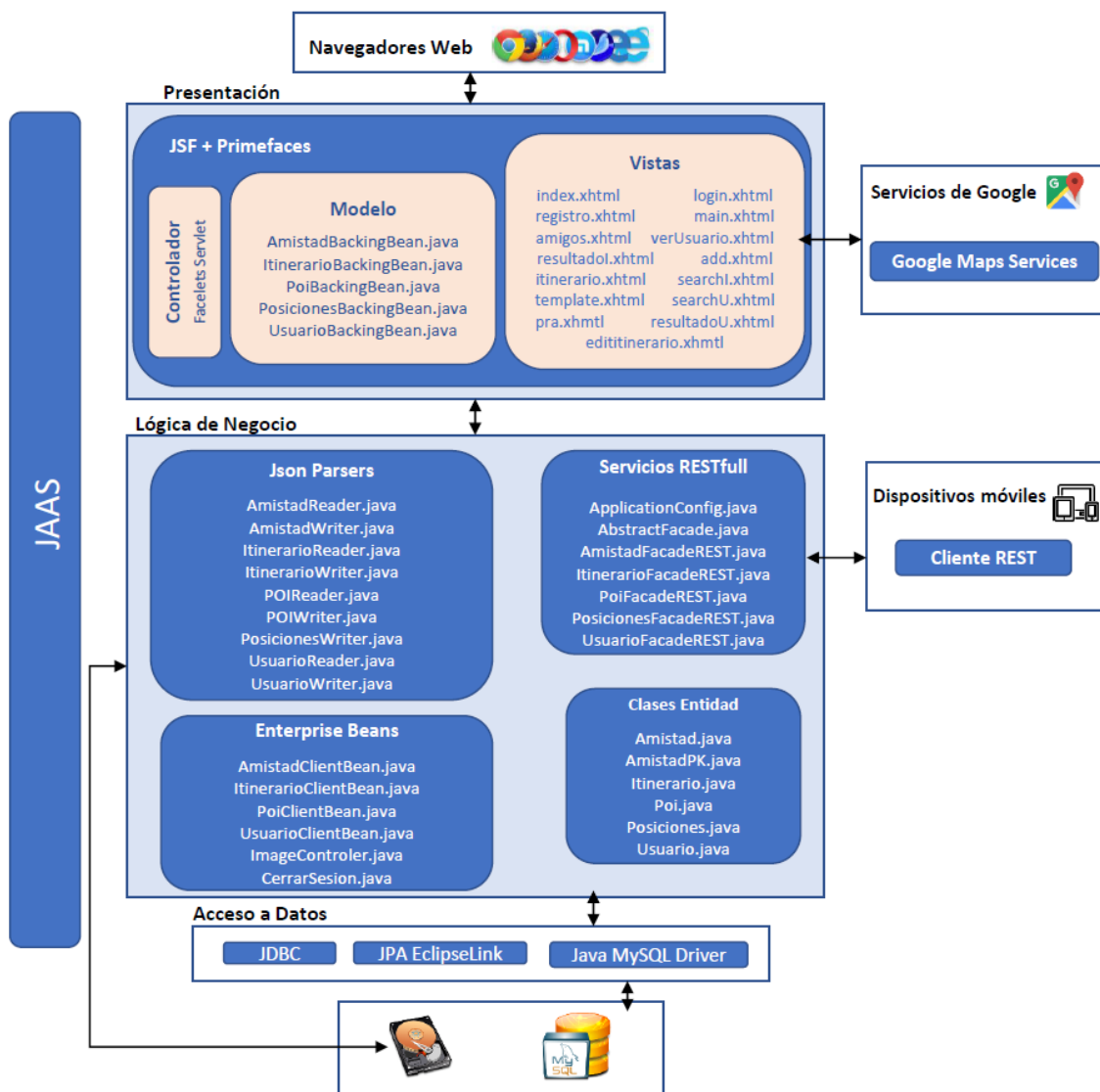


Figura 111. Arquitectura Lógica.

5.2 Arquitectura Física

La arquitectura física representa todos los componentes físicos que sostienen el funcionamiento del sistema, así como la relación en ellos.

El diagrama que representa la arquitectura física del proyecto puede verse a continuación en la *Figura 112*.

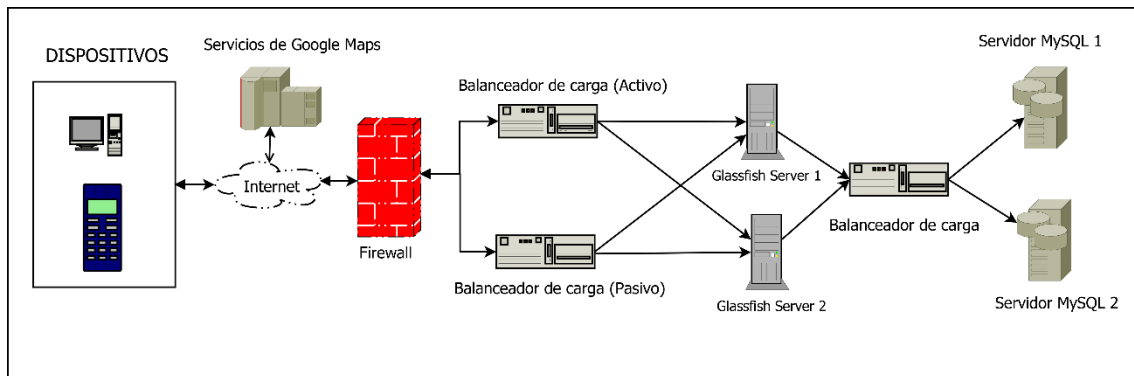


Figura 112. Arquitectura Física.

Los usuarios acceden al sistema GPSSM, desde su aplicación móvil (proyecto independiente), o desde su navegador web, a través de Internet debiendo de atravesar un cortafuegos para mantener la seguridad de la información. Después del Firewall encontramos un balanceador de carga y su réplica (por si falla el activo), que dirigirán el tráfico a uno de los dos servidores de aplicaciones según su carga de trabajo. Cuando los servidores de aplicaciones precisen de realizar una consulta a la base de datos, un tercer balanceador de carga distribuirá la petición entre uno de los dos servidores de bases de datos según su carga.

La arquitectura elegida busca un coste reducido para empezar el proyecto, pero totalmente escalable, y que está preparada para soportar más servidores de réplica si fuera necesario gracias a los balanceadores de carga.

5.3 Modelo Lógico de Datos

En este apartado se procede al diseño de la base de datos de acuerdo con el análisis realizado previamente (*ver punto 4.6*).

El proyecto va a ser implementado bajo una base de datos relacional, sustentando la decisión en los siguientes motivos:

- Permite representar relaciones de la realidad entre entidades.
- Proveen herramientas que garantizan evitar la duplicidad de registros.
- Garantizan la integridad referencial a través de claves externas.

Para representar el diseño de una base de datos relacional, se usa el 'Modelo Relacional'. Este modelo considera la base de datos como una colección de relaciones. De manera simple, una relación representa una tabla que no es más que un conjunto de filas, cada fila es un conjunto de campos. Las claves primarias de las distintas tablas se representan mediante una llave. Las

claves externas vienen indicadas mediante el color rojo en rombo o llave (si son a la vez clave primaria).

El Modelo Relacional del proyecto puede verse a continuación en la *Figura 113*.

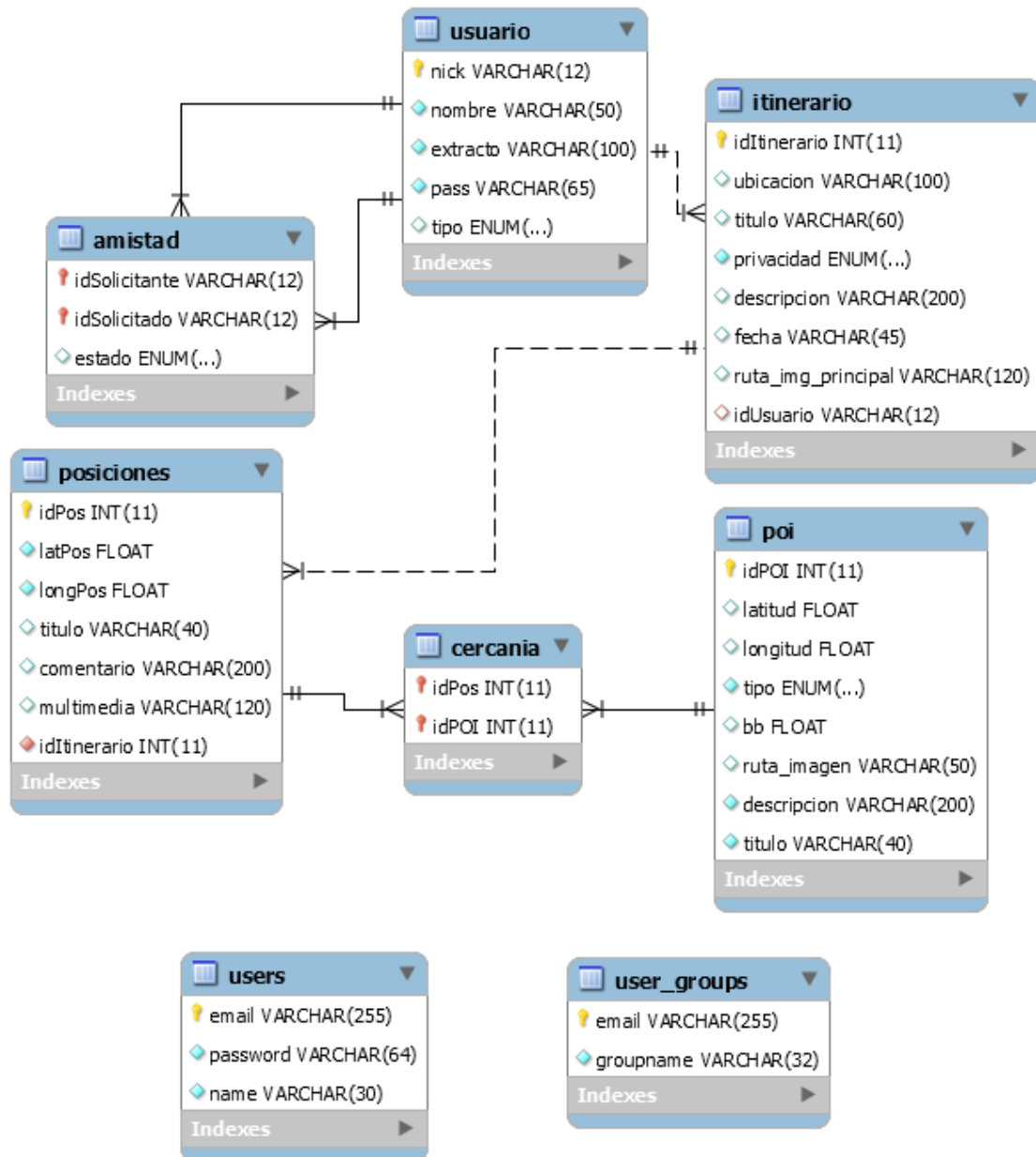


Figura 113. Modelo Relacional.

En el diseño de la base de datos se producen tres decisiones importantes. La primera de ellas, sobre la **‘Relación de especialización de Posiciones’** (ver *punto 5.3.2*). Se decide que las dos entidades relacionadas se junten en una sola tabla, ya que el número de campos por el que se diferencian es muy pequeño (3), y añadir una relación añadiría dificultad técnica al manejar los puntos multimedia, debido a la complicación añadida que tiene realizar consultas relacionales sobre entidades en JavaEE.

La segunda decisión es similar a la anterior, pero sobre la **‘Relación de especialización de POI’** (ver *punto 5.3.2*). La entidad de puntos de interés está concebida como un tipo superior con capacidad de tener diferentes subtipos, como puntos de realidad aumentada (PRA), puntos comerciales, puntos de información, etc. Pero en el alcance de esta versión del proyecto solo vamos a contemplar un tipo de punto de interés, por lo que agruparemos en una sola tabla los atributos de ‘POI’ y ‘PRA’, para simplificar la base de datos.

La tercera decisión, es consecuencia de la implementación del API de JAAS. Debido a la limitación de tiempo en el desarrollo del proyecto, vamos a simplificar su implementación añadiendo a la base de datos las tablas **‘users’** y **‘users_groups’** (ver en parte inferior de la *Figura 113*), que son las tablas genéricas para su implementación. Para mantener integridad de datos con las tablas genéricas de JAAS, con nuestra tabla de usuarios, se implementan una serie de Triggers sobre la tabla **‘usuario’**, que se detallan a continuación en la *Tabla 53*.

Triggers en Plataforma de gestión de Itinerarios GPSSM		
Trigger 1	<i>Tabla</i>	usuario
	<i>Disparador</i>	Before insert
	<i>Descripción</i>	Antes de insertar en la tabla ‘usuario’ los datos de registro, el trigger se dispara para rellenar la información de la tabla ‘users’ con el Nick, contraseña y nombre, así como Nick y rol en la tabla ‘users_groups’.
Trigger2	<i>Tabla</i>	usuario
	<i>Disparador</i>	After update
	<i>Descripción</i>	Después de actualizar en la tabla ‘usuario’, se realizan las pertinentes modificaciones en las tablas genéricas de JAAS antes mencionadas.
Trigger3	<i>Tabla</i>	usuario
	<i>Disparador</i>	After delete
	<i>Descripción</i>	Cuando se borra un registro de la tabla ‘usuario’, se borran los registros correspondientes registros relacionados en las tablas genéricas de JAAS antes mencionadas.

Tabla 53. Triggers de la base de datos.

El código de creación de la base de datos (DDL), con la implementación de los Triggers se incluye en el CD del proyecto. Ver **ANEXO IV** para ver detalle de la estructura de este.

5.4 Diagrama de Clases

El diagrama de clases incluye todas las clases que es preciso implementar para el desarrollo del proyecto. Debido al gran número de clases que posee el proyecto, el diagrama va a presentarse en distintas partes para facilitar su legibilidad.

La división va a realizarse en varias categorías: Clases Entidad, Servicios RESTfull, Modelo de JSF, Enterprise Beans, JsonParsers, y Clases de seguridad. Estas categorías se corresponden en su mayoría a los paquetes definidos en la arquitectura del sistema (ver *punto 6.1*).

Clases Entidad

Clases que forman las clases Entidad que abstraen de la base de datos. Ver *Figuras 114, 115, 116 y 117*. La entidad AmistadPK se genera para representar la clave primaria compuesta entre dos entidades Usuario.

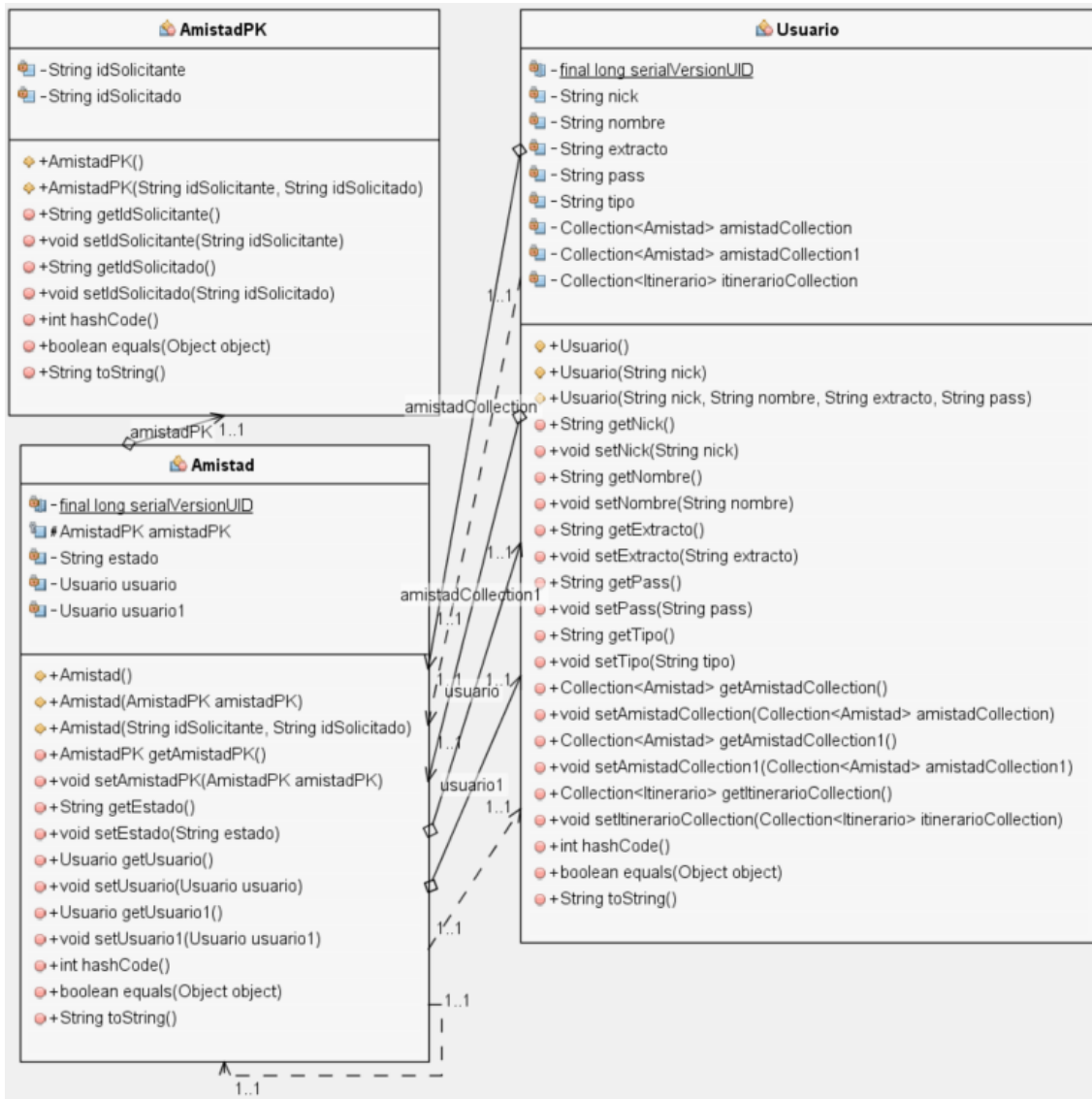


Figura 114. Entidades AmistadPK, Amistad y Usuario.

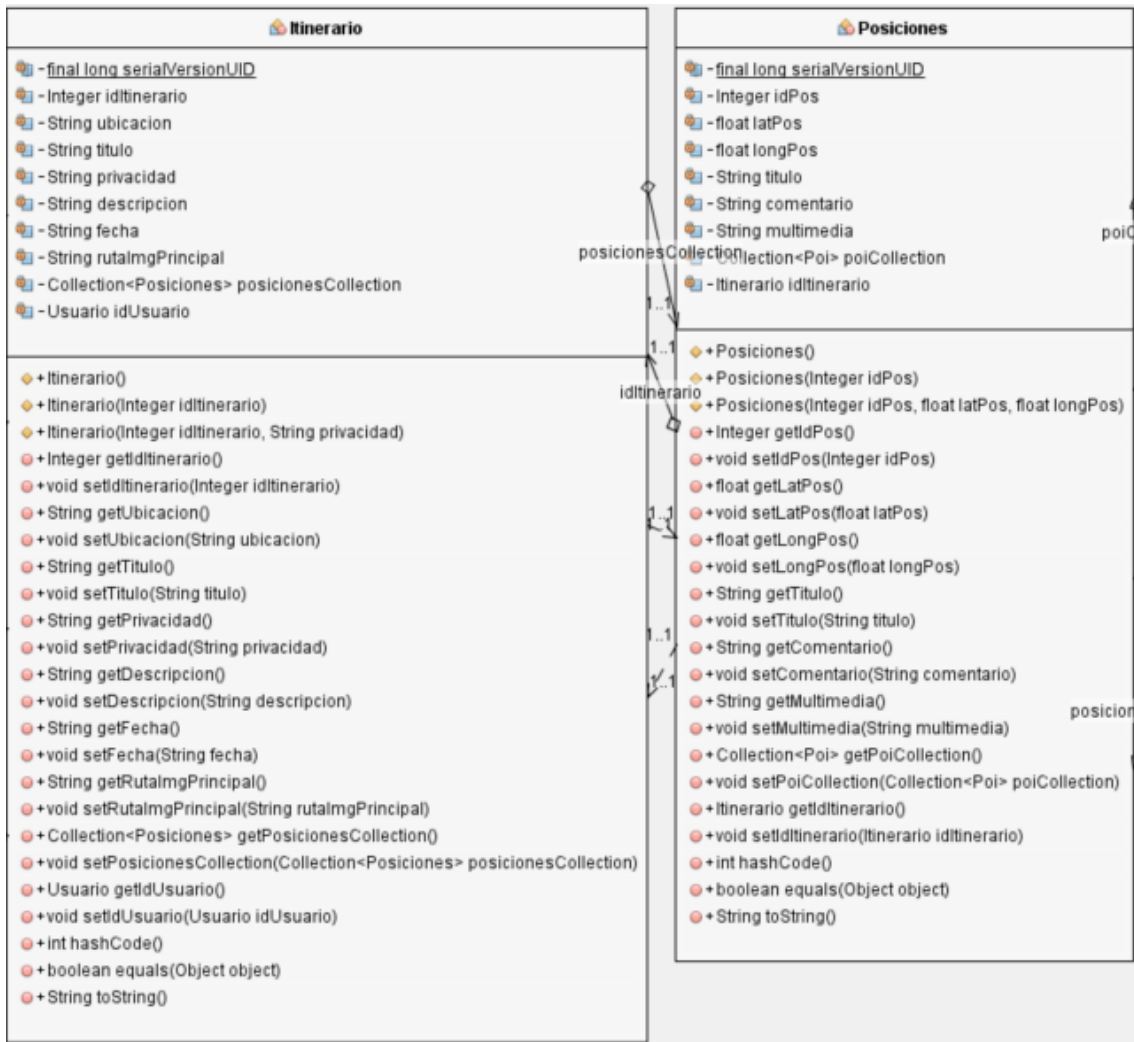


Figura 115. Entidades Itinerario y Posiciones.

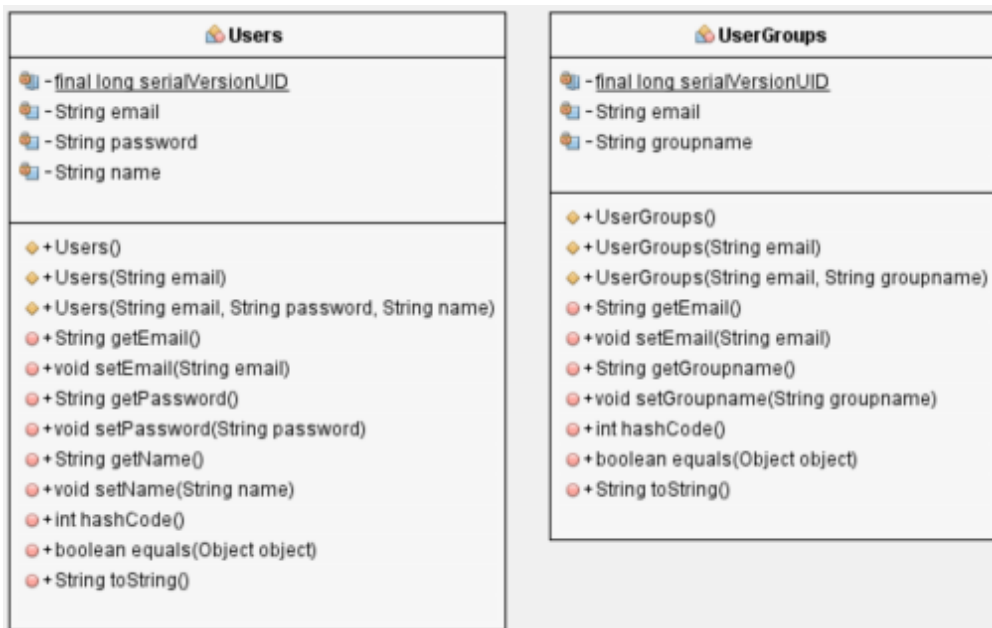


Figura 116. Entidades Users y UserGroups

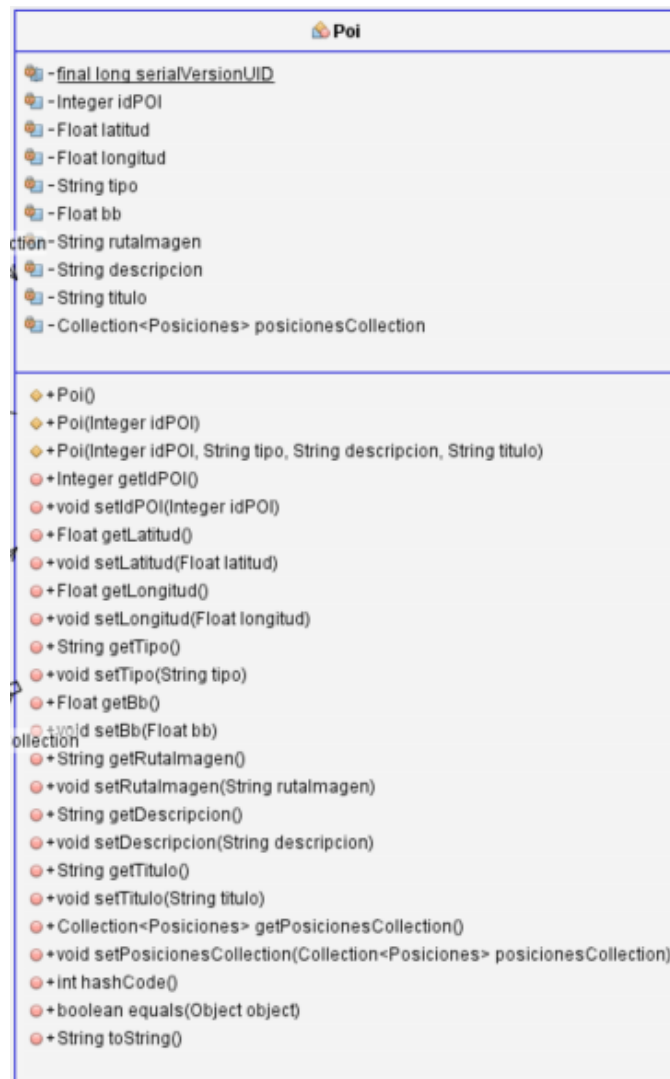


Figura 117. Entidad POI.

Servicios RESTfull

Clases que definen y establecen los servicios REST del sistema. Estos trabajan basándose en las clases entidad expuestas anteriormente.

Ver Figuras 118, 119 y 120.

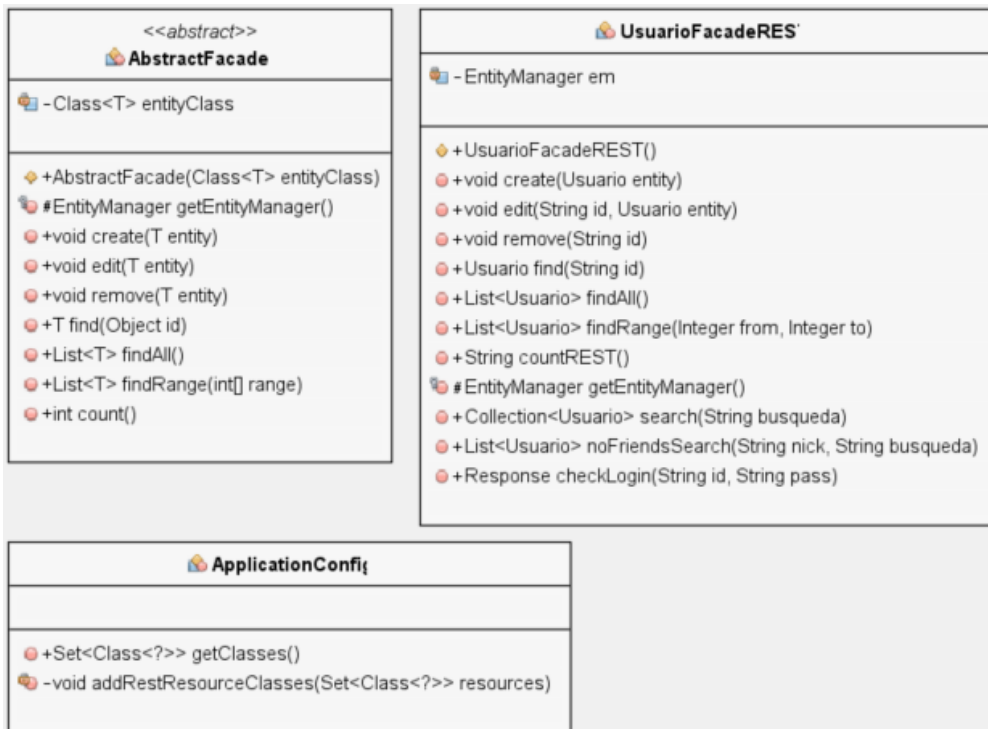


Figura 118. Clases AbstractFacade, ApplicationConfig y UsuarioFacadeREST.

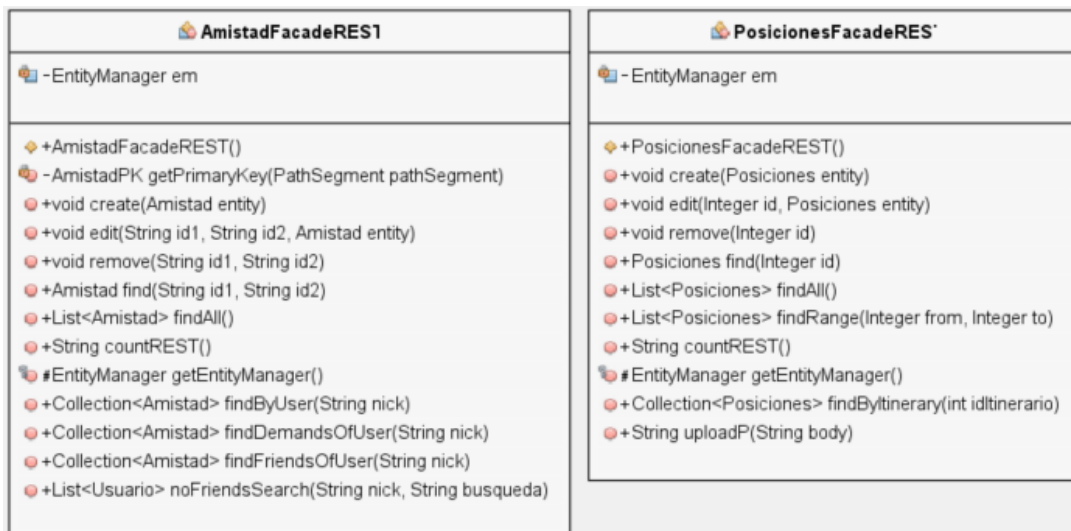


Figura 119. Clases AmistadFacadeREST y PosicionesFacadeREST.

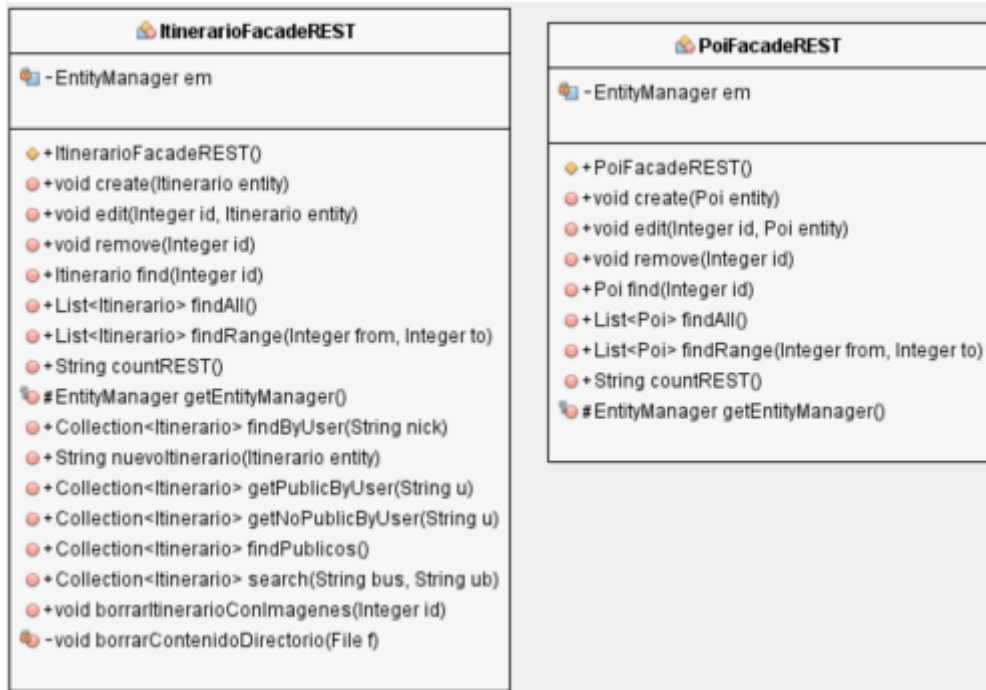


Figura 120. Clases ItinerarioFacadeREST y PoiFacadeREST.

Modelo de JSF

Engloba las clases que guardan el modelo de datos para Java Server Faces. Se denominan 'BackingBeans'. Ver Figuras 121 y 122.

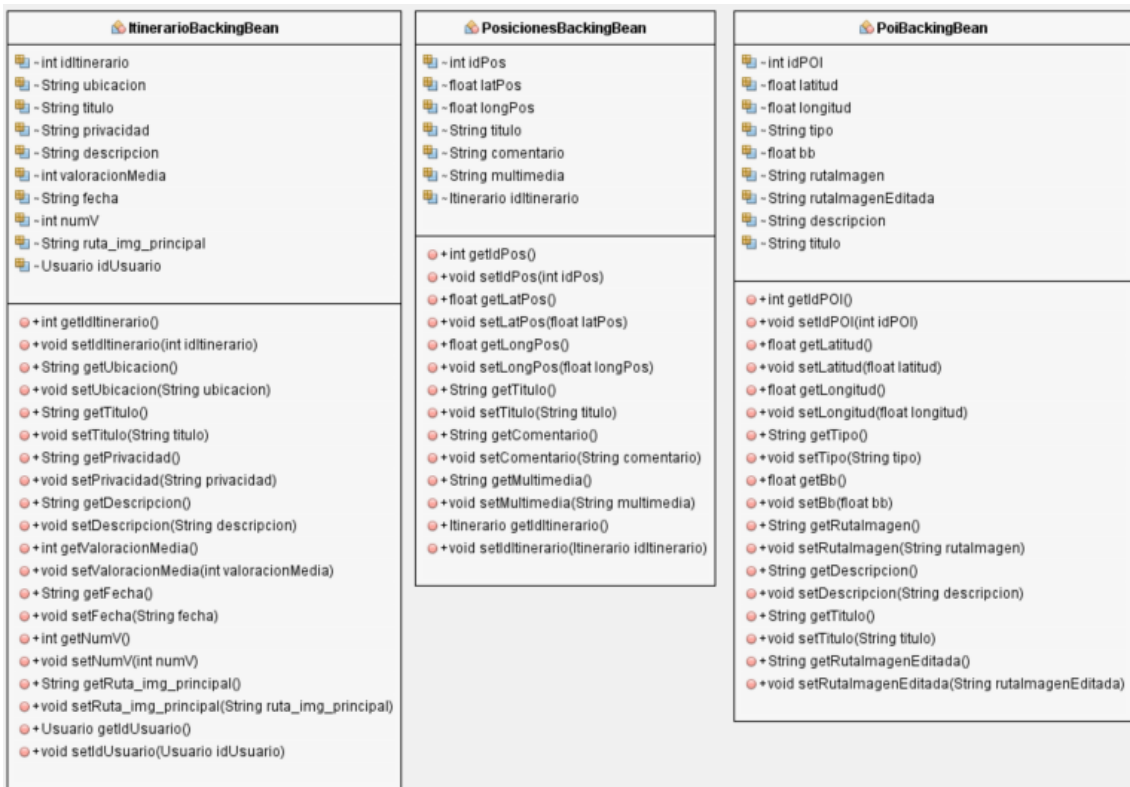


Figura 121. Clases ItinerarioBackingBean, PosicionesBackingBean y PoiBackingBean.

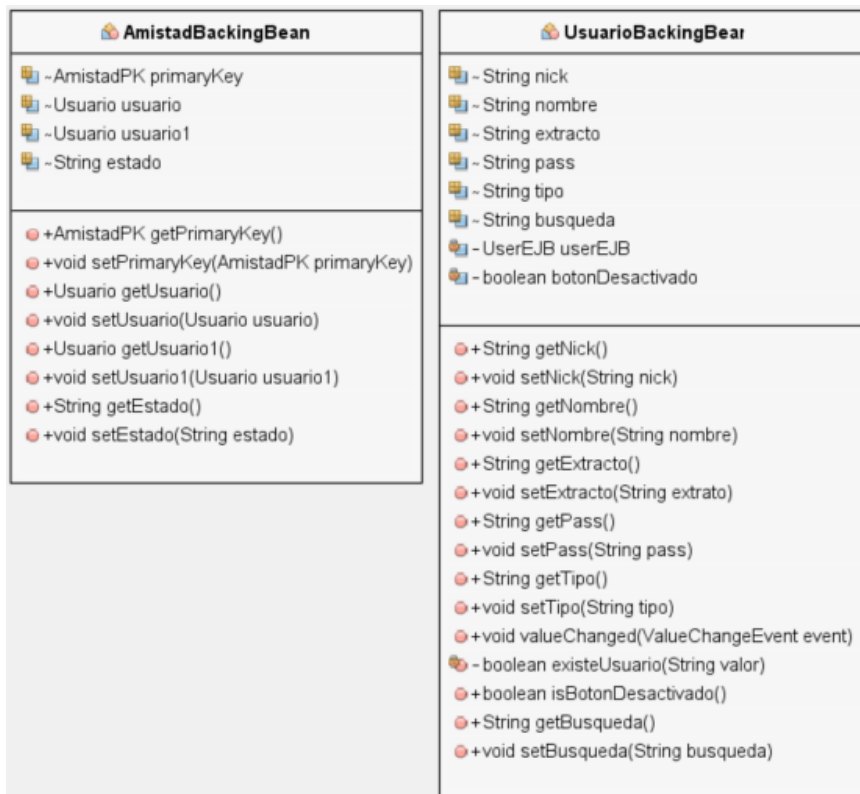


Figura 122. Clases AmistadBackingBean y UsuarioBackingBean.

Enterprise Beans

Engloba las clases que controlan la lógica de negocio. Ver Figuras 123, 124 y 125.

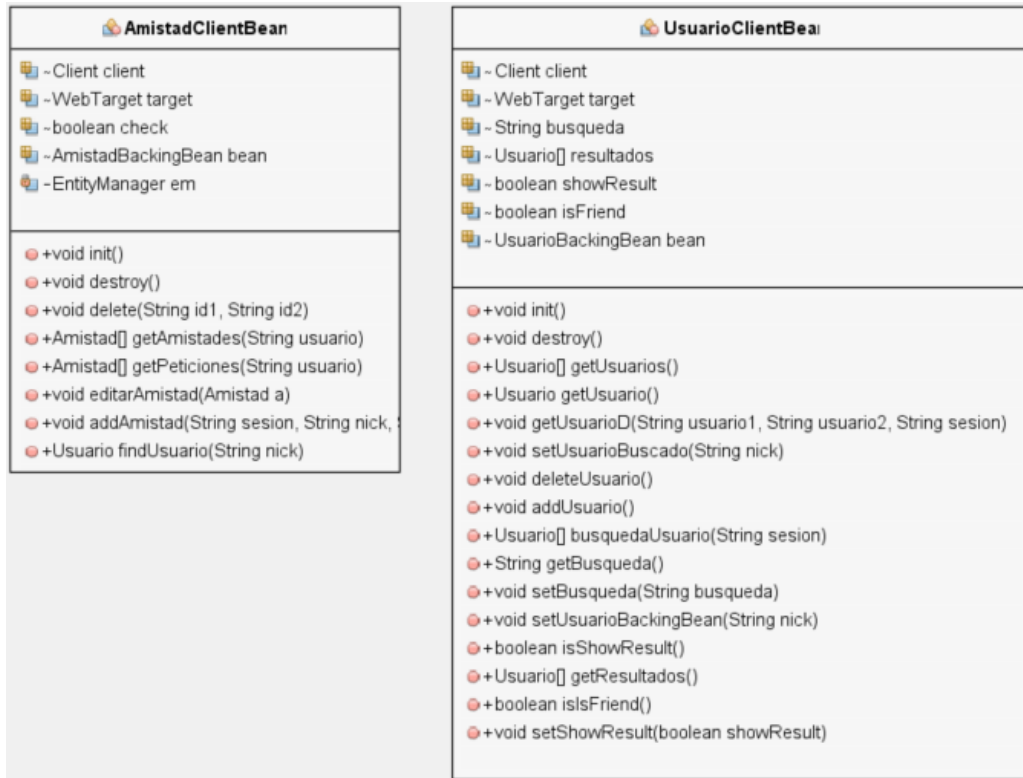


Figura 123. AmistadClientBean y UsuarioClientBean.



Figura 124. Clases PoiClientBean y ImageControler.

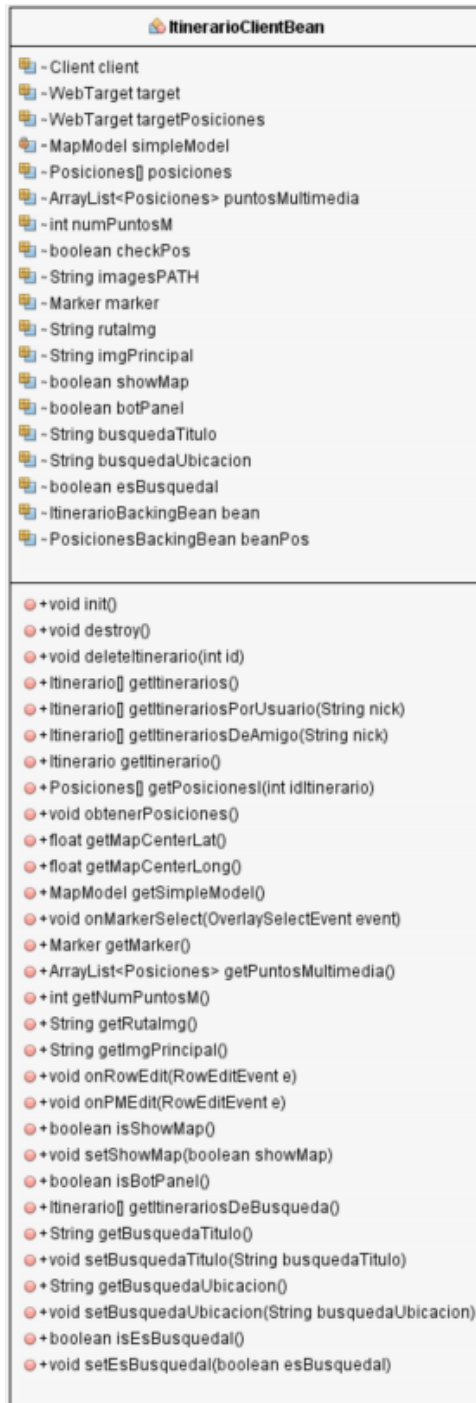


Figura 125. Clase ItinerarioClientBean.

JsonParsers

Engloba las clases encargadas de traducir entidades en JSON, y construir JSON para manejar las entidades.

Ver Figuras 126 y 127.

AmistadReader
<ul style="list-style-type: none"> + boolean isReadable(Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType) + Amistad readFrom(Class<Amistad> type, Type genericType, Annotation[] annotations, MediaType mediaType, MultivaluedMap<String, String> httpHeaders, InputStream entityStream)

AmistadWriter
<ul style="list-style-type: none"> + boolean isWritable(Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType) + long getSize(Amistad t, Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType) + void writeTo(Amistad a, Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType, MultivaluedMap<String, Object> httpHeaders, OutputStream entityStream)

ItinerarioReader
<ul style="list-style-type: none"> + boolean isReadable(Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType) + Itinerario readFrom(Class<Itinerario> type, Type genericType, Annotation[] annotations, MediaType mediaType, MultivaluedMap<String, String> httpHeaders, InputStream entityStream)

ItinerarioWriter
<ul style="list-style-type: none"> + boolean isWritable(Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType) + long getSize(Itinerario t, Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType) + void writeTo(Itinerario i, Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType, MultivaluedMap<String, Object> httpHeaders, OutputStream entityStream)

Figura 126. Clases AmistadReader, AmistadWriter, ItinerarioReader e Itinerario Writer.

PosicionesWriter
<ul style="list-style-type: none"> + boolean isWritable(Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType) + long getSize(Posiciones t, Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType) + void writeTo(Posiciones p, Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType, MultivaluedMap<String, Object> httpHeaders, OutputStream entityStream)

POWriter
<ul style="list-style-type: none"> + boolean isWritable(Class<?> type, Type type1, Annotation[] antns, MediaType mt) + long getSize(Poi t, Class<?> type, Type type1, Annotation[] antns, MediaType mt) + void writeTo(Poi p, Class<?> type, Type type1, Annotation[] antns, MediaType mt, MultivaluedMap<String, Object> mm, OutputStream out)

POIReader
<ul style="list-style-type: none"> + boolean isReadable(Class<?> type, Type type1, Annotation[] antns, MediaType mt) + Poi readFrom(Class<Poi> type, Type type1, Annotation[] antns, MediaType mt, MultivaluedMap<String, String> mm, InputStream in)

UsuarioWriter
<ul style="list-style-type: none"> + boolean isWritable(Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType) + long getSize(Usuario t, Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType) + void writeTo(Usuario u, Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType, MultivaluedMap<String, Object> httpHeaders, OutputStream entityStream)

UsuarioReader
<ul style="list-style-type: none"> + boolean isReadable(Class<?> type, Type genericType, Annotation[] annotations, MediaType mediaType) + Usuario readFrom(Class<Usuario> type, Type genericType, Annotation[] annotations, MediaType mediaType, MultivaluedMap<String, String> httpHeaders, InputStream entityStream)

Figura 127. Clases PosicionesReader, PoiWriter, PoiReader, Usuario Writer. Y UsuarioReader.

Clases de seguridad

Son clases que realizan necesarias para el funcionamiento de JAAS, como el cifrado de contraseñas, búsqueda de usuario para autenticación e inicio y cierre de sesión.

Ver *Figura 128*.

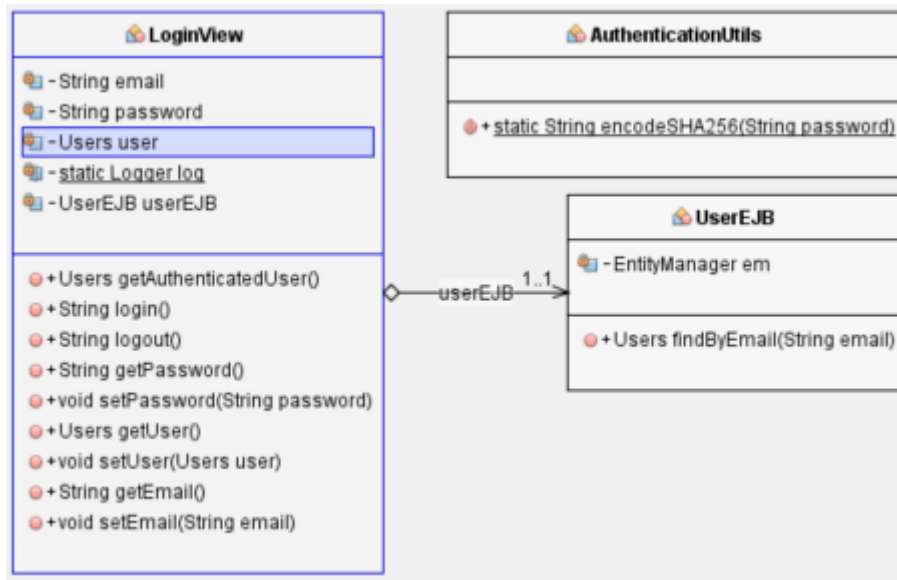


Figura 128. LoginView, AuthenticationUtils y UserEJB..

El diagrama completo puede consultarse de forma completa en el CD del proyecto. Ver **ANEXO IV** para conocer el detalle de la estructura de este.

5.5 Diagramas de secuencia

Los diagramas de secuencia describen las interacciones entre los componentes del sistema a lo largo del tiempo para realizar una determinada operación. Son usados para clarificar la interpretación de los casos de uso.

Es importante destacar que en la mayoría de diagramas hay dos flujos a seguir, dependiendo del actor que este ejecutando el caso de uso, PSM (Plataforma Software Móvil) o un usuario que accede desde portal web.

Los diagramas de secuencia realizados durante la etapa de diseño de este proyecto se detallan a continuación.

Registro de Itinerario

Este diagrama, que se corresponde con la *Figura 129*, está relacionado con los casos de uso PM1 y PM2 del actor Plataforma Móvil, que especifican la subida de un nuevo itinerario desde un dispositivo móvil.

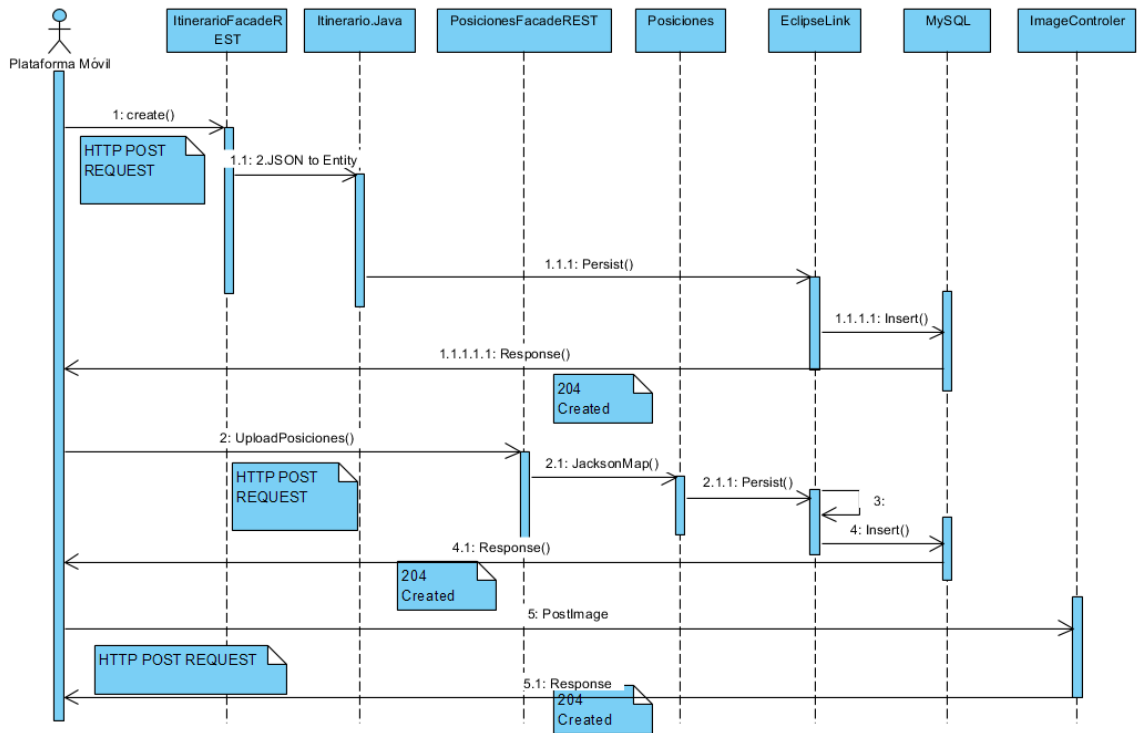


Figura 129. Diagrama de secuencia sobre registro de itinerario.

Borrado de Itinerario

Este diagrama, que se corresponde con la *Figura 130*, está relacionado con el caso de uso UGR4 del actor Usuario General Registrado, que especifica el borrado de un itinerario.

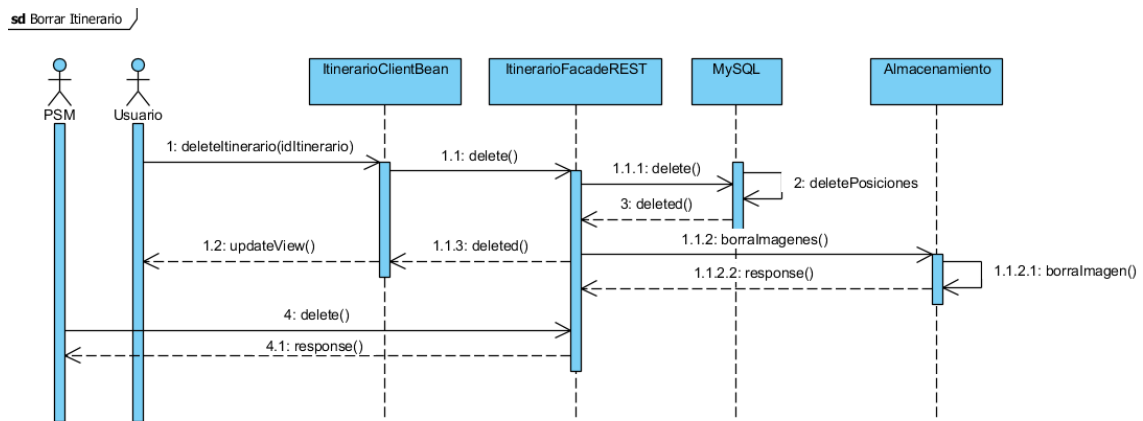


Figura 130. Diagrama de secuencia sobre eliminar itinerario.

Editar itinerario

Este diagrama, que se corresponde con la *Figura 131*, está relacionado con el caso de uso UGR2, que describe la modificación de los atributos un itinerario.

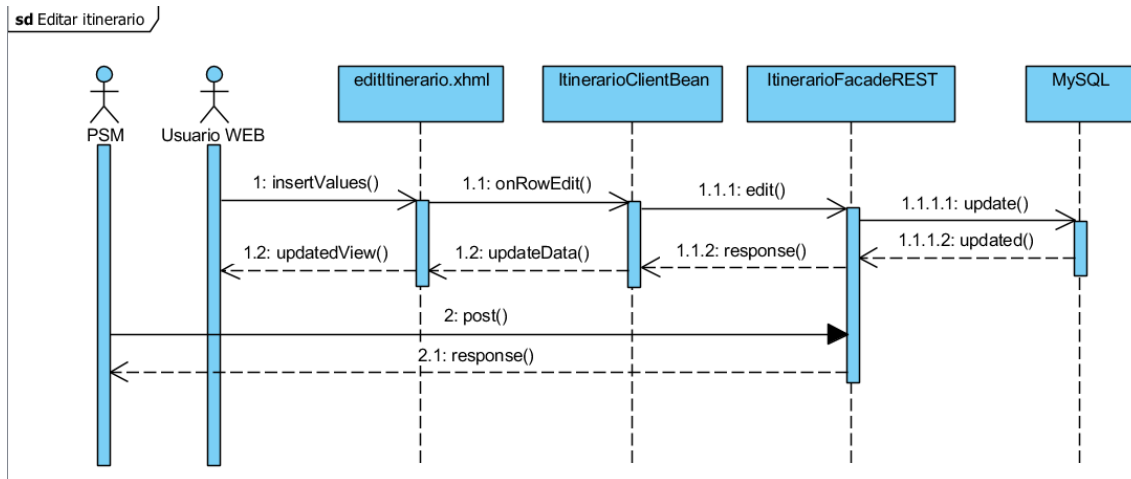


Figura 131. Diagrama de secuencia sobre editar itinerario.

Este diagrama de secuencia sería equivalente, debido a su gran similitud, en los casos de uso UGR3, GS3 y PM4, ya que son operaciones de actualización de registros. En el caso de GS3, que se corresponde con la modificación de puntos de interés solo discurriría el flujo del Usuario web. En el caso de uso PM4, que se corresponde con la modificación de una cuenta de usuario solo discurriría el flujo de PSM. EN UGR3, que se corresponde con la edición de puntos multimedia, se puede dar ambos flujos.

Gestionar Itinerarios

Este diagrama, que se corresponde con la Figura 132, está relacionado con el caso de uso UGR1, que describe la consulta del listado de itinerarios para su gestión.

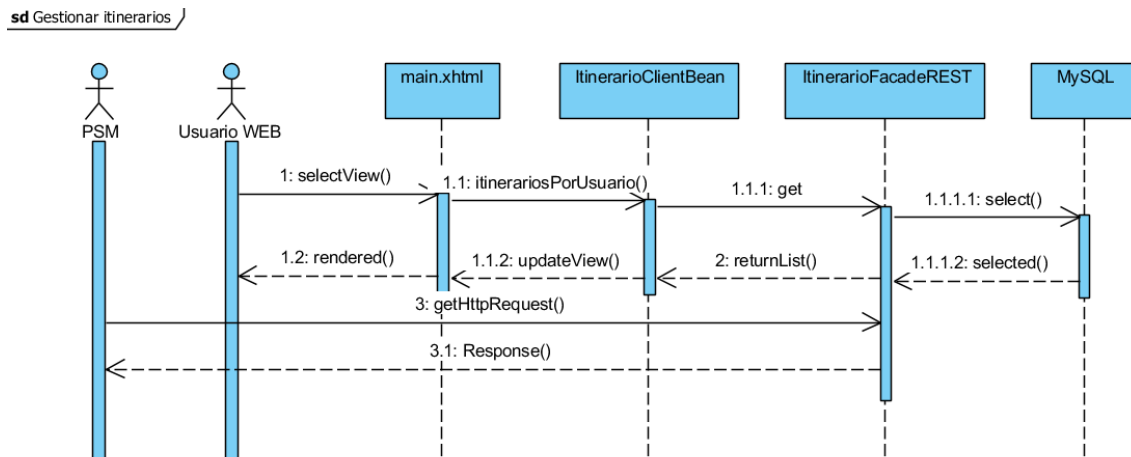


Figura 132. Diagrama de secuencia sobre gestionar itinerarios.

Este diagrama de secuencia sería equivalente en los casos de uso GS1, URG8, URG9 y PM7 ya que son operaciones de consulta de listados. El caso de uso GS1, que se corresponde con la Gestión de Puntos de Interés, solo discurriría por el flujo del Usuario Web. El caso de uso PM6, que se corresponde con la obtención de puntos de interés, solo discurriría por el flujo del actor PSM. El resto de casos de uso discurrirían por ambos flujos.

Buscar usuario

Este diagrama, que se corresponde con la *Figura 133*, está relacionado con el caso de uso UGR5, que describe la interacción del sistema al buscar un usuario.

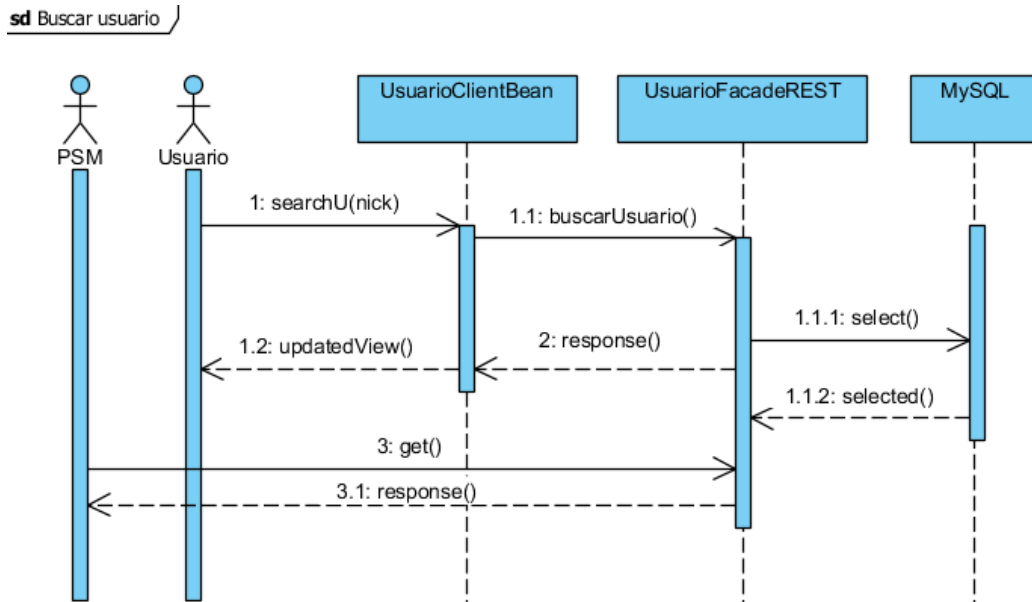


Figura 133. Diagrama de secuencia sobre buscar itinerario.

Este diagrama de secuencia sería equivalente en el caso de uso UGR13, que trata sobre la búsqueda de itinerarios, ya que se trata de una operación de búsqueda y guarda gran similitud.

Eliminar amistad

Este diagrama, que se corresponde con la *Figura 134*, está relacionado con el caso de uso UG12, que describe la interacción del sistema al eliminar una relación de amistad.

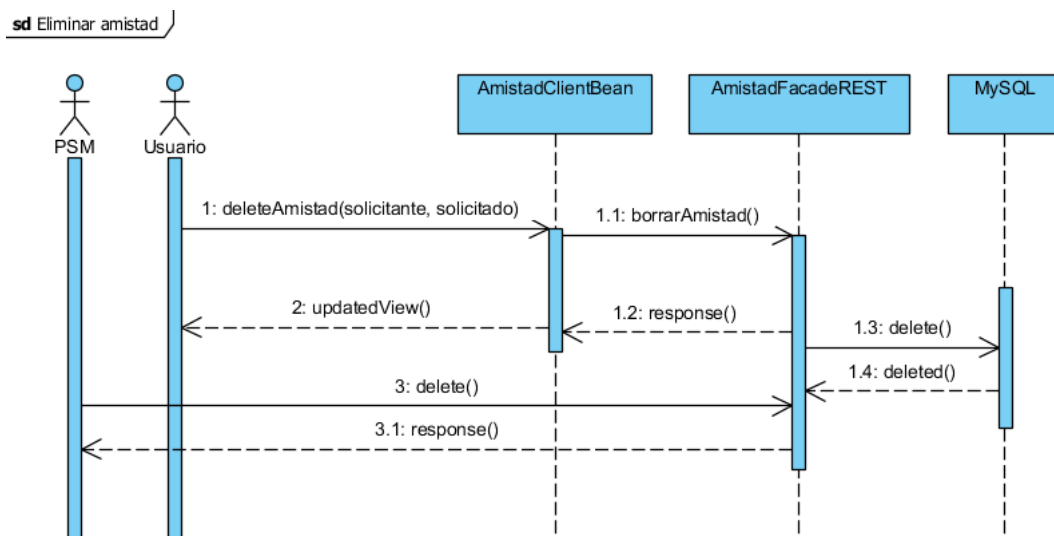


Figura 134. Diagrama de secuencia de eliminar una amistad.

Este diagrama de secuencia sería equivalente en los casos de uso GS4 y PM5, que tratan sobre la eliminación de puntos de interés y eliminación de cuenta de usuario respectivamente, ya que se trata de operaciones similares de eliminación. En caso de GS4 solo se discurriría por el flujo del Usuario Web, en cambio en PM5 solo discurriríamos por el flujo del actor PSM.

Resolver solicitud

Este diagrama, que se corresponde con la *Figura 135*, está relacionado con el caso de uso UG10, que describe la interacción del sistema al resolver una petición de amistad.

sd Resolver solicitud

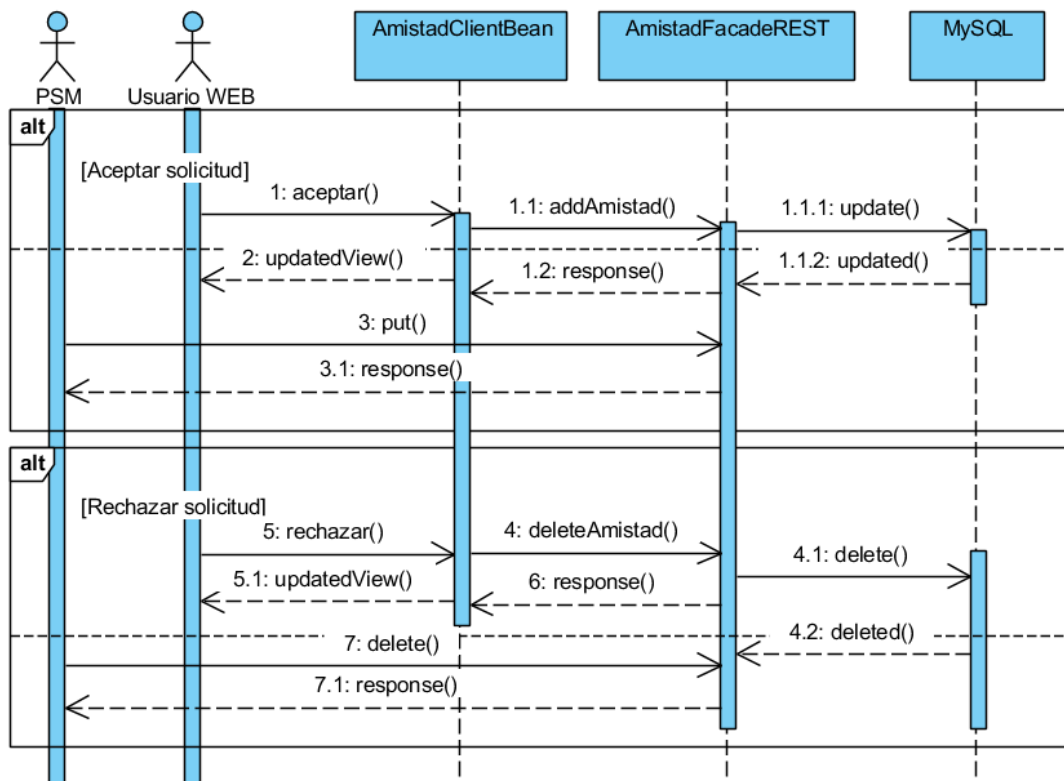


Figura 135. Diagrama de secuencia de resolver solicitud.

Este diagrama en concreto presenta dos secuencias alternativas para cada actor. Esto se debe a que cuando aceptamos una petición de amistad, el estado de la relación pasa de 'PENDIENTE' (como se establece al generar una solicitud) a 'ACTIVA', lo que requiere una operación de actualización en la base de datos. En cambio, cuando rechazamos la petición se produce un borrado en la base de datos de esta relación en estado 'PENDIENTE', permitiendo así que se pueda formular petición de nuevo en el futuro.

Login

Este diagrama, que se corresponde con la *Figura 136*, está relacionado con el caso de uso UG12.

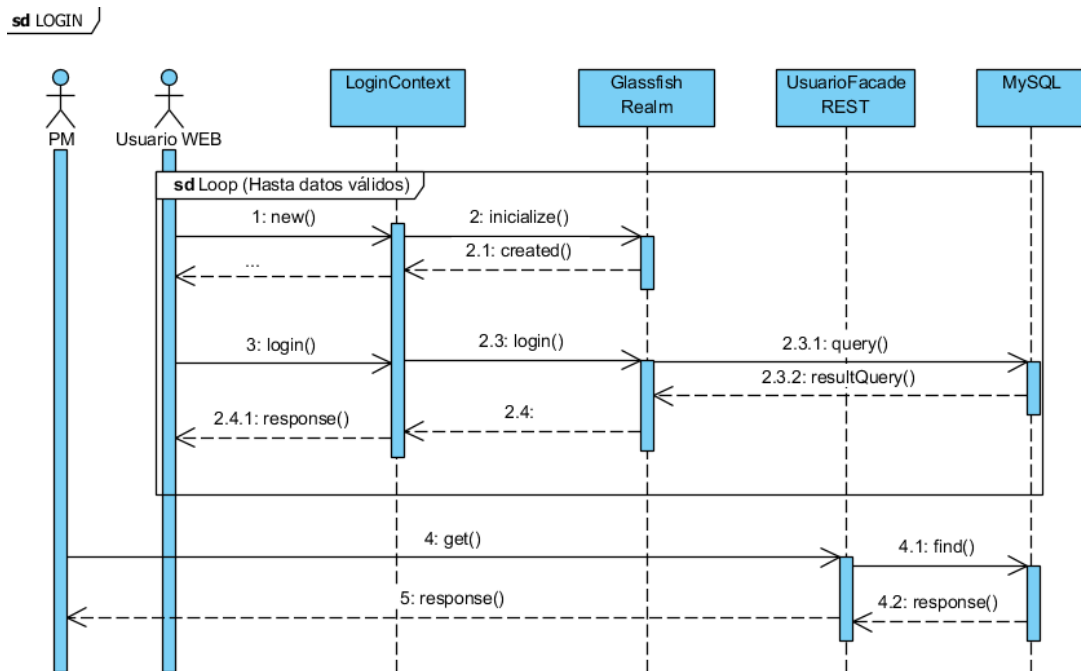


Figura 136. Diagrama de secuencia sobre login.

Registro de usuario

Este diagrama, que se corresponde con la *Figura 137*, está relacionado con el caso de uso UG11, que trata sobre el registro de un usuario en el sistema.

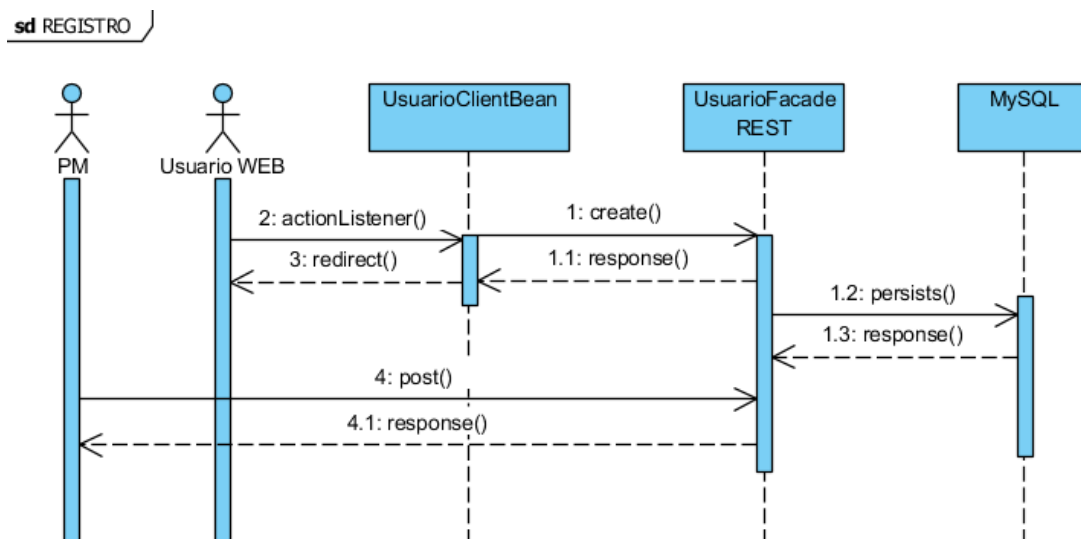


Figura 137. Diagrama de secuencia sobre registro.

Este diagrama de secuencia sería equivalente en los casos de uso GS2 y URG7, ya que son operaciones de consulta de listados y el diagrama sería de gran similitud. En el caso de GS2, que se corresponde con la creación de puntos de interés solo discurriría el flujo del Usuario web. En el caso de uso UGR7, que se corresponde a la solicitud de amistad, se pueden dar los flujos de ambos actores.

Logout

Este diagrama, que se corresponde con la *Figura 138*, está relacionado con el caso de uso UWR1, que trata del cierre de la sesión web por parte de un usuario.

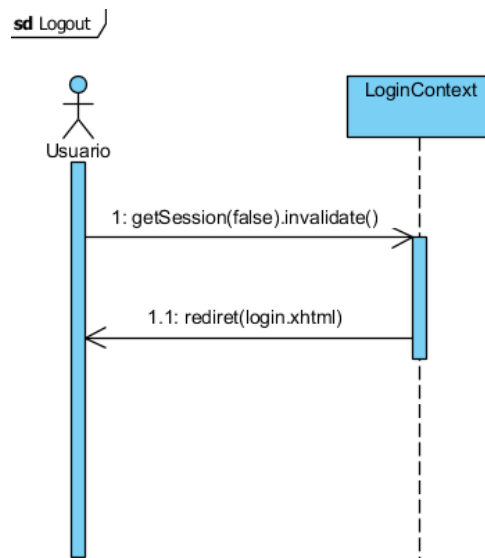


Figura 138. Diagrama de secuencia de cierre de sesión.

5.6 Diseño de interfaces

En este apartado se trata el diseño de las interfaces de usuario de la plataforma web del proyecto GPSSM.

Todas las interfaces web de este sistema van a seguir un esquema de cabecera, menú lateral izquierdo y contenido, como puede verse a continuación en la *Figura 139*.

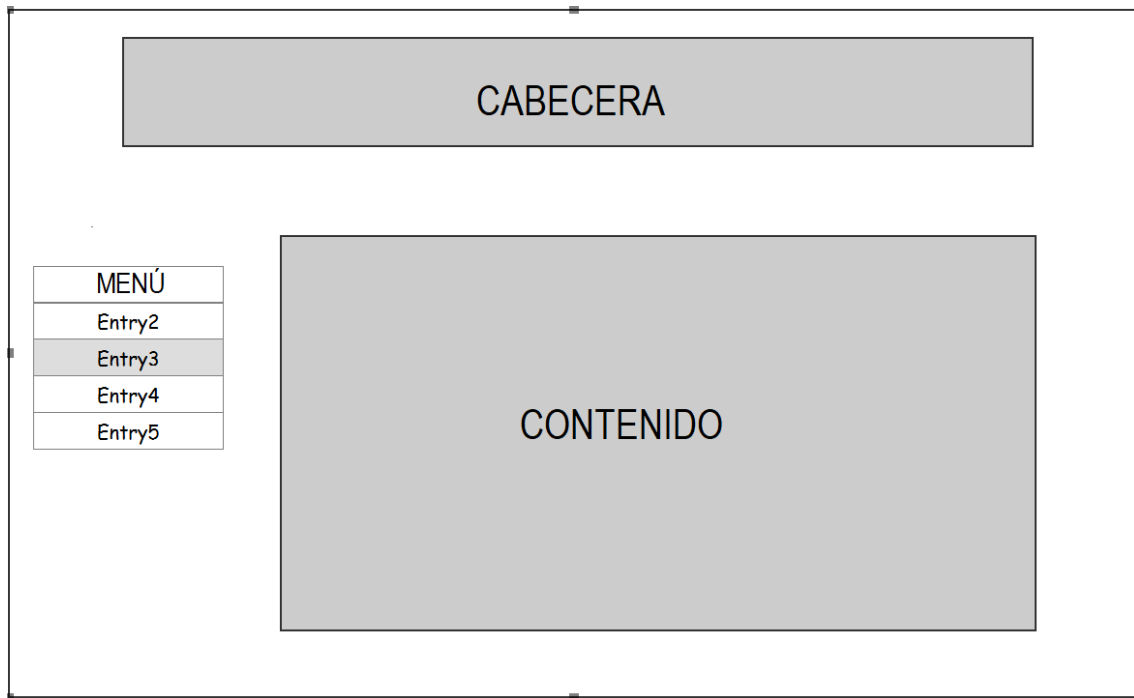


Figura 139. Interfaz general de la aplicación.

Los menús dependerán de si el tipo de usuario es 'Normal', o 'Gestor', así como de si el usuario está identificado en el sistema. Los distintos menús para cada tipo de usuario pueden verse en la Figura 140.

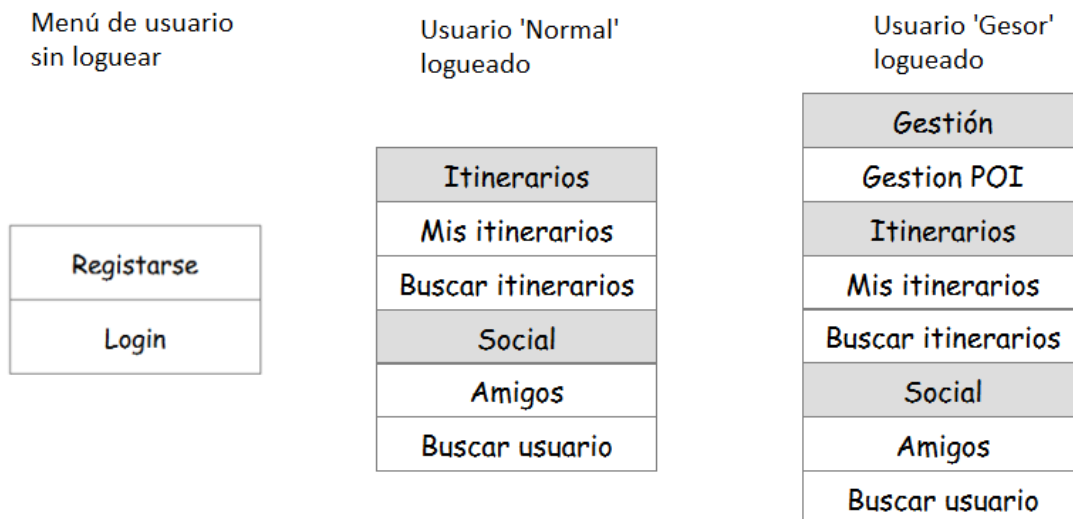


Figura 140. Menús de la aplicación.

Se pretende diseñar una interfaz amigable y sencilla, que haga que el proceso de familiarización del usuario con la aplicación y su uso sea rápido. Para ello se han diseñado una serie de interfaces para su uso en las principales funcionalidades de la aplicación web que se incluyen a continuación.

Vista	Login
Descripción	Pantalla que permite al usuario iniciar sesión.
Activación	<ul style="list-style-type: none"> • El usuario pulsa sobre la opción de menú 'Login'. • El usuario accede a una página restringida después de caducar su sesión. • El usuario ha cerrado sesión.
Boceto	
Eventos	<ul style="list-style-type: none"> • Iniciar sesión en el sistema. • Acceder a panel de usuario.

Tabla 54. Diseño de interfaz de Login.

Vista	Registro
Descripción	Pantalla que permite a un usuario invitado registrarse en sistema.
Activación	<ul style="list-style-type: none"> • El usuario pulsa sobre la opción de menú 'Registro'.
Boceto	
Eventos	<ul style="list-style-type: none"> • Registrarse en el sistema.

Tabla 55. Diseño de interfaz de Registro.

Vista	Panel de usuario																					
Descripción	Es la ventana principal de un usuario registrado, donde ve sus itinerarios y peticiones de amistad. Se corresponde con el ítem de menú 'Itinerarios'.																					
Activación	<ul style="list-style-type: none"> El usuario se identifica en el sistema. El usuario pulsa el ítem de menú 'Mis itinerarios'. 																					
Boceto	<p>Sistema GPSSM</p> <p>Peticiones de amistad</p> <table border="1"> <thead> <tr> <th>Solicitante</th> <th>Solicitado</th> <th>Resolver</th> </tr> </thead> <tbody> <tr> <td>lorem</td> <td>ipsum</td> <td><input checked="" type="checkbox"/> <input type="checkbox"/></td> </tr> <tr> <td>first</td> <td>second</td> <td><input checked="" type="checkbox"/> <input type="checkbox"/></td> </tr> </tbody> </table> <p>Tus itinerarios</p> <table border="1"> <thead> <tr> <th>Titulo</th> <th>Ubicación</th> <th>Fecha</th> <th>Ver</th> </tr> </thead> <tbody> <tr> <td>lorem</td> <td>ipsum</td> <td>dolor</td> <td><input type="button" value="▶"/></td> </tr> <tr> <td>first</td> <td>second</td> <td>third</td> <td><input type="button" value="▶"/></td> </tr> </tbody> </table>	Solicitante	Solicitado	Resolver	lorem	ipsum	<input checked="" type="checkbox"/> <input type="checkbox"/>	first	second	<input checked="" type="checkbox"/> <input type="checkbox"/>	Titulo	Ubicación	Fecha	Ver	lorem	ipsum	dolor	<input type="button" value="▶"/>	first	second	third	<input type="button" value="▶"/>
Solicitante	Solicitado	Resolver																				
lorem	ipsum	<input checked="" type="checkbox"/> <input type="checkbox"/>																				
first	second	<input checked="" type="checkbox"/> <input type="checkbox"/>																				
Titulo	Ubicación	Fecha	Ver																			
lorem	ipsum	dolor	<input type="button" value="▶"/>																			
first	second	third	<input type="button" value="▶"/>																			
Eventos	<ul style="list-style-type: none"> Aceptar petición de amistad. Rechazar petición de amistad. Solicitar ver detalle de un itinerario. 																					

Tabla 56. Diseño de interfaz Panel de usuario.

Vista	Ver itinerario
Descripción	Es la ventana en la que el usuario puede ver toda la información de un itinerario, su recorrido, puntos multimedia y fotos.
Activación	<ul style="list-style-type: none"> Un usuario selecciona un itinerario para consultar.
Boceto	<p>Sistema GPSSM</p> <p>Ver Itinerario Ejemplo</p> <p>Datos de Itinerario Fotos</p> <p>Recorrido</p>
Eventos	<ul style="list-style-type: none"> Editar itinerario. Eliminar itinerario.

Tabla 57. Diseño de interfaz Ver itinerario.

Vista	Buscar itinerario
Descripción	Esta vista permite a los usuarios buscar itinerarios por título, ubicación o ambos parámetros.
Activación	<ul style="list-style-type: none"> El usuario pulsa el ítem de menú 'Buscar itinerario'.
Boceto	
Eventos	<ul style="list-style-type: none"> Realizar búsqueda por título. Realizar búsqueda por ubicación. Realizar búsqueda por ambos parámetros.

Tabla 58. Diseño interfaz Buscar usuario.

Vista	Amigos
Descripción	Permite al usuario visualizar la lista de amigos que tiene en la plataforma.
Activación	El usuario pulsa sobre la opción de menú 'Amigos'.
Boceto	
Eventos	<ul style="list-style-type: none"> Eliminar amistad Ver perfil de usuario amigo

Tabla 59. Diseño interfaz Amigos.

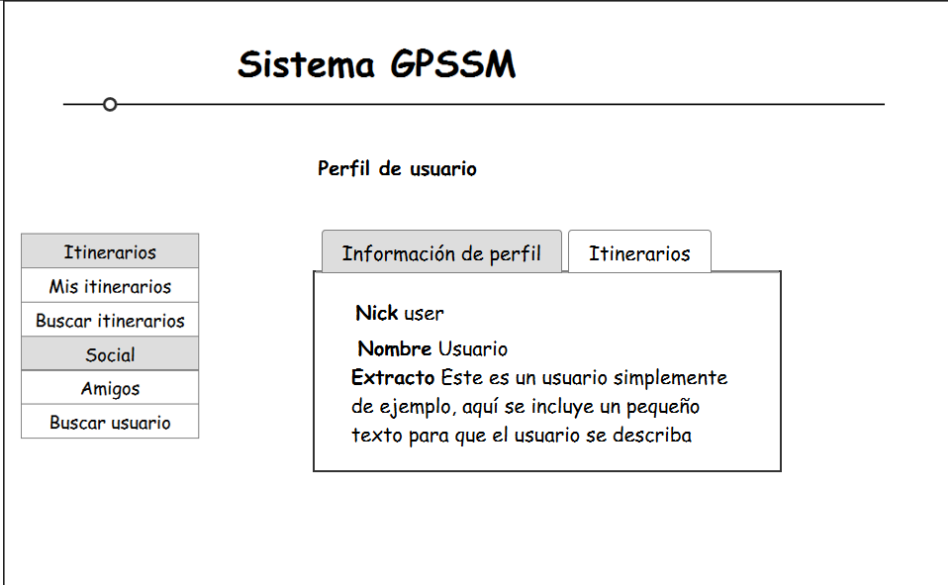
Vista	Ver perfil
Descripción	Vista que permite consultar los detalles de un usuario y sus itinerarios 'Públicos'. Si se accede desde la lista de amistades podremos ver además los itinerarios con privacidad 'Amistad'.
Activación	<ul style="list-style-type: none"> • Selección de ver perfil desde 'Amigos'. • Selección de perfil desde los resultados de búsqueda.
Boceto	
Eventos	<ul style="list-style-type: none"> • Ver pestaña de información. • Ver pestaña de itinerarios. • Ver itinerario seleccionado.

Tabla 60. Diseño interfaz Ver perfil.

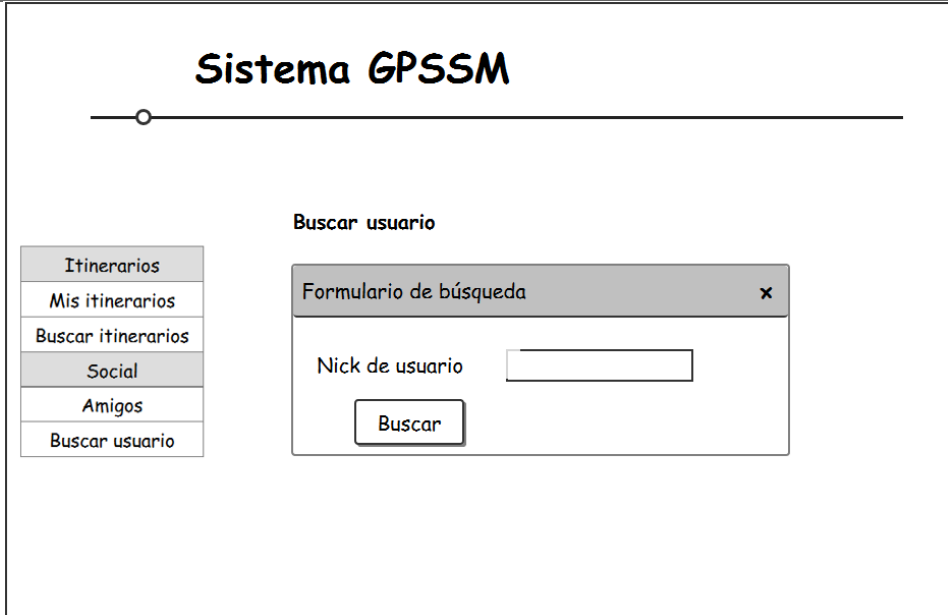
Vista	Buscar usuario
Descripción	Vista para buscar usuarios.
Activación	<ul style="list-style-type: none"> • El usuario pulsa el ítem de menú 'Buscar itinerario'.
Boceto	
Eventos	<ul style="list-style-type: none"> • Búsqueda de usuario por Nick.

Tabla 61. Diseño interfaz Buscar usuario.

Vista	Gestión de puntos de interés
Descripción	Vista en la que el usuario gestor consulta, edita y elimina los puntos de interés dados de alta en la plataforma.
Activación	<ul style="list-style-type: none"> El usuario pulsa el ítem de menú 'Gestión POI'.
Boceto	
Eventos	<ul style="list-style-type: none"> Editar punto. Eliminar punto. Ir a añadir punto.

Tabla 62. Diseño interfaz Buscar Gestión de puntos de interés.

Vista	Añadir punto de interés
Descripción	El usuario gestor utiliza esta vista para dar de alta nuevos puntos de interés.
Activación	<ul style="list-style-type: none"> El usuario pulsa el botón 'Añadir punto' en la vista 'Gestión de puntos de interés'
Boceto	
Eventos	<ul style="list-style-type: none"> Subir imagen para el punto de interés. Añadir punto de interés.

Tabla 63. Diseño interfaz Buscar Añadir punto de interés.

Capítulo 6

Implementación del proyecto

En este capítulo se realiza una descripción de las tecnologías y herramientas usadas para la implementación, así como una descripción de las decisiones y dificultades importantes en el proceso de convertir el diseño de la plataforma en el producto final.

6.1 Servidor de Aplicaciones

Las plataformas desarrolladas bajo las tecnologías J2EE, como el presente proyecto de Plataforma de Gestión de Itinerarios GPSSM, necesitan un servidor de aplicaciones para desplegarse.

Para la implementación de este proyecto se ha seleccionado el servidor de aplicaciones '**GlassFish 4.1**', que se trata de un software libre desarrollado por Sun Microsystems (posteriormente comprado por Oracle), en el que se implementan todas las especificaciones de Java EE 8. La elección de este servidor se debe a su carácter gratuito, su integración con el entorno de desarrollo utilizado (NetBeans), y a la familiarización con este debido a su uso durante la asignatura 'Plataformas Software Empresariales'.

6.2 Servidor de Bases de Datos

Imprescindible para implementar la persistencia del proyecto. El servidor de bases de datos proporciona los servicios necesarios para que el sistema gestor de bases de datos realice las operaciones de almacenamiento, consulta y administración de los datos.

Como servidor se ha elegido '**MySQL**' en su versión 5.7.22. MySQL es un sistema de gestión de bases de datos relacional, desarrollado bajo licencia dual GPL/Licencia comercial por Oracle Corporation, y está considerada como la base de datos open source más popular del mundo. Para su manejo MySQL implementa un lenguaje basado en el estándar SQL. La elección de este servidor se debe a su carácter gratuito y a la gran familiarización con él, ya que ha sido usado en múltiples asignaturas a lo largo del grado, y en especial en la asignatura de 'Administración de Bases de Datos'.

El sistema gestor de bases de datos MySQL puede utilizarse con distintos motores de almacenamiento. La elección del motor de almacenamiento es importante para el diseño de las transacciones ya que afecta a las características del sistema y en aspectos sobre la

integridad. El motor elegido para implementar en el servidor que utilizará esta plataforma es 'InnoDB'. Esta elección se sustenta en los siguientes motivos:

- Soporta transacciones.
- Se espera un uso abundante de operaciones de inserción y actualización en la base de datos, por lo que en determinadas circunstancias es conveniente controlar el nivel de aislamiento para las transacciones.
- Permite garantizar la integridad en las tablas, ya que cumple las características ACID (siglas en inglés de Atomicidad, Consistencia, Aislamiento y Durabilidad).
- Permite bloqueos a nivel de registro.

La conexión de la aplicación Java EE con el servidor de la base de datos, se realiza a través del driver '**mysql-connector 5.1.23**', y la implementación del Framework '**EclipseLink 2.5.2**'. EclipseLink es una implementación de API de Java JPA (Java Persistence API) que nos permite la gestión de datos relacionales en Java a través de clases entidad.

6.3 Principales tecnologías utilizadas

En esta sección se exponen las principales tecnologías usadas para pasar del diseño lógico a implementación.

Java EE

Plataforma que permite desarrollar y ejecutar aplicaciones en lenguaje Java, utilizando arquitecturas de N capas distribuidas y componentes moduladores que son ejecutados en un servidor de aplicaciones.

Java Server Faces 2.2

Se trata de un Framework para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario a través de la tecnología de Java Server Pages (JSP).

Primefaces 5.0

Biblioteca de componentes para Java Server Faces, de código abierto, que comprende un conjunto de componentes desarrollados para facilitar el desarrollo de interfaces web y la presentación de datos en estas. Tiene soporte para Ajax parcial, lo que permite elegir que componentes de una página actual actualizar.

CSS

Lenguaje marcado para definición de estilos en documentos html, xml o xhtml. Utilizado para retocar la estética del tema de Primefaces utilizado.

JavaScript

Lenguaje de programación interpretado y orientado a objetos, permite la ejecución de código en el lado del cliente. Utilizado para pedir confirmaciones al usuario antes de realizar la eliminación de registros.

Redmon Theme

Framework que implementa un tema alternativo al que incluye Primefaces por defecto.

REST

Representational State Transfer (REST), se trata de una arquitectura software para el intercambio de información a través de la 'World Wide Web' a través del protocolo HTTP. Permite desarrollar las operaciones CRUD en la nueva y el intercambio de información en diversos formatos como JSON y XML. Para su implementación en el proyecto se utiliza '**Jersey 2.10.4**', que es una implementación del API de Java **JAX-RS** (Java API for Rest Full Web Services).

JSON

JSON (JavaScript Object Notation) es un formato de texto ligero para el intercambio de información. Los servicios REST de la plataforma transfieren la información de las entidades a través de este formato a la plataforma software móvil y a la mayoría de las vistas.

XML

Metalenguaje para definir lenguajes marcados, que es usado también como estándar para el intercambio de información. Los servicios REST de la plataforma consumen y proporcionan en este formato los datos (además de en JSON) para una mayor compatibilidad en la comunicación con futuros sistemas o plataformas.

7.4 Dificultades técnicas solventadas

La primera tarea en el desarrollo del proyecto fue recordar todo lo aprendido en la asignatura de Plataformas Software Empresariales, los conceptos que necesitamos para trabajar con JAVA EE y sus APIs.

Resuelto este pequeño primer escalón, el primer gran problema fue las dificultades que plantea trabajar con una base de datos relacional y las clases entidad. A través del uso de entidades no podemos hacer consultas de SQL de forma tradicional, ya que con las clases entidad buscamos una abstracción de la base de datos para facilitar futuros cambios. Fue necesario un aprendizaje sobre las librerías de '**CriteriaQuery**', para poder realizar consultas relacionales sobre varias entidades.

La librería de **CriteriaQuery**, cuya implementación en un método REST del proyecto puede verse en la *Figura 141*, precisa de definir varios elementos: la tabla principal objeto de la consulta (*línea 121* de la figura, donde se define la clase entidad de los itinerarios 'Itinerario.class'), la tabla relacionada (*línea 123*, donde se define la clase entidad de los usuarios 'Usuario.class'), la colección de datos que relaciona a la entidad relacionada con la objeto de la consulta (*línea 124*, donde el elemento 'itinerarioCollection' representa la colección de itinerarios de un usuario que nos proporciona EclipseLink a través de la clase entidad) y por ultimo las condiciones de la consulta (que no podemos incluir en las 'NamedQuery' que usa la colección de datos proporcionada por la clase entidad) o clausula 'where' (*línea 125*, donde se indica que queremos buscar los itinerarios donde el usuario sea igual al Nick recibido como parámetro a través de la url en el método REST).

```

116     @GET
117     @Path("/fromuser/{nick}")
118     @Produces({MediaType.APPLICATION_XML, MediaType.APPLICATION_JSON})
119     public Collection<Itinerario> findByUser(@PathParam("nick") String nick) {
120         CriteriaBuilder cb = em.getCriteriaBuilder();
121         CriteriaQuery<Itinerario> query = cb.createQuery(Itinerario.class);
122
123         Root<Usuario> usuario = query.from(Usuario.class);
124         Join<Usuario, Itinerario> it = usuario.join("itinerarioCollection");
125         query.select(it).where(cb.equal(usuario.get("nick"), nick));
126
127         Query exquery = em.createQuery(query);
128         return exquery.getResultList();
129     }
130

```

Figura 141. Fragmento de Código con CriteriaQuery.

Como observación, el elemento 'createNativeQuery', que será mostrado en uso más adelante, también nos podría permitir realizar estas consultas, pero la sintaxis a utilizar en consultas relacionales es compleja y no permite usar la abstracción de las clases entidad, por eso se optó por **CriteriaQuery** como mejor solución.

El siguiente problema, también relacionado con la persistencia, se trataba de la integridad de los datos a la hora de subir un itinerario, y devolver a la plataforma móvil el identificador del itinerario creado, para almacenarlo y poder encontrarlo en la plataforma empresarial para futuras operaciones. Este ID es consultado a la BD a través de la llamada a la función sql 'LAST_INSERT_ID()', que nos devuelve el ID del último elemento insertado en una entidad. El problema que podía surgir es, que ante una situación de alta carga de trabajo y varios usuarios subiendo itinerarios concurrentemente, podría darse el caso de que el gestor de consultas de MySQL entremezclara las inserciones de varios itinerarios, no ejecutando inmediatamente después la consulta del ultimo ID insertado.

Por ello ha sido necesario definir la inserción del itinerario y la consulta del ultimo ID insertado como una transacción con un nivel de aislamiento 'Serializable', de forma que no obtengamos lecturas incorrectas y estas dos operaciones se ejecuten en bloque, garantizando la concordancia de respuesta a la plataforma móvil sobre el ID en la PSE del itinerario subido. EclipseLink, en el tipo de transacción 'JTA' (por defecto), define cada operación o consulta con la base de datos como una transacción independiente, porque lo fue necesario la investigación

para poder implementar una transacción de varias intrucciones, y además modificar el nivel de aislamiento por defecto (Repeatable Read). La solución fue el uso del objeto Java *'UnitOfWork'*, que nos permite agrupar todo tipo de operaciones de BD como una única unidad de trabajo.

```

132 @POST
133 @Path("/uploadfrommobile")
134 @Consumes({MediaType.APPLICATION_XML, MediaType.APPLICATION_JSON})
135 @Produces(MediaType.TEXT_PLAIN)
136 public String nuevoItinerario(Itinerario entity) throws SQLException {
137     String resultado;
138
139     //Cambiamos nivel de aislamiento
140     java.sql.Connection connection = em.unwrap(java.sql.Connection.class);
141     connection.setTransactionIsolation(Connection.TRANSACTION_SERIALIZABLE);
142
143     UnitOfWork uow = em.unwrap(UnitOfWork.class);
144
145     em.persist(entity);
146     //CONSULTA ULTIMO ID INSERTADO
147     Query q = em.createNativeQuery("SELECT idItinerario FROM itinerario where idItinerario=LAST_INSERT_ID() ");
148     resultado = q.getResultList().get(0).toString();
149     System.out.println("REsultado ID: "+resultado);
150     uow.commit();
151
152     //Devolvemos nivel de aislamiento
153     connection.setTransactionIsolation(Connection.TRANSACTION_REPEATABLE_READ);
154
155     return resultado;
156 }

```

Figura 142. Fragmento de Código con UnitOfWork.

El uso de la librería *UnitOfWork* puede apreciarse en la Figura 142. Primero en la línea 143, se define el objeto de unidad de trabajo. Después, se persiste la nueva entidad Itinerario y se consulta en la base de datos a través del elemento *'createNativeQuery'* cual es el id del Itinerario que acabamos de insertar. Y por último en la línea 150 de la figura se cierra la unidad de trabajo, cerrando el fragmento de operaciones que será ejecutado como una transacción única. Puede verse también en las líneas 141 y 153 como se cambia el nivel de aislamiento de las transacciones en la base de datos antes y después de la ejecución de esta.

```

113 @POST
114 @Path("/frommobile")
115 @Consumes(MediaType.TEXT_PLAIN)
116 @Produces(MediaType.TEXT_PLAIN)
117 public String uploadP(String body) throws JSONException, IOException, SQLException {
118     String resultado;
119     Posiciones pos = new Posiciones();
120     JSONArray jsonarray = new JSONArray(body);
121     //Cambiamos nivel de aislamiento
122     java.sql.Connection connection = em.unwrap(java.sql.Connection.class);
123     connection.setTransactionIsolation(Connection.TRANSACTION_SERIALIZABLE);
124     //DEFINIMOS TRANSACCION
125     UnitOfWork uow = em.unwrap(UnitOfWork.class);
126     for (int i = 0; i < jsonarray.length(); i++) {
127         JSONObject jsonobject = jsonarray.getJSONObject(i);
128         System.out.println(jsonobject.toString());
129         //mapeo del objeto json para convertirlo en objeto entidad
130         pos = new ObjectMapper().readValue(jsonobject.toString(), Posiciones.class);
131         em.persist(pos);
132     }
133     //obtenemos id de la ultima posicion insertada
134     Query q = em.createNativeQuery("SELECT idPos FROM posiciones where idPos=LAST_INSERT_ID() ");
135     resultado = q.getResultList().get(0).toString();
136     uow.commit();
137     //Devolvemos nivel de aislamiento
138     connection.setTransactionIsolation(Connection.TRANSACTION_REPEATABLE_READ);
139     return resultado;
140 }
141
142 }

```

Figura 143. Fragmento de código con librerías de Jackson ObjectMapper.

Este misma necesidad surgió para las posiciones, pero con un problema añadido. Se pretendía la inserción en la base de datos de todas las entidades 'Posiciones' en una misma transacción, por lo que era necesario recibir en un mismo JSON todas las posiciones de itinerario, procesarlas como entidad y persistirlas todas en una misma transacción. Para resolver este requerimiento fue necesario el uso de la librería '**Jackson ObjectMapper**', para mapear cada entidad contenida en el JSON, siendo procesado este como un **JSONArray**. Toda esta operación incluida dentro de una **UnitOfWork**, garantizando que todas o ninguna de las posiciones sean insertadas.

El uso de la librería de **Jackson ObjectMapper** requiere que se añadan al proyecto dos dependencias: '*jackson-core 2.6.2*' y '*jackson-databind 2.9.5*'. Estas pueden ser fácilmente añadidas a través de buscador de dependencias que proporciona NetBeans, sin necesidad de descargar los ficheros '*jar*' manualmente. En el código que puede verse en la *Figura 143*, puede observarse que la cadena de texto recibida, que será un Json, es almacenada en un **JSONArray** (ver línea 120). Una vez definida la transacción y cambiado el nivel de aislamiento, mediante un bucle for (ver línea 126 hasta 132), se recorre el **JSONArray** obteniendo los distintos objetos Json que se corresponden cada uno con un objeto 'Posiciones'. En la línea 130, se realiza el mapeo del Json en la entidad Posiciones gracias a el objeto '**ObjectMapper**' de la librería de **Jackson ObjectMapper**.

Finalmente, el problema más complicado de resolver, y que resulto con diferencia el más frustrante y costoso en tiempo, surgió a la hora de desarrollar la consulta de Itinerarios. Primefaces, el framework de estilo para JSF usado en este proyecto, tiene un objeto '**GMap**' para insertar mapas de Google Maps en las vistas xhtml. Como los propios mapas de Google permite dibujar polilíneas (para marcar el recorrido) y marcadores personalizados (para marcar los puntos multimedia). Este elemento se añadía sin problema a la vista, sin ningún tipo de información consultada a la base de datos. El problema, fue que, al consultar las posiciones para dibujar el recorrido, e implementar que estas se unieran con una polilíneas en el mapa para dibujar el recorrido, este sufría problemas de carga. Concretamente el mapa solo se conseguía cargar a veces, y no siempre con todos los elementos dibujados. Este problema no generaba ningún tipo de mensaje de error, ni si quiera en la consola de Javascript del navegador. La conclusión a la que llegué fue que el problema era que la consulta de las posiciones se demoraba más que el renderizado de la página, y por ello el mapa no llegaba a mostrarse la mayoría de las veces. Traté de realizar la consulta en una vista anterior pero no dio resultado, por lo que la siguiente prueba fue intentar algún tipo de carga en segundo plano, y que el mapa se renderizara una vez cargados todos los datos. Aunque a través de un modo del elemento 'OutputPanel', se implementó una carga después del renderizado de la página. Aunque esta solución mejoró el porcentaje de veces que se conseguía cargar el mapa con éxito, seguía fallando a veces sin un motivo aparente. Tras más de un mes, empleado en investigar el origen del problema y encontrar una posible solución, surgió la idea de emplear un botón que hiciera una recarga de la página mediante Ajax. Afortunadamente este método, a pesar de no ser la solución más estética funciona, por lo que implemente un botón, que pulsándose realizaba una actualización de la zona de la vista donde se encuentra el mapa, consiguiendo su carga con todos los elementos dibujados el 100% de las veces. Para una mejor estética, implemente que este botón desapareciera una vez pulsado.

Esta implementación puede verse en la *Figura 144*. En la línea 124 se define el botón que posteriormente se ocultará, y en la 136 el panel que contiene el elemento **GMap** (línea 139) y que inicialmente no se renderizará.

```

122 <div style="height:10px"></div> <h:form prependId="false" id="main_content" >
123 <p:panel id="botPanel" rendered="#{itinerarioClientBean.botPanel}" >
124 <p:commandButton update="main_content:outPanel" widgetVar="btnSend" id="btnsend"
125 value="Ver mapa" actionListener="#{itinerarioClientBean.setShowMap(true)}"
126 onclick="btnSend.disable()" /> </p:panel>
127 <div style="height:10px"></div>
128
129 <p:panel id="outPanel" widgetVar="outpanel" closable="true" toggleable="true" rendered="#{itinerarioClientBean.showMap}">
130 <div id="rec" class="ui-tabs ui-widget ui-widget-content ui-corner-all ui-hidden-container ui-tabs-top" >
131 <ul class="ui-tabs-nav ui-helper-reset ui-helper-clearfix ui-widget-header ui-corner-all" role="tablist">
132 <li class="ui-state-default ui-tabs-selected ui-state-active ui-corner-top" role="tab" aria-expanded="true">
133 <a>Recorrido</a></li></ul><div class="ui-tabs-panels">
134 <div id="rec2" class="ui-tabs-panel ui-widget-content ui-corner-bottom" role="tabpanel" aria-hidden="false">
135
136 <p:outputPanel deferred="true" deferredMode="visible">
137
138
139 <p:gmap id="gm" center="#{itinerarioClientBean.mapCenterLat}, #{itinerarioClientBean.mapCenterLong}" zc
140 <p:ajax event="overlaySelect" listener="#{itinerarioClientBean.onMarkerSelect}" />
141 <p:outputPanel style="text-align: center; display: block; margin: auto">
142 <p:graphicImage name="../../../Foto.png" height="150" />
143 <br />
144 <h:outputText value="#{itinerarioClientBean.marker.title}" /><br />
145
146 </p:outputPanel>
147 </p:gmap>
148
149 </p:outputPanel>
150

```

Figura 144. Fragmento de código de vista itinerario.xhtml

7.5 Herramientas utilizadas

En esta sección se mencionan las herramientas más destacadas utilizadas en la implementación del proyecto, así como también en la confección de esta memoria.

- **Dia diagramas:** Software de realización de diagramas usado para modelos entidad-relación, arquitectura física y diagrama de casos de uso.
- **EasyUML:** Plugin para NetBeans utilizado para realizar el diagrama de clases.
- **Firefox 61.0:** Navegador web utilizado para la ejecución del proyecto.
- **Microsoft Office Power Point 2016:** Programa ofimático de presentaciones utilizado para el desarrollo de diagramas de arquitectura y de iteraciones de desarrollo.
- **Microsoft Office Word 2016:** Programa ofimático de edición de documentos de texto utilizado para la realización de la presente memoria.
- **MySQL WorkBench 6.3:** Gestor gráfico de bases de datos MySQL, utilizado para obtener agilidad en pruebas sencillas con la base de datos, y para realizar el modelo Entidad-Relación.
- **NetBeans IDE 8.2:** Entorno de desarrollo para JAVA EE (y otros lenguajes), con Maven y GlassFish integrados para proporcionar facilidades de desarrollo al programador.
- **Postman:** Cliente REST Avanzado, utilizado para comprobar el envío y descarga de imágenes desde la plataforma.
- **QuickMockup:** Aplicación web para el diseño de interfaces de usuario (<https://jdittrich.github.io/quickMockup>).

- **RestClient** (extension para firefox): Cliente REST utilizado para probar todos los métodos que dan soporte a los servicios REST implementados.
- **Tomsplanner**: Aplicación web para la realización de diagramas de Gantt (<https://www.tomsplanner.es/?template=example>).
- **Visual Paradigm 15**: Software para la realización de diagramas de diseño de software, utilizado en la realización de los diagramas de secuencia.

Capítulo 7

Pruebas

En este capítulo figuran las pruebas más importantes realizadas en la etapa de 'Pruebas' en cada iteración (*ver detalle sobre iteraciones en punto 3.1*). Estas pruebas complementan a las llevadas a cabo durante el desarrollo.

Las pruebas desarrolladas para este proyecto son de caja negra. En estas pruebas se determina el correcto funcionamiento del sistema sin tener en cuenta como está diseñado por dentro, es decir, se busca una salida esperada para una determinada entrada.

7.1 Pruebas de la Iteración I

Las pruebas realizadas para la primera iteración son las siguientes:

ID de prueba	P01
Descripción	El usuario se registra en el sistema
Prerrequisito	Ninguno
Datos de entrada	Información de registro cumplimentada en todos los campos.
Resultado previsto	Se muestra el formulario de Login y el nuevo usuario queda registrado en la base de datos.
Resultado real	Correcto

Tabla 64. Prueba del sistema P01.

ID de prueba	P02
Descripción	El usuario se loguea en el sistema
Prerrequisito	El usuario está registrado
Datos de entrada	Nick y contraseña
Resultado previsto	Se autentica el usuario en el sistema y se muestra el panel de usuario con sus itinerarios.
Resultado real	Correcto

Tabla 65. Prueba del sistema P02.

ID de prueba	P03
Descripción	Se pretende comprobar el acceso al sistema con datos erróneos.
Prerrequisito	El usuario está registrado
Datos de entrada	Nick y contraseña errónea.
Resultado previsto	Se le muestra un mensaje de error al usuario y se le muestra un mensaje de error.
Resultado real	Incorrecto. No se autentica el usuario, pero tampoco se le muestra el mensaje de error.

Tabla 66. Prueba del sistema P03.

ID de prueba	P03b
Descripción	Se pretende comprobar el acceso al sistema con datos erróneos después de la corrección tras prueba P03.
Corrección previa	Se corrige llamada método en actionListener.
Prerrequisito	El usuario está registrado
Datos de entrada	Nick y contraseña errónea.
Resultado previsto	Se le muestra un mensaje de error al usuario y se le muestra un mensaje de error.
Resultado real	Correcto.

Tabla 67. Prueba del sistema P03b.

ID de prueba	P04
Descripción	Se pretende hacer una petición GET para obtener todos los usuarios de la plataforma.
Prerrequisito	Cliente REST
Datos de entrada	Cabecera 'Accept:application/json'
Resultado previsto	JSON con todos los usuarios registrados en la plataforma.
Resultado real	Correcto.

Tabla 68. Prueba del sistema P04.

ID de prueba	P05
Descripción	Se pretende actualizar la información de un usuario a través de una petición PUT
Prerrequisito	El usuario que se pretende modificar existe en la base de datos.
Datos de entrada	JSON con la nueva información de la entidad Usuario.
Resultado previsto	Respuesta '204 No Content', y modificación en la base de datos.
Resultado real	Correcto

Tabla 69. Prueba del sistema P05.

ID de prueba	P06
Descripción	Se pretende eliminar un usuario de la plataforma.
Prerrequisito	El usuario que se pretende borrar existe en la base de datos.
Datos de entrada	Nick del usuario (en URL)
Resultado previsto	Respuesta '204 No Content', y borrado del usuario en la base de datos.
Resultado real	Correcto

Tabla 70. Prueba del sistema P06.

7.2 Pruebas de la Iteración II

Las pruebas realizadas para la segunda iteración son las siguientes:

ID de prueba	P07
Descripción	Se pretende visualizar la lista de usuarios del sistema.
Prerrequisito	El usuario está registrado y logueado.
Datos de entrada	<i>Ninguno</i>
Resultado previsto	Se muestra un listado de los itinerarios del usuario
Resultado real	Correcto

Tabla 71. Prueba del sistema P07.

ID de prueba	P08
Descripción	Se pretende consultar un itinerario en la plataforma web.
Prerrequisito	El usuario está registrado y logueado, y además tiene al menos un itinerario registrado en la plataforma.
Datos de entrada	Id del itinerario a visualizar.
Resultado previsto	Se muestra la información del itinerario, así como una galería de las imágenes de este, y un mapa con el recorrido.
Resultado real	Incorrecto. El mapa no se muestra.

Tabla 72. Prueba del sistema P08.

ID de prueba	P08b
Descripción	Se pretende consultar un itinerario en la plataforma web, después de intentar una corrección tras fallo de la prueba P08.
Corrección previa	Se implementa un elemento OutputPanel con 'Lazy Load'
Prerrequisito	El usuario está registrado y logueado, y además tiene al menos un itinerario registrado en la plataforma.
Datos de entrada	Id del itinerario a visualizar.
Resultado previsto	Se muestra la información del itinerario, así como una galería de las imágenes de este, y un mapa con el recorrido.
Resultado real	Incorrecto. El mapa se muestra solo en algunas ocasiones.

Tabla 73. Prueba del sistema P08b.

ID de prueba	P08c
Descripción	Se pretende consultar un itinerario en la plataforma web, después de intentar una corrección tras fallo de la prueba P08b.
Corrección previa	Se implementa un botón que permite renderizar un OutputPanel con 'Lazy Load' que inicialmente está oculto.
Prerrequisito	El usuario está registrado y logueado, y además tiene al menos un itinerario registrado en la plataforma.
Datos de entrada	Id del itinerario a visualizar.
Resultado previsto	Se muestra la información del itinerario, así como una galería de las imágenes de este, y un mapa con el recorrido.
Resultado real	Correcto

Tabla 74. Prueba del sistema P08c.

ID de prueba	P09
Descripción	Se pretende modificar el título de un punto multimedia.
Prerrequisito	Usuario logueado es creador del itinerario.
Datos de entrada	Nuevo título
Resultado previsto	La información actualizada se presenta por pantalla y se realiza la modificación en la base de datos.
Resultado real	Correcto

Tabla 75. Prueba del sistema P09.

ID de prueba	P10
Descripción	Se pretende, mediante un POST desde un cliente REST subir un itinerario, y con una segunda petición POST, subir las posiciones.
Prerrequisito	El itinerario no existe en la plataforma
Datos de entrada	JSON con datos del itinerario y JSON con todas las posiciones.
Resultado previsto	Respuesta '204 No Content' en ambas peticiones.
Resultado real	Incorrecto, la petición de las posiciones devolvió '400 Bad Request'.

Tabla 76. Prueba del sistema P10.

ID de prueba	P10b
Descripción	Se pretende, mediante un POST desde un cliente REST subir un itinerario, y con una segunda petición POST, subir las posiciones. Prueba realizada tras corrección por fallo en prueba P10.
Corrección previa	Tras comprobar que el problema era que el método REST solo podía tratar una sola entidad Posiciones, se trató el json recibido por el método para subir posiciones como un 'JSONArray', y se usó la librería 'Jackson ObjectMapper', para persistir como entidad cada objeto del JSONArray.
Prerrequisito	El itinerario no existe en la plataforma
Datos de entrada	JSON con datos del itinerario y JSON con todas las posiciones.
Resultado previsto	Respuesta '204 No Content' en ambas peticiones.
Resultado real	Correcto

Tabla 77. Prueba del sistema P10b.

ID de prueba	P11
Descripción	Se pretenden la imagen de un punto multimedia de un itinerario.
Prerrequisito	El itinerario y sus posiciones ya están creados en la base de datos.
Datos de entrada	Nombre de la imagen e ID de itinerario.
Resultado previsto	Respuesta '204 No Content'.
Resultado real	Correcto

Tabla 78. Prueba del sistema P11.

7.3 Pruebas de la Iteración III

Las pruebas realizadas para la tercera iteración son las siguientes:

ID de prueba	P12
Descripción	Se pretende comprobar si el sistema muestra todas las peticiones de amistad pendientes de resolver.
Prerrequisito	El usuario está logueado y tiene peticiones sin resolver.
Datos de entrada	<i>Ninguno</i>
Resultado previsto	Lista de peticiones de amistad en las que el usuario es solicitado.
Resultado real	Correcto

Tabla 79. Prueba del sistema P12.

ID de prueba	P13
Descripción	Se pretende comprobar la funcionalidad de aceptar una petición de amistad
Prerrequisito	El usuario está logueado y tiene al menos una petición pendiente.
Datos de entrada	<i>Ninguno</i>
Resultado previsto	La petición desaparece, y el solicitante aparece en la lista de amigos.
Resultado real	Correcto

Tabla 80. Prueba del sistema P13.

ID de prueba	P14
Descripción	Se pretende comprobar la funcionalidad de rechazar una petición de amistad
Prerrequisito	El usuario está logueado y tiene al menos una petición pendiente.
Datos de entrada	<i>Ninguno</i>
Resultado previsto	La petición desaparece de la lista, y es borrada de la base de datos.
Resultado real	Correcto

Tabla 81. Prueba del sistema P14.

ID de prueba	P15
Descripción	Se pretende eliminar a un usuario de la lista de amigos.
Prerrequisito	El usuario está logueado y tiene al menos una relación de amistad.
Datos de entrada	ID de usuario solicitante e id de usuario solicitado
Resultado previsto	La relación de amistad desaparece de la tabla de amistades, y es borrada de la base de datos.
Resultado real	Correcto

Tabla 82. Prueba del sistema P15.

ID de prueba	P16
Descripción	Se pretende acceder al perfil de usuario de un amigo.
Prerrequisito	El usuario está logueado y tiene al menos una relación de amistad.
Datos de entrada	ID usuario amigo.
Resultado previsto	Se visualiza el perfil e itinerarios del usuario deseado.
Resultado real	Correcto

Tabla 83. Prueba del sistema P16.

7.4. Pruebas de la Iteración IV

Las pruebas realizadas para la cuarta iteración son las siguientes:

ID de prueba	P17
Descripción	Se pretende realizar la creación de un punto de interés.
Prerrequisito	El usuario logueado es del tipo 'Gestor'.
Datos de entrada	Datos de entidad 'POI' e imagen.
Resultado previsto	El punto de interés es añadido
Resultado real	Correcto

Tabla 84. Prueba del sistema P17.

ID de prueba	P18
Descripción	Se pretende realizar la modificación de un punto de interés.
Prerrequisito	El usuario logueado es del tipo 'Gestor'.
Datos de entrada	Atributo para modificar.
Resultado previsto	El sistema cambia el POI en la base de datos y muestra el resultado modificado.
Resultado real	Correcto

Tabla 85. Prueba del sistema P18.

ID de prueba	P19
Descripción	Se pretende realizar la eliminación de un punto de interés.
Prerrequisito	El usuario logueado es del tipo 'Gestor'.
Datos de entrada	ID POI.
Resultado previsto	El punto de interés es eliminado y desaparece de la tabla de consulta.
Resultado real	Incorrecto, el punto es eliminado de la base de datos, pero sigue mostrándose por pantalla.

Tabla 86. Prueba del sistema P19.

ID de prueba	P19b
Descripción	Se pretende realizar la eliminación de un punto de interés y comprobar que desaparece de la pantalla, después de la modificación motivada por el fallo de la prueba <i>P19</i> .
Corrección previa	Se añade una llamada Ajax en el botón de borrado para recargar la página tras el borrado.
Prerrequisito	El usuario logueado es del tipo 'Gestor'.
Datos de entrada	ID POI.
Resultado previsto	El punto de interés es eliminado y desaparece de la tabla de consulta.
Resultado real	Correcto

Tabla 87 Prueba del sistema P19b.

Capítulo 8

Manuales

En este capítulo se incluyen los manuales necesarios para la puesta en marcha de la plataforma y su uso eficaz por parte de los usuarios.

8.1 Manual de instalación y despliegue

Antes de desplegar y utilizar la aplicación empresarial de GPSSM es necesario la preparación de un servidor con múltiples servicios. Para ello se requiere llevar cabo múltiples instalaciones y configuraciones.

8.1.1 Preparación del servidor

La plataforma se ha implementado sobre un sistema operativo Ubuntu Server 16.04 LTS, aunque podría serlo en otros entornos gracias a la característica multiplataforma de JAVA.

En este servidor, debemos de instalar un servidor de bases de datos MySQL. Concretamente instalamos la versión '5.7.22-0ubuntu0.16.04.1'. Esta instalación se lleva a cabo mediante el comando '*sudo apt-get install mysql-server*' y la posterior configuración de la instalación corriendo el comando '*sudo mysql_secure_installation*'. Posteriormente accedemos al terminal mediante el comando '*mysql -u root -p*', donde realizaremos la ejecución de la DDL (incluida en el CD del proyecto, cuyo contenido viene detallado en el ANEXO II).

Para la ejecución de del proyecto en una máquina virtual java, debemos proceder a la instalación de este lenguaje de programación, concretamente en su versión 8. Para su instalación desde el terminal de comando se deben de ejecutar las siguientes líneas:

```
“sudo add-apt-repository ppa:webupd8team/java  
sudo apt-get update  
sudo apt-get install oracle-java8-installer”
```

Por último, es necesaria la instalación del servidor de aplicaciones Glassfish. Este servidor desplegará la aplicación. Se ha utilizado la versión 4.1 porque fue la usada durante las prácticas de la asignatura de 'Plataformas Software Empresariales', y no dio ningún tipo de problema o

limitación. La versión puede descargarse en la siguiente dirección: [‘http://download.oracle.com/glassfish/4.1/release/index.html’](http://download.oracle.com/glassfish/4.1/release/index.html).

Una vez descargado el fichero procedemos a su instalación, siguiendo los siguientes comandos:

```
“wget http://download.oracle.com/glassfish/4.1/release/glassfish-4.1.zip  
unzip glassfish-4.1.1.zip -d /opt”
```

Es necesario establecer la variable PATH del Sistema para Glassfish:

```
“nano ~/.profile”
```

Y añadimos al final la línea:

```
export PATH=/opt/glassfish4/bin:$PATH
```

Arrancamos el servidor, mediante e siguiente comando, y comprobamos en la dirección <http://localhost:4848>.

```
“asadmin start-domain”
```

Por último, es necesario establecer la configuración para que nos permita acceder a la administración de GlassFish. Para ello introducimos los siguientes comandos:

```
“asadmin change-admin-password –domain_name domain1  
asadmin enable-secure-admin –port 4848  
asadmin restart-domain”
```



Figura 145. Pantalla de administración de GlassFish.

Tras introducir el primer comando tendremos que introducir una contraseña, que nos pedirá al introducir el segundo comando. El último comando es para reiniciar el servidor. Después, ya podemos acceder a la administración del servidor (*ver en figura 145*) en la dirección antes mencionada (<http://localhost:4848>).

8.1.2 Configuración de GlassFish

Una vez instalado el servidor GlassFish es necesario realizar algunos ajustes.

El primero de ellos se trata del puerto de ejecución del dominio. Por defecto en GlassFish los proyectos se despliegan a través del puerto 8080, y debemos de cambiarlo al puerto 80, ya que la máquina virtual proporcionada por la UVA tiene ese puerto redirigido para su salida al exterior, y no es posible hacerlo a través del 8080.

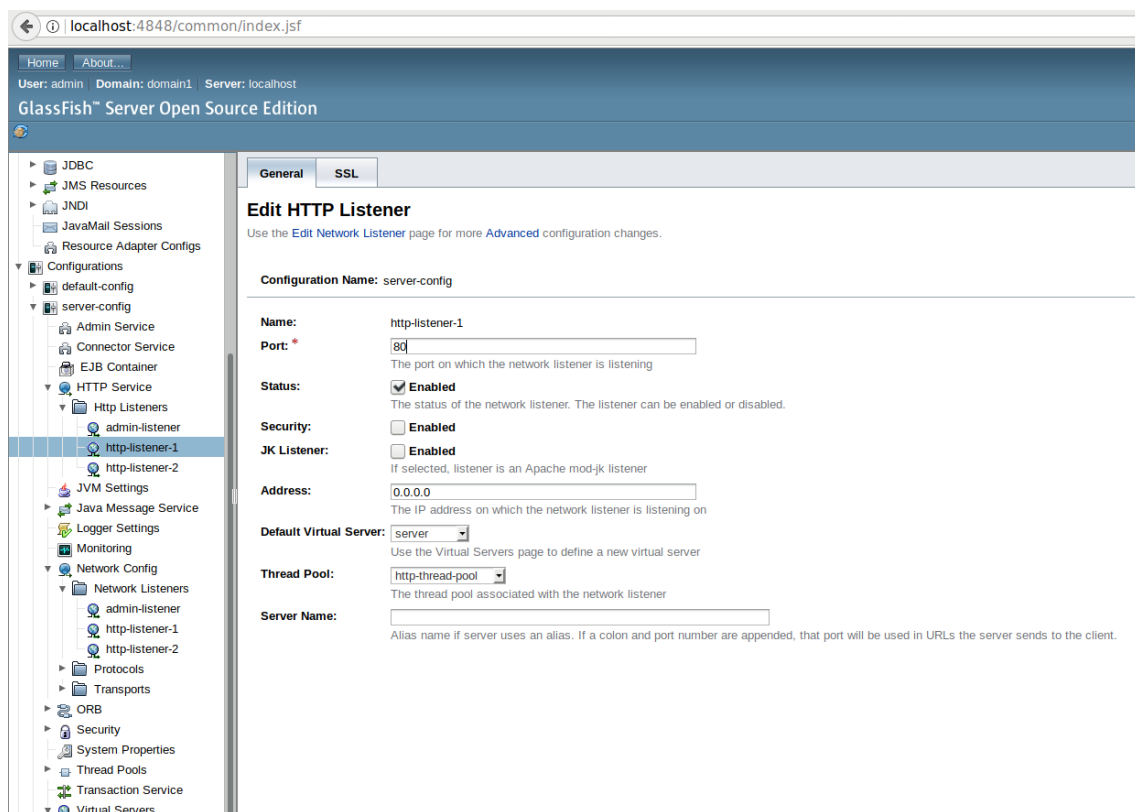


Figura 146. Configuración de puerto de salida HTTP.

Para ello, en el panel de configuración de GlassFish, seleccionamos en el menú izquierdo 'Configurations → server-config → HTTP Service → http listeners → http-listener 1', y en el campo 'Port' cambiamos el valor 8080 por el 80.

El siguiente ajuste que debemos realizar en el servidor de aplicaciones, es la configuración del 'JDBC Security Realm', que realizará la gestión de la seguridad y la autenticación en la aplicación, gracias al uso del API Java 'JAAS' (Java Authentication and Authorization Service).

El primer paso para añadir el Realm, es la configuración en GlassFish de un Pool de conexiones previamente configurado en la aplicación. Para ello es necesario disponer del fichero 'glassfish-resources.xml', este fichero está disponible en el CD del proyecto (ver **ANEXO II para ver detalle de contenidos**). Como podemos ver en la *Figura 147*, en el panel de administración de Glassfish debemos de ir al menú a la sección de 'Resources', e indicar que deseamos añadir un nuevo recurso. Marcamos 'XML File to Be Uploaded to the Server' y seleccionamos el fichero 'glassfish-resources.xml'.

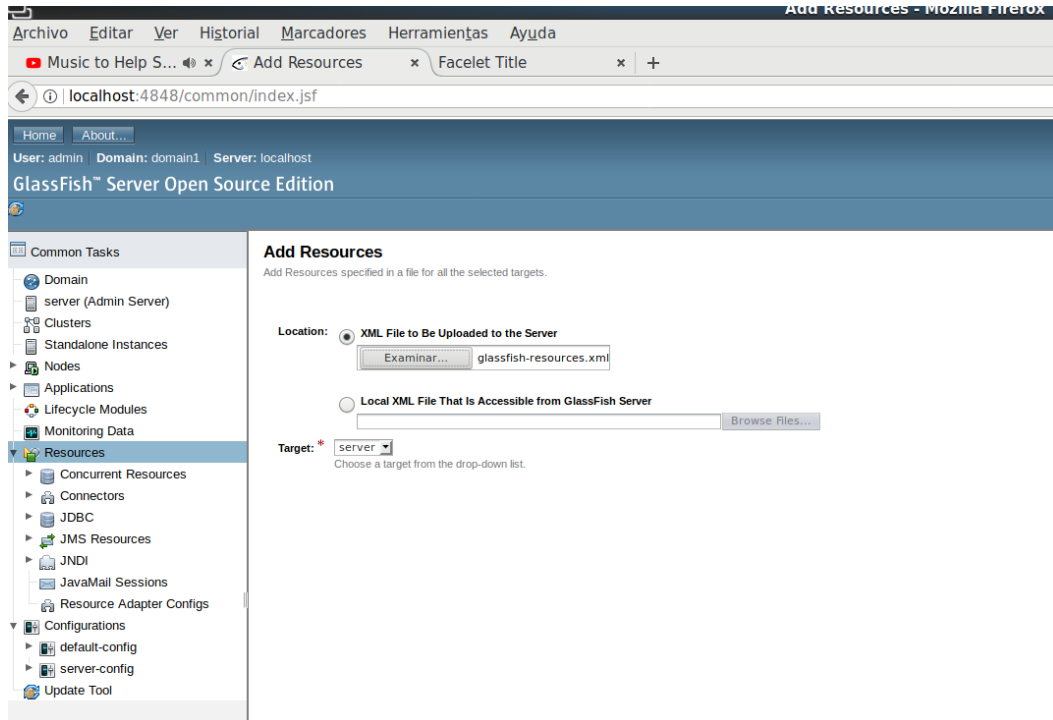


Figura 147. Configuración de 'pool' de conexiones.

Una vez pulsemos en 'Ok', tras recargar la página podemos comprobar qué, desplegando el menú de 'Resources →JDBC', tanto en la pestaña de 'JDBC Resources' y en la de 'JDBC Connection Pool', han aparecido respectivamente los elementos resaltados en la *Figura 148*.

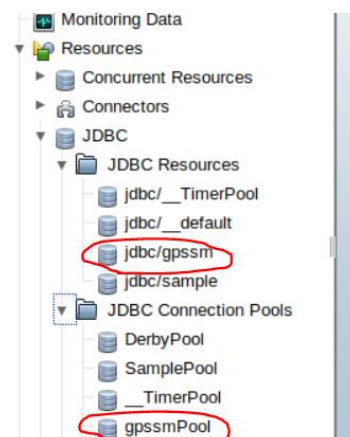


Figura 148. Recursos JDBC añadidos a GlassFish.

Por último, debemos de crear el 'Realm', para lo que es necesario ir a 'Configurations → server-config → Security → Realms'. Una vez en esta pantalla pulsamos en 'New' para crear un nuevo Realm.

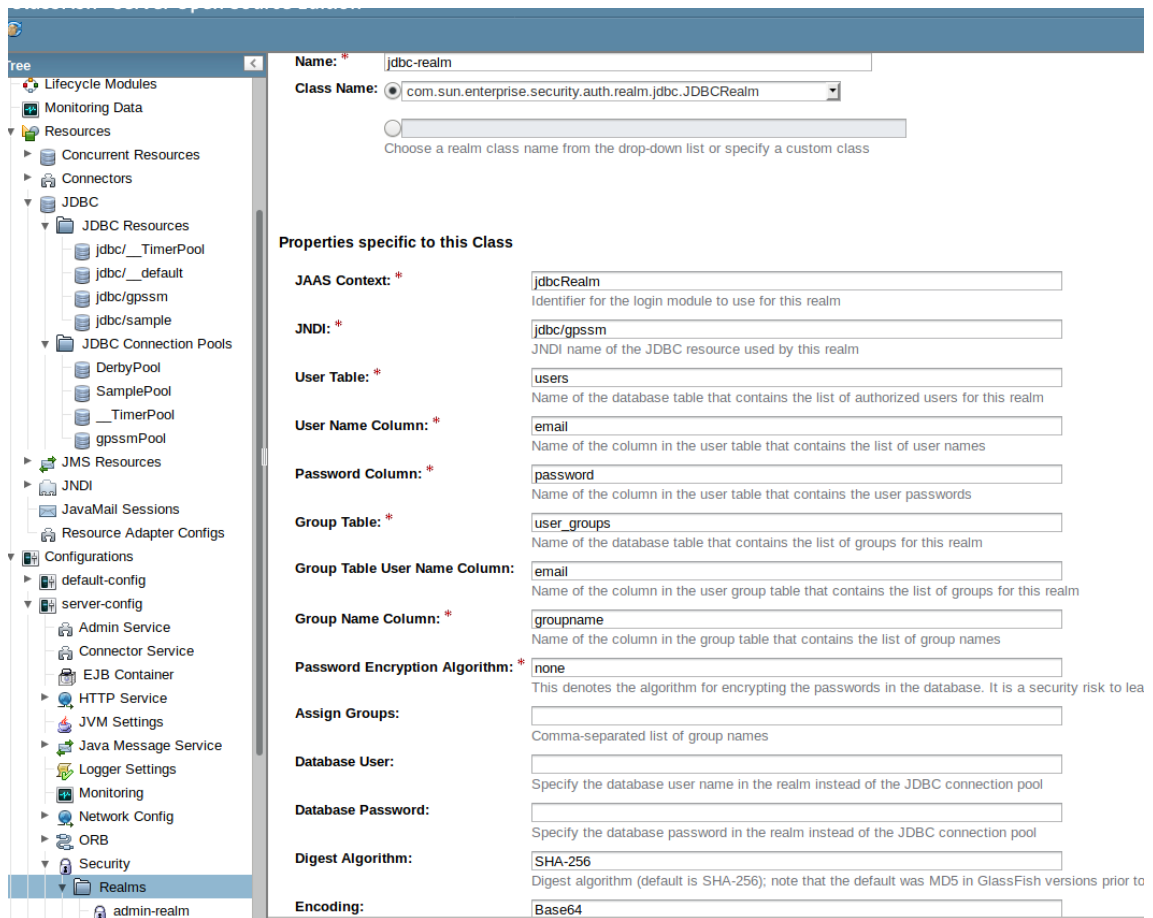


Figura 149. Recursos JDBC añadidos a GlassFish.

Una vez abierta la nueva pantalla, en la *Figura 149* podemos observar los datos que son preciso rellenar. Se trata de la información el nombre del Realm, el recurso creado anteriormente, y los campos clave de las tablas de la base de datos que van a usarse con JAAS. También es necesario incluir el tipo de codificación que se va a usar para codificar las contraseñas.

8.1.3 Despliegue de la aplicación

Para el despliegue de aplicaciones en GlassFish debemos de seleccionar la sección 'Aplicaciones' del menú de configuración (<http://localhost:4848>). Dentro de este menú debemos de pulsar sobre el botón 'Desplegar', marcamos en el formulario la 'archivo empaquetado que Cargar al Servidor'. Proporcionamos el fichero de extensión '.war' del proyecto y pulsamos en 'Aceptar'.

Desplegar Aplicaciones o Módulos

Especifique la ubicación de la aplicación o del módulo que desea desplegar. Una aplicación puede estar en un archivo empaquetado o se puede especificar como directorio.

Ubicación: Archivo Empaquetado que Cargar en el Servidor

GPSSM-1.0-SNAPSHOT.war

Archivo empaquetado local o directorio accesible desde GlassFish Server

Tipo: *

Raíz de Contexto:
Ruta de acceso relativa a la URL base del servidor.

Nombre de Aplicación: *

Servidores Virtuales:
Asocia un nombre de dominio de Internet a un servidor físico.

Estado: **Activada**
Permite a los usuarios acceder a la aplicación.

Implicit CDI: **Activada**
Implicit discovery of CDI beans

Precompilar JSP:
Precompila páginas JSP durante el despliegue.

Ejecutar Verificador:
Verifica la sintaxis y semántica del descriptor de despliegue. Los paquetes de verificador deben estar instalados.

Forzar Nuevo Despliegue:
Fuerza el nuevo despliegue incluso aunque esta aplicación ya se haya desplegado o ya exista.

Mantener Estado:
Mantiene sesiones web, instancias SFSB y temporizadores EJB creados de forma persistente entre nuevos despliegues.

Figura 150. Despliegue de la aplicación en GlassFish.

Una vez desplegada, la aplicación será accesible en la URL 'http://localhost/GPSSM', accediendo con un navegador desde dentro del propio servidor.

8.2 Manual de usuario

Para explicar a los usuarios finales el uso de las principales funcionalidades del portal web del sistema, se incluye el siguiente manual.

Lo primero que encuentra el usuario al abrir el portal web es una pantalla de bienvenida donde se describe la herramienta, y un enlace donde descargar la aplicación móvil (en concreto para el sistema operativo móvil Android).

La web en todas sus interfaces cuenta con una cabecera, un menú a la izquierda, y un espacio de contenido. Los menús varían según el usuario este autenticado o no, y en caso de estarlo, según sea el tipo de usuario. Como puede verse en la *Figura 151*, los tres menús aumentan o cambian sus enlaces. A la izquierda del todo, un usuario sin identificar puede loguearse en el sistema o registrarse. En el menú del centro, que se corresponde a los usuarios del tipo

‘Normal’, podemos ver que podemos acceder a las funcionalidades de ver nuestros itinerarios, buscar itinerarios, ver nuestros amigos, buscar usuarios y cerrar sesión. Por último, a la derecha del todo podemos observar el menú del usuario identificado del tipo ‘Gestor’, que difiere del anterior en que añade un enlace a la funcionalidad de gestionar los puntos de interés.



Figura 151. Menús de la aplicación.

Para organizar el manual, va a clasificarse la funcionalidad por usuario sin identificar, usuario normal y usuario gestor. Hay que tener en cuenta que el usuario gestor tiene acceso a todas las pantallas y puede realizar todas las acciones que realiza el usuario normal como puede apreciarse en su menú de usuario.

8.2.1 Manual del usuario sin identificar

El usuario sin identificar desarrolla principalmente las acciones de registro y login.

El registro, como puede apreciarse en la *Figura 152*, es un formulario simple que requiere la intrducción de todos los datos. El nick de usuario debe de ser único y no existir previamente en la plataforma, si fuera así el sistema nos avisaría. También si nos quedara algún campo sin rellenar o las contraseñas no coincidieran.

Sistema GPSSM
Plataforma de Gestión de Itinerarios GPS Sightseeing Moments

Menu

Gestión de usuario

- Acceder
- Registro

Formulario de registro

Nick:

Nombre:

Extracto:

Contraseña:

Verificación de password:

Figura 152. Pantalla de Registro.

Una vez rellenados todos los datos, el sistema nos redirige a la pantalla de Login, la cual podemos observar en la *Figura 153*.

Sistema GPSSM
Plataforma de Gestión de Itinerarios GPS Sightseeing Moments

Menu

Gestión de usuario

- Acceder
- Registro

Acceso de usuario

Nick:

Contraseña:

Figura 153. Pantalla de Login.

En esta pantalla el usuario se identifica en el sistema mediante su Nick y su contraseña. En caso de ser correctos accede al sistema. En caso de error en uno de los credenciales, o de dejar un campo en blanco, esto se le notificará al usuario.

8.2.2 Manual del Usuario Normal

Una vez identificado en el sistema, lo primero que visualizará el usuario será la ventana principal o sección de 'Mis itinerarios'. En esta sección el usuario visualizará una lista de los itinerarios que tenga dados de alta en la plataforma, así como una lista de las solicitudes de amistad pendientes de resolver si las tuviera.

Como puede observarse en la *Figura 154* el usuario puede aceptar la petición de amistad, o rechazarla. En caso de lo primero el usuario solicitante pasará a estar en la lista de amigos, en caso de rechazar, se mostrará un mensaje de advertencia, que en caso de ser aceptado eliminará la petición.

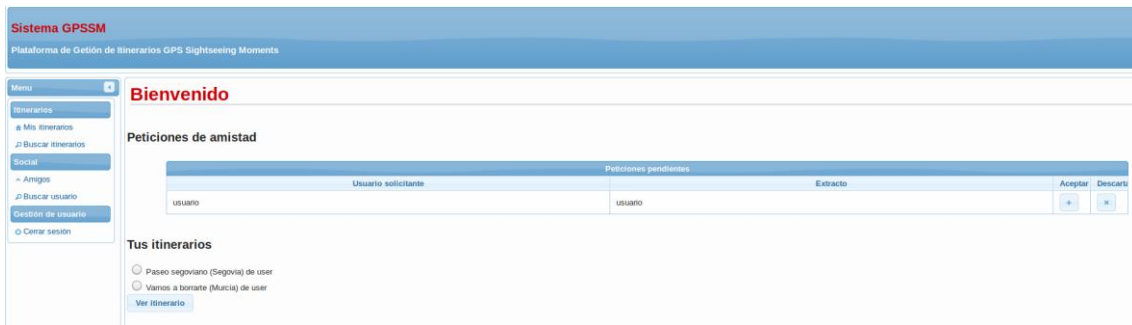


Figura 154. Pantalla principal o de Mis Itinerarios.

Cada itinerario presente en la lista puede ser consultado, para ello debe de ser seleccionado y el usuario debe pulsar sobre el botón 'Ver itinerario'.



Figura 155. Pantalla de consulta de itinerario.

Como puede verse en la *Figura 155*, la visualización del itinerario nos muestra un menú de opciones horizontal (Volver, editar y eliminar) y debajo un cajón con dos pestañas. En este cajón tenemos una pestaña con la información descriptiva del itinerario y su foto principal, y en la otra tenemos una galería de imágenes del itinerario, donde verlas en detalle con su título y descripción.

Abajo del todo puede ver un botón llamado 'Ver mapa', que pulsándole nos permite realizar la carga del mapa, con el recorrido del itinerario y la localización de las imágenes tomadas durante el recorrido Este resultado puede apreciarse en la *Figura 156*. Desplazando el ratón encima de los ítems se mostrará. un pequeño recuadro de texto con el título de este (inicio, fin o título de la foto).

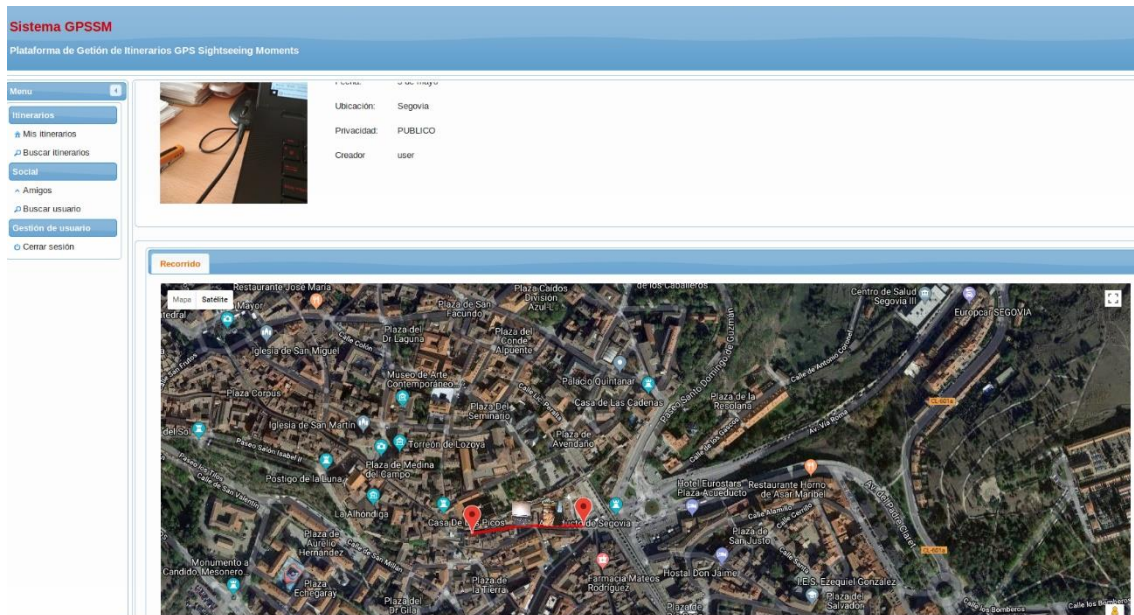


Figura 156. Pantalla con mapa del itinerario.

Si pulsamos sobre 'Editar', del menú horizontal, se nos abrirá la ventana que podemos observar en la Figura 157. En esta ventana podemos observar dos tablas. La primera con los atributos del itinerario, y la segunda con los atributos de cada punto multimedia perteneciente al itinerario. En ambas tablas podemos observar a la derecha del todo un lápiz. Pulsando en el podemos editar la información que es editable de cada fila. Una vez escritos los cambios podemos validarlos pulsando sobre '✓', o en '✗' para descartarlos.



Figura 157. Pantalla de edición de itinerario.

Es importante reseñar que no todos los atributos pueden editarse, algunos están restringidos, como la ubicación del itinerario o las coordenadas de los puntos. En caso de seleccionarse 'Eliminar', en el menú horizontal, se pedirá una confirmación por parte del usuario antes de eliminar el itinerario.

La opción de buscar itinerarios, que podemos ver en la Figura 158, nos permite mediante un formulario con dos campos realizar la búsqueda de itinerarios públicos. Esta búsqueda puede

realizarse por el título, la ubicación, o ambas. Los resultados pueden consultarse seleccionándolos, obteniendo una pantalla igual que a la de la consulta de un itinerario propio, pero sin la posibilidad de editar o borrar.



Figura 158. Pantalla de búsqueda de itinerarios.

En el apartado social, el usuario puede ver su lista de amistades seleccionando 'Amigos' en el menú izquierdo. Como puede verse en la Figura 159, se muestra una lista de relaciones de amistad y dos botones al lado derecho de cada fila. El botón de la derecha nos permite eliminar la relación de amistad.

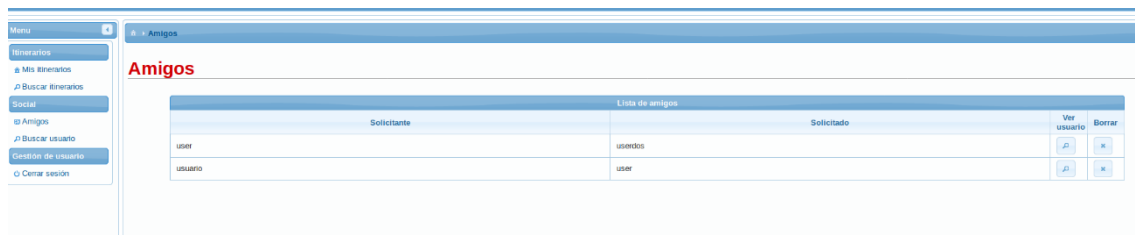


Figura 159. Pantalla de amistades.

El botón izquierdo, en cambio, nos permite visualizar el perfil del usuario con el que mantenemos la relación de amistad. Permitiendo visualizar su información en una pestaña, y en otra sus itinerarios públicos, y los visibles solo para amigos, como puede verse en la Figura 160.



Figura 160. Pantalla perfil de usuario.

En la opción del menú que está situada debajo, llamada 'Buscar usuario', es posible realizar la búsqueda de usuarios por su Nick (*ver Figura 161*), de forma que podremos consultar su perfil e itinerarios públicos. Para ello cada resultado de búsqueda tiene a su derecha un botón, con el símbolo de una lupa.

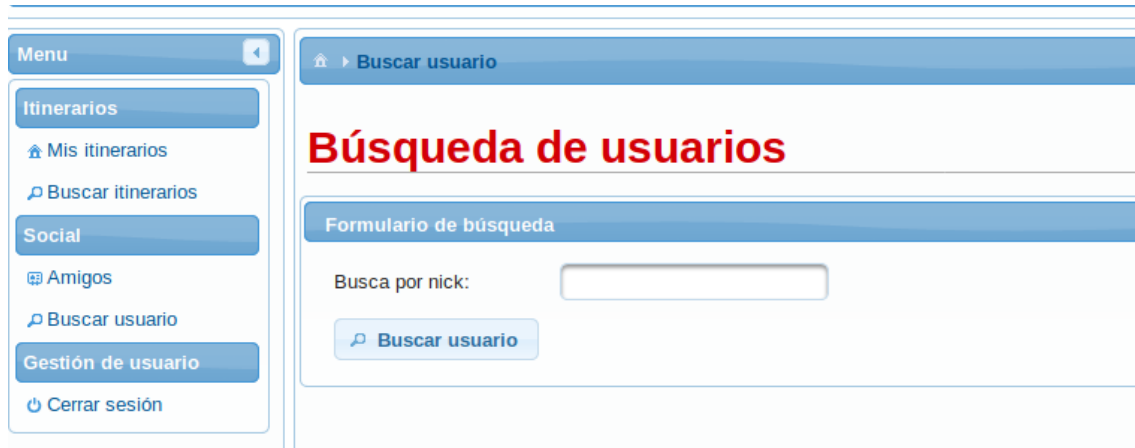


Figura 161. Pantalla de búsqueda de usuarios.

El resultado será una ventana para visualizar el perfil de usuario igual que la que obtenemos para ver el perfil de nuestros amigos, pero con la posibilidad de pulsar el botón 'Solicitar amistad' para enviarle una petición de amistad, como puede verse en la *Figura 162*.



Figura 162. Pantalla de Ver perfil de usuario buscado.

8.2.3 Manual del Usuario Gestor

Los usuarios 'Gestor' tienen la funcionalidad extra de poder gestionar los puntos de realidad aumentada. Accediendo a ella mediante el ítem 'Gestionar puntos' del menú izquierdo, se accede a la pantalla que podemos observar en la *Figura 163*. En esta pantalla se muestra una lista de puntos que están incluidos en la plataforma.







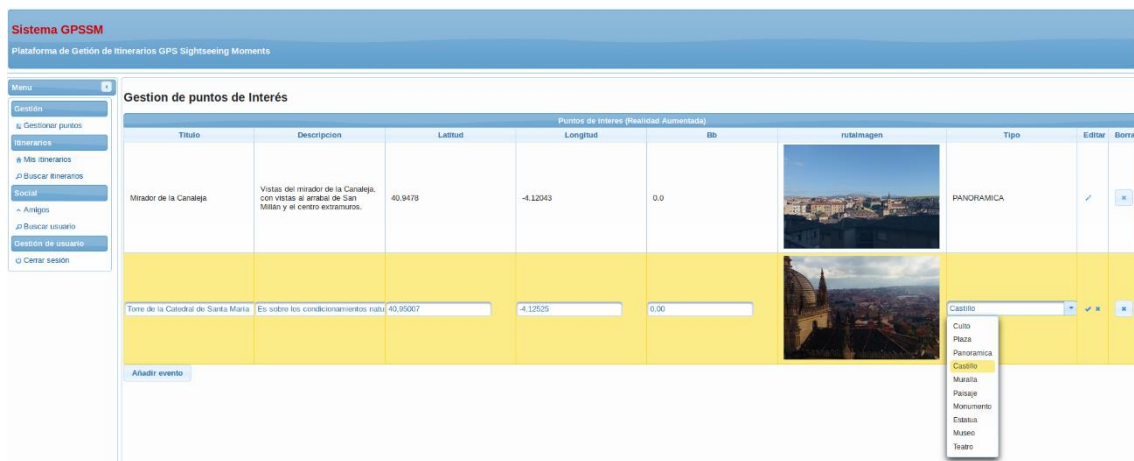






Sistema GPSSM								
Plataforma de Gestión de Itinerarios GPS Sightseeing Moments								
Gestion de puntos de Interés								
Puntos de Interés (Realidad Aumentada)								
Titulo	Descripcion	Latitud	Longitud	Bs	rutainagen	Tipo	Editar	Borrar
Mirador de la Canaleja	Vistas del mirador de la Canaleja, con vistas al arsenal de San Millán y el centro extramuros.	40.9478	-4.12043	0.0		PANORAMICA		
Torre de la Catedral de Santa María	Es sobre los condicionamientos naturales de sacros, escarpes y vaguadas, de cuevas y desfiladeros, como se ha ido constituyendo históricamente la ciudad.	40.95007	-4.12525	0.0		PANORAMICA		

Figura 163. Pantalla de gestión de puntos de interés.

Al lado de cada punto podemos observar un lapicero, que nos permite editar la información descriptiva de los puntos. A la derecha del lapicero, podemos encontrar un botón con el símbolo 'X', que permite eliminar el punto después de confirmar un dialogo emergente.



Sistema GPSSM								
Plataforma de Gestión de Itinerarios GPS Sightseeing Moments								
Gestion de puntos de Interés								
Puntos de Interés (Realidad Aumentada)								
Titulo	Descripcion	Latitud	Longitud	Bs	rutainagen	Tipo	Editar	Borrar
Mirador de la Canaleja	Vistas del mirador de la Canaleja, con vistas al arsenal de San Millán y el centro extramuros.	40.9478	-4.12043	0.0		PANORAMICA		
Torre de la Catedral de Santa María	Es sobre los condicionamientos natu	40.95007	-4.12525	0.00		Castillo		

- Castillo
- Culto
- Plaza
- Panoramica
- Castillo
- Marina
- Paisaje
- Monumento
- Estadua
- Museo
- Teatro

Figura 164. Pantalla de edición de puntos de interés.

Para añadir un nuevo punto, el usuario gestor debe de pulsar el botón de añadir situado en la parte inferior izquierda de la tabla de resultados. Al pulsarle, se muestra al usuario la pantalla de la *Figura 165*.

Sistema GPSSM
Plataforma de Gestión de Itinerarios GPS Sightseeing Moments

Menu

- Gestión
 - Gestionar puntos
- Itinerarios
 - Mis itinerarios
 - Buscar itinerarios
- Social
 - Amigos
 - Buscar usuario
- Gestión de usuario
 - Cerrar sesión

Nuevo punto de interés (Realidad aumentada)

+ Choose Upload Cancel

torre catedral.png 912.4 KB

Título: * Je la Catedral de Santa Maria

Descripción: * ndo históricamente la ciudad.

Tipo: Panorámica

Latitud: * 40,95007

Longitud: * -4,12525

BB: 0,00

Enviar

Figura 165. Pantalla de creación de un punto de interés.

Para añadir un punto de interés debemos de añadir una imagen. Para ello el usuario debe de pulsar en el botón 'Choose', y una vez ha elegido en el cuadro de exploración la imagen, pulsar en el botón 'Upload' para subir la imagen.

Una vez subida la imagen, al usuario se le mostrará un mensaje de 'Sucesful', como puede apreciarse en la Figura 167. Después para la creación del punto, el usuario solo tiene que rellenar todos los campos y pulsar sobre el botón 'Enviar'.

Sistema GPSSM
Plataforma de Gestión de Itinerarios GPS Sightseeing Moments

Menu

- Gestión
 - Gestionar puntos
- Itinerarios
 - Mis itinerarios
 - Buscar itinerarios
- Social
 - Amigos
 - Buscar usuario
- Gestión de usuario
 - Cerrar sesión

Nuevo punto de interés (Realidad aumentada)

+ Choose Upload Cancel

Successful

Título: * Mirador de la Canaleja

Descripción: * Millán y el centro extramuros.

Tipo: Panorámica

Latitud: * 40,94781

Longitud: * -4,12043

BB: 0,00

Enviar

Figura 166. Pantalla de creación de un punto de interés II.

Una vez creado el sistema redirigirá al listado de puntos de interés, donde aparecerá el nuevo punto.

Capítulo 9

Conclusiones

En esta sección se abordan las conclusiones tras el desarrollo del proyecto, una reflexión sobre lo aprendido y una mención a las posibles futuras líneas de trabajo identificadas.

La idea del proyecto GPS Sightseeing Moments surgió inicialmente durante el curso 2016/2017 en una clase de principios de curso en la asignatura 'Multimedia', cuando mi compañero Pablo Hernando García y nuestro tutor Miguel Ángel Martínez debatíamos sobre posibles ideas para desarrollar una práctica final que sería conjunta con la asignatura de 'Plataformas Software Móviles'. Debido a mi gran afición a los viajes y a fotografiar todo aquello que descubría en estos, y tras comentarlo con Pablo y recibir su apoyo y aportación a la idea, propusimos la idea construir una aplicación móvil que permitiría recordar los viajes de la gente con su recorrido y sus fotos agrupadas. Así desarrollamos un pequeño prototipo que guardaba esta información y la representaba de forma local.

Tras la gran satisfacción que obtuvimos con este pequeño prototipo decidimos llevar a cabo, animados por nuestros tutores Miguel Ángel y Aníbal, el desarrollo de esta herramienta como Trabajo de Fin de Grado, convirtiéndola en una red social que permitiera compartir la información de nuestros viajes. La amplia investigación del mercado nos afianzó en nuestra idea y nos permitió ver los factores diferenciadores que ofrecía nuestra propuesta como se ha expuesto en las conclusiones del estado del arte.

El desarrollo de este proyecto ha supuesto una dura prueba de resistencia. Ya que no resulta fácil ser constante en un proyecto en el que debes de ser autónomo, y compaginarlo con actividad laboral y otras responsabilidades. Pero, por otro lado, esta etapa de trabajo ha permitido poder extender una gran relación de amistad, y sobre todo de trabajo en equipo con mi compañero Pablo. Sin duda, trabajar en equipo es una de las cualidades que considero más importantes hoy en día, y con este proyecto creo que los dos la hemos consolidado aún más en nosotros mismos.

9.1 Conocimientos adquiridos

Después de tomar la decisión de llevar a cabo el proyecto como trabajo de fin de grado, fue necesario decidir que parte desarrollaría cada uno (plataforma empresarial o plataforma móvil). Mi elección por la plataforma empresarial se debió a que, tras cursar la asignatura de 'Plataformas Software Empresariales' me quedé con muchas ganas de ampliar los conocimientos adquiridos. Ya que, aunque la duración de la asignatura es demasiado breve para la complejidad de Java EE, me resultó motivadora despertando en mí las ganas y la inquietud de probar la implementación un sistema de servidor que se comunicara con

aplicaciones móviles, para así integrar los conocimientos de PSE y 'Plataformas Software Móviles'. Durante el desarrollo de la plataforma me vi obligado a recordar todo lo aprendido en la asignatura de PSE y a ampliar conocimientos, así como a darme cuenta de los fallos cometidos por la inexperiencia, tanto en la especificación de requisitos como en el desarrollo.

Una de las grandes lecciones aprendidas durante el desarrollo fue la creación de los servicios REST. Aunque los IDE nos pueden generar unos métodos genéricos, estos solo nos proporcionan el equivalente a consultas básicas (obtener todos, obtener un elemento en concreto, editar un elemento en concreto, o eliminarlo). Para poder realizar consultas relacionales, controlar las transacciones, realizar búsquedas, controlar códigos de respuesta, enviar respuestas personalizadas al cliente... he necesitado un aprendizaje sobre las librerías EclipseLink, CriteriaQuery, UnitOfWork, Response, etc. Pero, sobre todo, he necesitado comprender el funcionamiento de estos servicios y sus particularidades, como las utilidades de cada tipo de petición (POST, GET, PUT, DELETE), los tipos de 'Content-type' o los tipos de parámetros (FormParam, PathParam) que pueden ser usados.

Por otra parte, he profundizado mis conocimientos sobre el tratamiento de Json, ya que anteriormente solo había tratado con objetos sencillos expresados con este formato. En este proyecto debido a las relaciones entre entidades he tenido que aprender a generar y parsear cadenas de Json que podían contener en su interior varios objetos anidados.

Este proyecto también me ha permitido conocer los objetos definidos en las librerías de Primefaces, y conocer sus limitaciones. He usado multitud de elementos nuevos como GMap o fileUpload. Pero también he aprendido que al tratarse de un Framework que resulta opaco al desarrollador, los problemas que pueden surgir al combinar entre sí elementos con llamadas de Ajax o Javascript (como fileUpload y dataTable con edición de filas) no pueden ser resueltos por el desarrollador, y tienen que buscarse alternativas o desarrollarse sin el uso del Framework.

Por último, es importante reconocer que este proyecto ha permitido integrar los conocimientos de las que han sido mis asignaturas favoritas en la carrera 'Modelado de Software', 'Plataformas Software Móviles', 'Plataformas Software Empresariales', 'Administración de Bases de Datos', 'Programación y Estructura de Datos' y 'Programación Orientada a Objetos', así como con otras asignaturas como 'Gestión de Proyectos de Tecnologías de la Información' o 'Proceso de Desarrollo del Software'.

9.2 Líneas de trabajo para el futuro

El proyecto GPSSM, limitado inicialmente en características para cumplir los tiempos de entrega de los trabajos de fin de grados, posee un gran potencial debido a su enfoque, no presente en el mercado actualmente, sobre la experiencia de descubrimiento turístico y la posibilidad de compartirlo con nuestros amigos. Pero para su lanzamiento, es preciso potenciar sus factores diferenciadores, así como buscar una financiación a la viabilidad del proyecto.

A continuación, se pretende plasmar algunas de las mejoras identificadas para mejorar en el futuro en el proyecto GPSSM, concretamente en esta parte de la plataforma software empresarial. Son las siguientes:

- Inclusión de un nuevo tipo de punto de interés sobre establecimientos comerciales buscando una rentabilidad económica, es decir, que ciertos establecimientos paguen una cantidad de dinero periódica por ser promocionados en la plataforma cuando se están realizando itinerarios. Si bien estos serían mostrados en el dispositivo móvil, la gestión de estos puntos se realizaría desde la PSE.
- Migración a una base de datos especial, buscando las ventajas de estas para realizar cálculos espaciales y poder realizar el cálculo de estadísticas e información relevante sobre los recorridos.
- Posibilidad de compartir itinerarios para usuarios no registrados, tratándose solo de itinerarios 'Públicos', buscando una mayor integración en redes sociales para así tener una mayor promoción de la herramienta.
- Permitir borrado de cuenta y modificación de perfil desde la PSE, que no ha sido incluido en esta versión para evitar añadir extensión al desarrollo de la plataforma móvil debido a la limitación de tiempo.
- Permitir generar rutas desde la PSE, a través de ficheros tipo KMZ, KML, o ficheros en formato GeoJSON. Así como la exportación en estos formatos de los itinerarios creados en GPSSM.

Bibliografía

Core J2EE Patterns (paperback): Best Practices and Design Strategies

Alur, Deepak / Crupi, John / Malks, Dan. (10 Jun. 2003) *Prentice Hall Core*. [Libro].

Patterns of Enterprise Application Architecture (The Addison-Wesley Signature Series)

Martin Fowler (1 nov 2002). *The Addison-Wesley Signature Series*. [Libro].

Plataformas Software Empresariales (Curso 2017/2018)

Bregón, A. (2017). *Plataformas Software Empresariales*. [Presentación] Disponible: <https://campusvirtual.uva.es> [Último acceso: 03 Jul. 2018].

Plataformas Software Empresariales (Curso 2016/2017)

Bregón, A. (2016). *Plataformas Software Empresariales*. [Presentación] Disponible: <https://cursoanterior.campusvirtual.uva.es> [Último acceso: 29 Jun. 2018].

Plataformas Software Móviles (Curso 2016/2017)

Bregón, A. (2016). *Plataformas Software Móviles*. [Presentación] Disponible: <https://cursoanterior.campusvirtual.uva.es> [Último acceso: 29 Jun. 2018].

Administración de Bases de Datos (Curso 2016/2017)

Martínez, MA. (2016). *Administración de Bases de Datos*. [Presentación] Disponible: <https://cursoanterior.campusvirtual.uva.es> [Último acceso: 13 Abril. 2018].

PrimeFaces Showcase

Primefaces.org. (2018). *PrimeFaces Showcase*. [Web] Disponible: <https://www.primefaces.org/showcase> [Último acceso 20 Jun. 2018].

Stack Overflow - Where Developers Learn, Share, & Build Careers

Stack Overflow. (2018). Stack Overflow - Where Developers Learn [Web] Disponible: <https://stackoverflow.com> [Último acceso 24 Jun. 2018].

Prime Community Forum

Forum.primefaces.org. (2018). Prime Community Forum. [Web] Disponible: <https://forum.primefaces.org> [Último acceso 20 Jun. 2018].

GitHub - Build software better, together

GitHub. (2018). Build software better, together. [Web] Disponible: <https://github.com/> [Último acceso 27 Mayo. 2018].

ANEXOS

ANEXO I

Glosario

- **API:** Se define como interfaz de programación de aplicaciones. Se trata de una biblioteca con métodos, funciones y rutinas de programación que facilitan el desarrollo dentro de un determinado entorno.
- **BD:** Abreviatura de Base de Datos.
- **GeoJSON:** Formato estándar abierto para representar elementos geográficos, basado en JSON (*ver más adelante*).
- **JAAS:** Java Authentication and Authorization Service. Se trata de una implementación de seguridad y autenticación para aplicaciones Java.
- **Java EE:** Es una plataforma de programación en lenguaje java que permite implementar arquitecturas de N capas, con componentes modulares, que es ejecutada sobre un servidor de aplicaciones.
- **JSF (Java Server Faces):** Es un Framework Java para el desarrollo de interfaces de usuario para aplicaciones JavaEE.
- **JSON:** Se trata de un de un formato de texto ligero para el intercambio de información.
- **KLM:** Adaptación del lenguaje XML para la representación de datos geográficos en tres dimensiones.
- **KMZ:** Formato comprimido de KML que soporta la inclusión de contenido de imágenes.
- **Primefaces:** Es una biblioteca complementaria para el desarrollo con JSF con soporte para Ajax.
- **PSE:** Abreviatura del nombre de la asignatura Plataformas Software Empresariales.
- **PSM:** Abreviatura del nombre de la asignatura Plataformas Software Móviles.
- **REST:** Arquitectura software para el intercambio de información a través de la 'World Wide Web' a través del protocolo HTTP.

ANEXO II

Casos de Uso

En este anexo se incluyen los casos de uso que no han sido incluidos en el *Punto 4.2.2*. Los casos de uso se organizan por actores (*ver actores en punto 4.1 del documento y listado en punto 4.2.1*).

Usuario General Invitado

ID y NOMBRE:	UGI1- LOGIN
Versión	1.0
Actor principal	Usuario General Invitado
Actores secundarios	<i>No contemplado</i>
Descripción	Un usuario desea identificarse en la plataforma.
Precondición	PRE-1: El usuario debe de estar registrado en el sistema
Secuencia WEB	1 – El usuario introduce sus credenciales de usuario. 2 – El sistema comprueba con la base de datos las credenciales del usuario (EXCP-1).
Secuencia PSE	1 – El actor PSE realiza una petición con el Nick de usuario. 2 – El sistema una respuesta con la información del usuario con el Nick solicitado.
Postcondición WEB	POST-1: El sistema abre la sesión del usuario en el sistema
Postcondición PSM	POST-2: El sistema envía la respuesta a la petición.
Excepciones	EXCP-1: Si las credenciales de usuario no son correctas, el sistema muestra un mensaje de error y el caso de uso vuelve al paso 1. EXCP-2: Si se recibe un Nick de un usuario no existente, el sistema devuelve un código de respuesta 404.
Prioridad	Alta
Dependencias	DEP-1: Requisitos de información RI.EU.01, RI.EU.04 DEP-2: Regla de negocio RN.25. DEP-3: Regla de negocio RN.26.
Comentarios	Este acceso se realiza desde la interfaz web de la PSE o plataforma móvil.

Tabla 88. Caso de uso UGI1.

ID y NOMBRE:	UGI2 – REGISTRAR USUARIO
Versión	1.0
Actor principal	Usuario General Invitado
Actores secundarios	<i>No contemplado</i>
Descripción	Un usuario desea registrarse en la plataforma.
Precondición	<i>No contemplada</i>
Secuencia WEB	1- El usuario introduce los atributos del perfil de usuario. 2 – El sistema recoge y valida la información introducida (EXCP-1) (EXCP-2). Después cifra la contraseña y guarda la información en la base de datos.
Secuencia PSM	1 – La plataforma móvil envía los datos de la entidad Usuario, correspondientes al nuevo usuario registrado. 2 – El sistema persiste el contenido en la base de datos y genera una respuesta http 204 (EXCP-2).
Postcondición	POST-1: El sistema notifica el resultado de la operación.
Excepciones	EXCP-1: Si alguno de los datos no supera la validación, se genera un mensaje de error y se reinicia el caso de uso. EXCP-2: Si el usuario ya existe se genera una respuesta advirtiéndolo al actor.
Prioridad	Alta
Dependencias	DEP-1: Requisitos de información RI.EU.01, RI.EU.02, RI.EU.03, RI.EU.04 DEP-2: Regla de negocio RN-02 DEP-3: Reglas de negocio RN.15, RN.17 y RN.18
Comentarios	La contraseña, en cumplimiento de DEP-4, se enviará desde la plataforma móvil ya codificada.

Tabla 89. Caso de uso UGI2.

Usuario General Registrado

ID y NOMBRE:	UGR1 – GESTIONAR ITINERARIOS PROPIOS
Versión	1.0
Actor principal	Usuario General Registrado
Actores secundarios	<i>No contemplado</i>
Descripción	La plataforma móvil solicita todos los itinerarios producidos por un usuario para su consulta y gestión.
Precondición	PRE-1: El usuario está logueado en el portal web o dispositivo móvil.
Secuencia WEB	1 – El usuario solicita el listado de sus itinerarios. 2 – El sistema devuelve la lista de itinerarios. 3 – El usuario selecciona un itinerario. 4 – El sistema consulta las posiciones y puntos multimedia del itinerario seleccionado y muestra la información por pantalla. 5 – El sistema ofrece las opciones de editar atributos del itinerario (secuencia A), editar información de un punto de interés (secuencia B), o eliminar el itinerario (secuencia C). 6 – El usuario indica si desea ejecutar una de las secuencias alternativas, o finaliza el caso de uso.

Secuencia PSM	<p>1 – El dispositivo móvil hace una petición sobre todos los itinerarios del usuario.</p> <p>2 – El sistema envía un JSON con la información de todos los itinerarios del usuario.</p> <p>2 – El dispositivo móvil solicita las posiciones del itinerario.</p> <p>3 – El sistema devuelve un JSON con todas las posiciones del itinerario.</p> <p>3 – El dispositivo móvil interpreta la información de los puntos multimedia y solicita las imágenes al sistema.</p> <p>4 – El sistema envía las imágenes al dispositivo móvil.</p> <p>5 – El dispositivo móvil puede indicar el seguimiento de una de las secuencias alternativas, o se finaliza el caso de uso.</p>						
Secuencia alternativa	<table border="1"> <tr> <td>A</td> <td>4a.1 – El sistema ejecuta el CU “UGR2 – EDITAR ATRIBUTOS”.</td> </tr> <tr> <td>B</td> <td>4b.1 – El sistema ejecuta el CU “UGR3 – EDITAR INFO PUNTO DE INTERES”.</td> </tr> <tr> <td>C</td> <td>4c.1 – El sistema ejecuta el CU “UGR4 – ELIMINAR ITINERARIO”.</td> </tr> </table>	A	4a.1 – El sistema ejecuta el CU “UGR2 – EDITAR ATRIBUTOS”.	B	4b.1 – El sistema ejecuta el CU “UGR3 – EDITAR INFO PUNTO DE INTERES”.	C	4c.1 – El sistema ejecuta el CU “UGR4 – ELIMINAR ITINERARIO”.
A	4a.1 – El sistema ejecuta el CU “UGR2 – EDITAR ATRIBUTOS”.						
B	4b.1 – El sistema ejecuta el CU “UGR3 – EDITAR INFO PUNTO DE INTERES”.						
C	4c.1 – El sistema ejecuta el CU “UGR4 – ELIMINAR ITINERARIO”.						
Postcondición	POST-1: El sistema notifica las operaciones realizadas						
Excepciones	<i>No contempladas</i>						
Prioridad	Alta						
Dependencias	DEP-1: Requisito de información RI.EU.01 DEP-2: Requisitos de información RI.EI.01, RI.EI.02, RI.EI.03, RI.EI.04, RI.EI.05,						
Comentarios							

Tabla 90. Caso de uso UGR1.

ID y NOMBRE:	UGR2 – EDITAR ATRIBUTOS
Versión	1.0
Actor principal	Usuario General Registrado
Actores secundarios	<i>No contemplado</i>
Descripción	El usuario realiza modificaciones sobre los atributos de un itinerario.
Precondición	PRE-1: El usuario está logueado en el portal web o plataforma móvil. PRE-2: El actor ha consultado un itinerario. PRE-3: actor es el propietario del itinerario.
Secuencia normal	1 – El actor comunica al sistema los cambios en los atributos. 2 – El sistema valida los nuevos registros para los atributos (EXCP-1).
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema registra en la base de datos las modificaciones.
Excepciones	EXCP-1: Si alguno de los datos no supera la validación, se transmite un mensaje de error y se reinicia el caso de uso.
Prioridad	Alta
Dependencias	DEP-1: Requisitos de información RI.EI.03, RI.EI.04, RI.EI.05
Comentarios	

Tabla 91. Caso de uso UGR2.

ID y NOMBRE:	UGR3 – EDITAR INFO PUNTOS MULTIMEDIA
Versión	1.0
Actor principal	Usuario General Registrado
Actores secundarios	<i>No contemplado</i>
Descripción	El usuario realiza modificaciones sobre algún punto multimedia de un determinado itinerario.
Precondición	PRE-1: El usuario está logueado en el portal web o plataforma móvil. PRE-2: El actor ha consultado un itinerario. PRE-3: EL actor es el propietario del itinerario.
Secuencia normal	1 – El usuario comunica al sistema el cambio en atributos de un punto multimedia. 2 – La plataforma valida los registros recibidos (EXCP-1).
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema registra en la base de datos las modificaciones sobre el punto multimedia.
Excepciones	EXCP-1: Si alguno de los datos no supera la validación, se transmite un mensaje de error y se reinicia el caso de uso.
Prioridad	Alta
Dependencias	DEP-1: Requisitos de información RI.EI.01 DEP-2: Requisitos de información RI.EPM.01, RI.EPM.02
Comentarios	

Tabla 92. Caso de uso UGR3.

ID y NOMBRE:	UGR4 – ELIMINAR ITINERARIO
Versión	1.0
Actor principal	Usuario General Registrado
Actores secundarios	<i>No contemplado</i>
Descripción	El usuario realiza una petición para eliminar un itinerario.
Precondición	PRE-1: El usuario está logueado en el portal web o plataforma móvil. PRE-2: El actor ha consultado un itinerario. PRE-3: EL actor es el propietario del itinerario.
Secuencia normal	1 – El actor realiza una solicitud para el borrado del itinerario. 2 – El sistema recibe el identificador del itinerario para identificar el itinerario a borrar. 3- Si la petición se produjo desde la plataforma web se sigue la secuencia alternativa, si no se ejecutan el flujo de postcondición A
Secuencia alternativa	3.1 – Se requiere confirmación al usuario para eliminar el usuario. 3.2 – Si el usuario confirma el borrado se sigue flujo de postcondición A. Si no se sigue flujo de Postcondición B.
Postcondición A	POST-1: El sistema elimina de la base de datos el itinerario indicado. POST-2: El sistema realiza el borrado de las imágenes de puntos multimedia del almacenamiento interno.
Postcondición B	POST-1: El sistema no elimina el itinerario.
Excepciones	<i>No contempladas</i>
Prioridad	Alta
Dependencias	DEP-1: Requisitos de información RI.EI.01
Comentarios	

Tabla 93. Caso de uso UGR4.

ID y NOMBRE:	UGR5 – BUSCAR USUARIO
Versión	1.0
Actor principal	Usuario General Registrado
Actores secundarios	<i>No contemplado</i>
Descripción	El usuario realiza la búsqueda de un usuario
Precondición	PRE-1: El usuario está logueado en el portal web o plataforma móvil.
Secuencia normal	1 – El actor indica una clave de búsqueda. 2 – El sistema compara la clave de búsqueda con los Nick de usuario registrados en la plataforma.
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema devuelve el resultado de la consulta.
Excepciones	<i>No contempladas</i>
Prioridad	Alta
Dependencias	DEP-1: Requisitos de información RI.EI.01
Comentarios	El sistema ofrece resultados de usuarios con los que no existe una relación de amistad.

Tabla 94. Caso de uso UGR5.

ID y NOMBRE:	UGR6 – VER PERFIL
Versión	1.0
Actor principal	Usuario General Registrado
Actores secundarios	<i>No contemplado</i>
Descripción	El actor solicita la información de perfil de un usuario que ha sido encontrado en una búsqueda.
Precondición	PRE-1: El usuario está logueado en el portal web o plataforma móvil.
Secuencia normal	1 – El actor solicita ver el perfil de un usuario. 2 – El sistema recopila la información de perfil del usuario con el Nick solicitado.
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema devuelve la información de la entidad Usuario.
Excepciones	<i>No contempladas</i>
Prioridad	Alta
Dependencias	DEP-1: Requisitos de información RI.EI.01
Comentarios	La validación del borrado con el usuario final se realiza en la plataforma móvil.

Tabla 95. Caso de uso UGR6.

ID y NOMBRE:	UGR7 – SOLICITAR AMISTAD
Versión	1.0
Actor principal	Usuario General Registrado
Actores secundarios	<i>No contemplado</i>
Descripción	El sistema recoge una petición de amistad de un usuario hacia otro.
Precondición	PRE-1: El usuario está logueado en el portal web o plataforma móvil. PRE-2: No existe relación de amistad o solicitud sin resolver con el usuario que se desea solicitar la amistad.
Secuencia normal	1 – El actor envía la petición de amistad (DEP-1). 2 – EL sistema recibe la petición. (DEP-2).
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema ejecuta la sentencia de inserción en la base de datos.
Excepciones	<i>No contempladas</i>
Prioridad	Alta
Dependencias	DEP-1: Requisito de información RI.EU.01 del solicitante y el usuario que recibirá la solicitud. DEP-2: Requisito de información RI.EA.01
Comentarios	En DEP-2 el requisito de información tomará el valor 'PENDIENTE'.

Tabla 96. Caso de uso UGR7.

ID y NOMBRE:	UGR8 – CONSULTAR ITINERARIOS DE USUARIO
Versión	1.0
Actor principal	Usuario General Registrado
Actores secundarios	<i>No contemplado</i>
Descripción	El actor solicita los itinerarios que tiene un usuario.
Precondición	PRE-1: El usuario está logueado en el portal web o plataforma móvil.
Secuencia normal	1 – El actor solicita los itinerarios que ha producido un usuario (DEP-1). 2 – El sistema consulta los itinerarios del usuario cuyo valor de privacidad es 'PUBLICO'.
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema envía la respuesta.
Excepciones	EXCP-1: Si el usuario no tiene itinerarios creados el sistema establece como respuesta un valor nulo. EXCP-2: Si el usuario no tiene itinerarios que mostrar del grado de privacidad permitido, el sistema establece como respuesta un valor nulo.
Prioridad	Alta
Dependencias	DEP-1: Requisito de información RI.EU.01 DEP-2: Requisito de información RI.EU.04 DEP-3: Requisitos de información RI.EI.02, RI.EI.03, RI.EI.05 DEP-4: Reglas de negocio RN.11, RN.12, RN.13, RN.14
Comentarios	Solo se mostrarán os itinerarios públicos al tratarse de una extensión de ver perfil de un usuario buscado, de forma que no habrá una relación de amistad.

Tabla 97. Caso de uso UGR8.

ID y NOMBRE:	UGR9 – CONSULTAR LISTADO SOLICITUDES
Versión	1.0
Actor principal	Usuario General Registrado
Actores secundarios	<i>No contemplado</i>
Descripción	Un usuario consulta la lista de solicitudes de amistad sin resolver al acceder al sistema, una vez autenticado.
Precondición	PRE-1: El usuario está logueado en el portal web o plataforma móvil.
Secuencia normal	1 – El actor solicita las solicitudes de amistad pendientes. 2 – El sistema consulta las relaciones de amistad, cuyo estado se encuentra en 'Pendiente' (EXCP-1).
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema devuelve el resultado de la consulta.
Excepciones	EXCP-1: Si el usuario no tiene relaciones en estado 'Pendiente', se genera como resultado un valor nulo.
Prioridad	Alta
Dependencias	DEP-1: Requisito de información RI.EU.01 DEP-2: Requisito de información RI.EA.01
Comentarios	

Tabla 98. Caso de uso UGR9.

ID y NOMBRE:	UGR10 – RESOLVER SOLICITUD				
Versión	1.0				
Actor principal	Usuario General Registrado				
Actores secundarios	<i>No contemplado</i>				
Descripción	Un actor envía al sistema la resolución ante una petición de amistad.				
Precondición	PRE-1: El usuario está logueado en el portal web o plataforma móvil.				
Secuencia normal	1 – El actor envía la respuesta a una petición de amistad. 2 – El sistema recoge la respuesta. Si la respuesta es de confirmación de la amistad el sistema sigue la Secuencia A . En caso contrario sigue la Secuencia B .				
Secuencia alternativa	<table border="1"> <tbody> <tr> <td>A</td> <td>2a.1 – El sistema generará una sentencia de actualización en la base de datos para modificar el estado de la relación a 'Activa' (DEP-1).</td> </tr> <tr> <td>B</td> <td>2b.1 – El sistema generará una sentencia para eliminar de la base de datos la información de relación entre los dos usuarios.</td> </tr> </tbody> </table>	A	2a.1 – El sistema generará una sentencia de actualización en la base de datos para modificar el estado de la relación a 'Activa' (DEP-1).	B	2b.1 – El sistema generará una sentencia para eliminar de la base de datos la información de relación entre los dos usuarios.
A	2a.1 – El sistema generará una sentencia de actualización en la base de datos para modificar el estado de la relación a 'Activa' (DEP-1).				
B	2b.1 – El sistema generará una sentencia para eliminar de la base de datos la información de relación entre los dos usuarios.				
Postcondición	POST-1: El sistema ejecuta la sentencia en la base de datos.				
Excepciones	<i>No contempladas</i>				
Prioridad	Alta				
Dependencias	DEP-1: Requisito de información RI.EA.01				
Comentarios	Si se desestima la petición, se elimina la información de la base de datos para permitir una nueva petición en el futuro.				

Tabla 99. Caso de uso UGR10.

ID y NOMBRE:	UGR11 – CONSULTAR LISTADO DE AMISTADES
Versión	1.0
Actor principal	Usuario General Registrado
Actores secundarios	<i>No contemplado</i>
Descripción	El actor solicita el conjunto de amistades de un usuario de la plataforma.
Precondición	PRE-1: El usuario está logueado en el portal web o plataforma móvil.
Secuencia normal	1 – El actor solicita el conjunto de usuarios que tienen una relación de amistad con él. 2 – El sistema consulta en la base de datos la petición y genera como respuesta la información del conjunto de usuarios con relación de amistad (DEP-1) (EXCP-1).
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema devuelve el resultado de la consulta.
Excepciones	EXCP-1: Si el usuario no tiene relaciones de amistad activas, se genera como respuesta un valor nulo.
Prioridad	Media
Dependencias	DEP-1: Requisito de información RI.EA.01
Comentarios	El requisito contemplado en DEP-1 deberá encontrarse en estado 'ACTIVA'.

Tabla 100. Caso de uso UGR11.

ID y NOMBRE:	UGR12 – ELIMINAR AMISTAD
Versión	1.0
Actor principal	Usuario General Registrado
Actores secundarios	<i>No contemplado</i>
Descripción	El usuario dese eliminar una relación de amistad con otro usuario
Precondición	PRE-1: El usuario está logueado en el portal web o plataforma móvil.
Secuencia normal	1 – El actor solicita la eliminación de la relación de amistad entre dos usuarios. 2 – El sistema busca en la base de datos la relación indicada y la elimina, generando un valor de que la operación se ha realizado correctamente.
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema confirma el resultado de la operación.
Excepciones	<i>No contemplada</i>
Prioridad	Media
Dependencias	DEP-1: Requisito de información RI.EU.01
Comentarios	La excepción de este caso se contempla para el caso de que dos usuarios pretendan eliminar la relación de amistad que les une simultáneamente.

Tabla 101. Caso de uso UGR12.

ID y NOMBRE:	UGR13 – BUSCAR ITINERARIO	
Versión	1.0	
Actor principal	Usuario General Registrado	
Actores secundarios	<i>No contemplado</i>	
Descripción	El actor envía información de búsqueda para buscar coincidencias en los itinerarios registrados en la plataforma que son públicos.	
Precondición	PRE-1: El usuario está logueado en el portal web o plataforma móvil.	
Secuencia normal	1 – El actor envía los parámetros para realizar una búsqueda de itinerarios en la base de datos. 2 – El sistema recibe los parámetros de la búsqueda, e identifica cuales ha recibido. Si solo ha recibido palabras de búsqueda para el título Secuencia A . Si ha recibido información sobre ubicación el sistema sigue la Secuencia B . Si se recibe ambas cosas sigue la Secuencia C .	
Secuencia alternativa	A	2a.1 – El sistema realiza una búsqueda en la base de datos, sobre el título de los itinerarios, con las palabras de búsqueda 2a.2 – El sistema recopila, como resultado de la búsqueda, el conjunto los itinerarios encontrados (EXCP-1).
	B	2b.1 – El sistema realiza una búsqueda por el nombre de la ubicación recibida (EXCP-1). 2b.3 – El sistema recopila, como resultado de la búsqueda, el conjunto los itinerarios encontrados (EXCP-1).
	C	2c.1 – El sistema realiza una búsqueda en el contenido del título y la ubicación según los datos de búsqueda recibidos. (EXCP-1). 2c.3 – El sistema recopila, como resultado de la búsqueda, el conjunto los itinerarios encontrados (EXCP-1).
Postcondición	POST-1: El sistema comunica el resultado de la búsqueda.	
Excepciones	EXCP-1: Si la búsqueda o búsquedas no han obtenido resultados, se genera un valor nulo como respuesta a la consulta realizada al sistema.	
Prioridad	Alta	
Dependencias	DEP-1: Regla de negocio RN.04 DEP-2: Requisitos de información RI.EI.02, RI.EI.03.	
Comentarios	Los itinerarios pueden buscarse por título y/o ubicación.	

Tabla 102. Caso de uso UGR13.

ID y NOMBRE:	UGR14 – CONSULTAR ITINERARIO
Versión	1.0
Actor principal	Usuario General Registrado
Actores secundarios	<i>No contemplado</i>
Descripción	Un usuario solicita la información sobre un itinerario de otro usuario para poder ser consultado.
Precondición	PRE-1: El usuario está logueado en el portal web o plataforma móvil.
Secuencia normal	1 – El actor solicita la información almacenada sobre un itinerario. 2 – El sistema consulta todos los atributos sobre el itinerario (DEP-3). 3 – El sistema recopila el conjunto de todas las posiciones que forman parte del itinerario. 4 – Si el sistema tiene puntos multimedia se ejecuta la secuencia alternativa .
Secuencia alternativa	5 – El sistema recopila las imágenes que se corresponden a los puntos multimedia.
Postcondición	POST-1: El sistema comunica la información solicitada sobre el itinerario.
Excepciones	<i>No contempladas</i>
Prioridad	Alta
Dependencias	DEP-2: Requisito de información RI.EI.01 DEP-3: Requisitos de información RI.EI.02, RI.EI.03, RI.EI.04, RI.EI.05, RI.EI.06 DEP-4: Requisitos de información RI.EP.01, RI.EP.02, RI.EP.03, RI.EP.04, RI.EPM.01, RI.EPM.02, RI.EPM.03
Comentarios	La información sobre un itinerario incluye sus atributos, todos los puntos que lo conforman, y los puntos multimedia capturados.

Tabla 103. Caso de uso UGR14.

Usuario Web Registrado

ID y NOMBRE:	UWR1 – LOGOUT
Versión	1.0
Actor principal	UsuarioWeb Registrado
Actores secundarios	<i>No contemplado</i>
Descripción	Un usuario identificado en el sistema desea cerrar su sesión
Precondición	PRE-1: El usuario está logueado en el portal web.
Secuencia normal	1 – El usuario solicita destruir su sesión. 2 – El sistema redirige al usuario al portal de Login.
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema destruye la sesión del usuario.
Excepciones	<i>No contemplada</i>
Prioridad	Alta
Dependencias	<i>No contemplada</i>
Comentarios	

Tabla 104. Caso de uso UWR1.

Plataforma Móvil

ID y NOMBRE:	PM1 – ENVIAR ITINERARIO CREADO
Versión	1.0
Actor principal	Plataforma móvil
Actores secundarios	<i>No contemplado</i>
Descripción	El sistema recibe un itinerario creado en el dispositivo móvil.
Precondición	PRE-1: El usuario se comunica con el sistema logueado en la plataforma móvil de GPSSM.
Secuencia normal	1 – La plataforma envía los datos descriptivos del itinerario. 2 – El sistema valida los registros recibido (EXCP-1), y solicita la información del recorrido. 3 – La plataforma envía los puntos que forman el recorrido del itinerario. 4 – El sistema valida los registros recibido (EXCP-1), y solicita el envío de los puntos que contienen contenido multimedia. 5 – La plataforma envía la información relativa a los puntos multimedia. 7 – La plataforma dispara el caso de uso PM2 .
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema guarda en la base de datos toda la información del nuevo itinerario.
Excepciones	EXCP-1: Si alguno de los datos no supera la validación, se transmite un mensaje de error y se reinicia el caso de uso.
Prioridad	Alta
Dependencias	DEP-1: Requisitos de información RI.EI.02, RI.EI.03, RI.EI.04, RI.EP.02, RI.EP.03, RI.EP.04, RI.EPM.01, RI.EPM.02, RI.EPM.03 DEP-5: Regla de negocio RN.01, RN.09
Comentarios	En la base de datos no se almacena el contenido multimedia, si no que se almacena la ruta donde se almacena en el servidor.

Tabla 105. Caso de uso PM1.

ID y NOMBRE:	PM2 –ENVIAR CONTENIDO MULTIMEDIA
Versión	1.0
Actor principal	Plataforma móvil
Actores secundarios	<i>No contemplado</i>
Descripción	La plataforma móvil realiza la subida del contenido multimedia producido en la construcción del itinerario.
Precondición	PRE-1: El usuario se comunica con el sistema logueado en la plataforma móvil de GPSSM.
Secuencia normal	1 – La plataforma envía un contenido multimedia. 2 – El sistema almacena el archivo y obtiene la ruta del sistema de ficheros donde se encuentra el fichero (EXCP-1). 3 – Si la plataforma desea enviar más contenido vuelve al paso 1.
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema transmite un código de operación realizada
Excepciones	EXCP-1: Si falla la transmisión de un fichero se genera un código de error, se vuelve al paso 1.
Prioridad	Alta
Dependencias	DEP-1: Requisito de información RI.EPM.03
Comentarios	

Tabla 106. Caso de uso PM2.

ID y NOMBRE:	PM3 – CONSULTAR CUENTA PROPIA
Versión	1.0
Actor principal	Plataforma móvil
Actores secundarios	<i>No contemplado</i>
Descripción	La plataforma móvil solicita el perfil de un usuario para que este mismo pueda consultarlo.
Precondición	PRE-1: El usuario se comunica con el sistema logueado en la plataforma móvil de GPSSM.
Secuencia normal	1 – La plataforma solicita la información de perfil del usuario. 2 – El sistema consulta en la base de datos la información de perfil del usuario.
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema envía la información de perfil a la plataforma.
Excepciones	<i>No contempladas</i>
Prioridad	Alta
Dependencias	DEP-1: Requisitos de Información RI.EU.01, RI.EU.02, RI.EU.03, RI.EU.04, RI.EU.05
Comentarios	

Tabla 107. Caso de uso PM3.

ID y NOMBRE:	PM4 – MODIFICAR CUENTA
Versión	1.0
Actor principal	Plataforma móvil
Actores secundarios	<i>No contemplado</i>
Descripción	La plataforma móvil comunica las modificaciones que ha realizado un usuario en su perfil.
Precondición	PRE-1: El usuario se comunica con el sistema logueado en la plataforma móvil de GPSSM.
Secuencia normal	1 – La plataforma móvil envía las modificaciones sobre los atributos. 2 – El sistema realiza una validación sobre los datos (EXCP-1). 3 – El sistema escribe en la base de datos.
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema notifica un código de resultado de la operación.
Excepciones	EXCP-1: Si alguno de los datos no supera la validación, se genera un mensaje de error y finaliza el caso de uso.
Prioridad	Alta
Dependencias	DEP-1: Requisitos de información RI.EU.02, RI.EU.03, RI.EU.04 DEP-2: Reglas de negocio RN.17, RN.18
Comentarios	

Tabla 108. Caso de uso PM4.

ID y NOMBRE:	PM5 – ELIMINAR CUENTA
Versión	1.0
Actor principal	Plataforma móvil
Actores secundarios	<i>No contemplado</i>
Descripción	La plataforma comunica el borrado de la cuenta de un usuario.
Precondición	PRE-1: El usuario se comunica con el sistema logueado en la plataforma móvil de GPSSM.
Secuencia normal	1 – La plataforma comunica la orden de borrar un usuario (DEP-2). 2 – El sistema ejecuta la sentencia de borrado del usuario.
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema comunica a la plataforma la realización de la operación.
Excepciones	<i>No contempladas</i>
Prioridad	Alta
Dependencias	DEP-1: Regla de negocio RN.16 DEP-2: Requisito de información RI.EU.01
Comentarios	

Tabla 109. Caso de uso PM5.

ID y NOMBRE:	PM6 – OBTENER PUNTOS DE INTERÉS
Versión	1.0
Actor principal	Plataforma móvil
Actores secundarios	<i>No contemplado</i>
Descripción	La plataforma móvil solicita los puntos de interés almacenados en la plataforma
Precondición	PRE-1: El usuario se comunica con el sistema logueado en la plataforma móvil de GPSSM.
Secuencia normal	1 – La plataforma móvil solicita los puntos de interés. 2 – El sistema consulta los puntos de realidad aumentada en la base de datos.
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema envía a la plataforma el conjunto de todos los puntos.
Excepciones	<i>No contempladas</i>
Prioridad	Alta
Dependencias	DEP-1: Requisitos de información RI.EPOI.01, RI.EPOI.02, RI.EPOI.03 DEP-2: Requisitos de información RI.EPRA.01, RI.EPRA.02, RI.EPRA.03, RI.EPRA.04, RI.EPRA.05
Comentarios	Para evitar una conexión constante móvil/PSE para consultar los puntos de realidad aumentada cercanos, se descarga la información sobre todos.

Tabla 110. Caso de uso PM6.

ID y NOMBRE:	PM7 – OBTENER CONTENIDO MULTIMEDIA
Versión	1.0
Actor principal	Plataforma móvil
Actores secundarios	<i>No contemplado</i>
Descripción	La plataforma móvil solicita la imagen correspondiente a un punto
Precondición	PRE-1: El usuario se comunica con el sistema logueado en la plataforma móvil de GPSSM.
Secuencia normal	1 – La plataforma móvil solicita un nombre de imagen y su directorio. 2 – El sistema identifica si se le está solicitando la imagen de un punto multimedia o un punto de interés. 3 – El sistema busca la imagen en el almacenamiento interno (EXCP-1).
Secuencia alternativa	<i>No contemplada</i>
Postcondición	POST-1: El sistema envía una respuesta con la imagen.
Excepciones	<i>No contempladas</i>
Prioridad	Alta
Dependencias	DEP-1: Requisito de información RI.EPM.03 DEP-2: Requisito de información RI.EPRA.03
Comentarios	El directorio se refiere al id de itinerario o identificación de que se está consultando un punto de interés.

Tabla 111. Caso de uso PM7.

ANEXO III

Requisitos funcionales

En este anexo se incluyen los requisitos funcionales que no han sido incluidos en el *Punto 4.4*. Los requisitos funcionales se organizan por actores y los casos de uso realizados por estos (*ver actores en punto 4.1 del documento y listado de casos de uso por actor en punto 4.2.1*).

Usuario General Invitado

Caso de Uso	Descripción
UG11	Login
Requisito Funcional	Descripción
RF.21	El sistema comprueba si el Nick introducido el portal web existe en la base de datos.
RF.22	El sistema cifra la contraseña introducida por el usuario en el portal web para compararla, con la ya cifrada, que se encuentra en la base de datos almacenada.
RF.23	Si el Nick introducido en la web, esta almacenado en la base de datos, el sistema comprueba que la contraseña introducida coincide con la almacenada en la base de datos.
RF.24	Si alguno de las credenciales introducidos en la web no coincide, se muestra un mensaje de error por pantalla al usuario, y se le vuelven a solicitar las credenciales de acceso.
RF.25	Si las credenciales introducidas en la web son correctas, el sistema abre una sesión al usuario.
RF.26	El sistema recibe una petición de la plataforma móvil con credenciales de acceso.
RF.27	El sistema responde con la entidad del usuario las credenciales son correctas.
RF.28	El sistema responde con un mensaje de error las credenciales son correctas.

Tabla 112. Requisitos Funcionales en CU UG11.

Caso de Uso	Descripción
UGI2	Registrar usuario
Requisito Funcional	Descripción
RF.29	El sistema valida los datos de registro recibidos por el portal web.
RF.30	El sistema valida si las dos contraseñas introducidas son iguales.
RF.31	El sistema cifra la contraseña registrada a través de la web.
RF.32	Si los datos de registro introducidos por la web no superan la validación se le notifica al usuario por pantalla y se le requiere de nuevo la información.
RF.33	El sistema recibe de la plataforma móvil la entidad del usuario registrado.
RF.34	Si la validación de la entidad enviada por la PSM es correcta el sistema almacena en la base de datos el nuevo usuario.
RF.35	Si la validación de la entidad falla el sistema devuelve un error a la PSM.

Tabla 113. Requisitos Funcionales en CU UGI2.

Usuario General Registrado

Caso de Uso	Descripción
UGR1	Gestionar itinerarios propios
Requisito Funcional	Descripción
RF.36	Si se está accediendo desde el portal Web, el sistema comprueba que el usuario tiene sesión abierta.
RF.37	El sistema devuelve el listado de itinerarios cuyo creador es el Nick de usuario recibido.
RF.38	El sistema recibe la id del itinerario que se desea consultar.
RF.39	El sistema devuelve la información descriptiva del itinerario a consultar.
RF.40	El sistema devuelve la información de las posiciones del itinerario.
RF.41	El sistema devuelve cada imagen asociada a puntos multimedia.
RF.42	Si la consulta es a través del portal web, se muestran por pantalla los resultados.

Tabla 114. Requisitos Funcionales en CU UGR1.

Caso de Uso	Descripción
UGR2	Editar atributos
Requisito Funcional	Descripción
RF.36	(Ver en requisitos funcionales de UGR1).
RF.43	El sistema recibe todos los atributos e identifica los cambios.
RF.44	El sistema valida las modificaciones.
RF.45	Si alguno de los datos no supera la validación se genera un error y se vuelven a solicitar.
RF.46	Si se supera la validación, el sistema actualiza en la base de datos los atributos del itinerario.

Tabla 115. Requisitos Funcionales en CU UGR2.

Caso de Uso	Descripción
UGR3	Editar info puntos multimedia
Requisito Funcional	Descripción
RF.36	(Ver en requisitos funcionales de UGR1).
RF.47	El sistema recibe los atributos de un punto multimedia con una o varias modificaciones.
RF.44	(Ver en requisitos funcionales de UGR2).
RF.45	(Ver en requisitos funcionales de UGR2).
RF.48	Si se supera la validación, el sistema actualiza en la base de datos los atributos del punto multimedia.

Tabla 116. Requisitos Funcionales en CU UGR3.

Caso de Uso	Descripción
UGR4	Eliminar itinerario
Requisito Funcional	Descripción
RF.36	(Ver en requisitos funcionales de UGR1).
RF.49	El sistema recibe el identificador del itinerario a borrar.
RF.50	Si la petición procede del portal web, se pide confirmación al usuario.
RF.51	Si el usuario que solicita el borrado a través del portal web no confirma su eliminación se aborta el proceso.
RF.52	El sistema realiza el borrado del itinerario en la base de datos, eliminando en cascada toda la información relacionada en la base de datos.
RF.53	El sistema elimina del almacenamiento interno las imágenes de los puntos multimedia del itinerario.

Tabla 117. Requisitos Funcionales en CU UGR4.

Caso de Uso	Descripción
UGR5	Buscar usuario
Requisito Funcional	Descripción
RF.36	(Ver en requisitos funcionales de UGR1).
RF.54	El sistema recoge la cadena proporcionada por el actor y realiza una búsqueda en la base de datos sobre los Nick de usuario que contienen esa cadena.
RF.55	El sistema comunica los resultados mediante respuesta (web) o respuesta (PSM).

Tabla 118. Requisitos Funcionales en CU UGR5.

Caso de Uso	Descripción
UGR6	Ver perfil
Requisito Funcional	Descripción
RF.36	(Ver en requisitos funcionales de UGR1).
RF.56	El sistema recibe una consulta con un Nick de usuario.
RF.57	El sistema devuelve por pantalla (web) o por respuesta http(PSM), toda la información de la entidad usuario correspondiente al Nick recibido.

Tabla 119. Requisitos Funcionales en CU UGR6.

Caso de Uso	Descripción
UGR7	Solicitar amistad
Requisito Funcional	Descripción
RF.36	(Ver en requisitos funcionales de UGR1).
RF.59	El sistema recibe del actor el identificador del usuario solicitante, el del usuario solicitado, y registra en la base de datos una petición en estado 'PENDIENTE'.

Tabla 120. Requisitos Funcionales en CU UGR7.

Caso de Uso	Descripción
UGR8	Consultar itinerarios de usuario
Requisito Funcional	Descripción
RF.36	(Ver en requisitos funcionales de UGR1).
RF.60	El sistema recibe el identificador del usuario del cual se desea obtener los itinerarios.
RF.61	El sistema consulta en la base de datos los itinerarios del usuario recibido con privacidad en modo 'PUBLICO'.
RF.62	El sistema devuelve por pantalla (web) o por respuesta http (PSM), el listado de itinerarios públicos del usuario.
RF.63	Si la consulta no devuelve resultados, se le notifica al usuario por pantalla (acceso web), o se devuelve un valor nulo de respuesta.

Tabla 121. Requisitos Funcionales en CU UGR8.

Caso de Uso	Descripción
UGR9	Consultar listado de solicitudes
Requisito Funcional	Descripción
RF.36	(Ver en requisitos funcionales de UGR1).
RF.64	El sistema recibe el Nick del usuario del cual se requieren las solicitudes de amistad sin resolver.
RF.65	El sistema consulta la base de datos las solicitudes en estado 'Pendiente' para el usuario recibido.
RF.66	El sistema devuelve por pantalla (web) o por respuesta http (PSM), el listado de peticiones de amistad del usuario.
RF.63	(Ver en requisitos funcionales de UGR8).

Tabla 122. Requisitos Funcionales en CU UGR9.

Caso de Uso	Descripción
UGR10	Resolver solicitud
Requisito Funcional	Descripción
RF.36	(Ver en requisitos funcionales de UGR1).
RF.67	El sistema recibe el Nick de solicitante y solicitado, y la respuesta a la solicitud de amistad.
RF.68	Si la respuesta es afirmativa, se cambia el estado de la petición a 'ACTIVA' en la base de datos.
RF.69	Si la respuesta es negativa, se elimina la petición de la base de datos.

Tabla 123. Requisitos Funcionales en CU UGR10.

Caso de Uso	Descripción
UGR11	Consultar listado de amistades
Requisito Funcional	Descripción
RF.36	(Ver en requisitos funcionales de UGR1).
RF.70	El sistema recibe del actor un Nick de usuario.
RF.71	El sistema consulta en la base de datos las relaciones de amistad en estado 'ACTIVA' que contiene como solicitante o solicitado al Nick recibido.
RF.63	(Ver en requisitos funcionales de UGR8).
RF.72	El sistema devuelve por pantalla (web) o por respuesta http (PSM), el listado de amistades del usuario.

Tabla 124. Requisitos Funcionales en CU UGR11.

Caso de Uso	Descripción
UGR12	Eliminar amistad
Requisito Funcional	Descripción
RF.36	(Ver en requisitos funcionales de UGR1).
RF.73	El sistema recibe el Nick de solicitante y solicitado de una relación de amistad.
RF.74	El sistema elimina de la base de datos relación de amistad.
RF.75	El sistema confirma por pantalla (acceso web) o mediante una respuesta (PSM) el resultado de la operación.

Tabla 125. Requisitos Funcionales en CU UGR12.

Caso de Uso	Descripción
UGR13	Buscar itinerario
Requisito Funcional	Descripción
RF.36	(Ver en requisitos funcionales de UGR1).
RF.76	El sistema recibe los parámetros de búsqueda recibidos.
RF.77	Si solo ha recibido parámetro de título, el sistema consulta las coincidencias por título.
RF.78	Si solo ha recibido parámetro de ubicación, el sistema consulta las coincidencias por ubicación.
RF.79	Si el sistema ha recibido parámetros de título y ubicación, realiza una consulta buscando coincidencias en ambos atributos de la entidad itinerario.
RF.63	(Ver en requisitos funcionales de UGR8).
RF.80	El sistema devuelve por pantalla (web) o por respuesta http (PSM), el listado de itinerarios devuelto por la consulta.

Tabla 126. Requisitos Funcionales en CU UGR13.

Caso de Uso	Descripción
UGR14	Consultar itinerario
Requisito Funcional	Descripción
RF.36	(Ver en requisitos funcionales de UGR1).
RF.81	El sistema recibe el identificador del itinerario a consultar.
RF.82	El sistema devuelve los atributos del itinerario consultado.
RF.84	El sistema devuelve las posiciones del itinerario consultado.
RF.85	El sistema devuelve la imagen correspondiente a los puntos de realidad multimedia.
RF.86	Si la consulta es a través del portal web, se muestran por pantalla los resultados.

Tabla 127. Requisitos Funcionales en CU UGR14.

Usuario Web Registrado

Caso de Uso	Descripción
UWR1	Logout
Requisito Funcional	Descripción
RF.87	El sistema destruye la sesión del usuario.
RF.88	El sistema redirige al usuario al portal de login.

Tabla 128. Requisitos Funcionales en CU UWR1.

Plataforma Móvil

Caso de Uso	Descripción
PM1	Enviar itinerario creado
Requisito Funcional	Descripción
RF.89	El sistema recoge la información de la entidad itinerario.
RF.90	El sistema valida la información de la entidad itinerario.
RF.91	El sistema persiste la información de la entidad itinerario.
RF.92	El sistema recibe todas las posiciones del itinerario.
RF.93	El sistema mapea cada entidad de posición y las persiste en la base de datos.
RF.94	Si alguna entidad no supera la validación, se transmite un mensaje de error.
RF.95	Si la operación se realiza correctamente el sistema genera un código de operación realizada.

Tabla 129. Requisitos Funcionales en CU PM1.

Caso de Uso	Descripción
PM2	Enviar contenido multimedia
Requisito Funcional	Descripción
RF.96	El sistema extrae de la petición en nombre del archivo y el id de itinerario.
RF.97	El sistema genera la ruta a partir a partir de una ruta predefinida, el id de itinerario como directorio y el nombre de archivo recibido.
RF.98	El sistema guarda en el almacenamiento interno la imagen.
RF.99	Si falla la carga del archivo, el sistema responde con un código de error.
RF.95	(Ver en requisitos funcionales de PM1).

Tabla 130. Requisitos Funcionales en CU PM2.

Caso de Uso	Descripción
PM3	Consultar cuenta propia
Requisito Funcional	Descripción
RF.100	El sistema recibe el Nick de usuario
RF.101	El sistema envía una respuesta con los datos de la entidad usuario correspondientes con el Nick recibido.

Tabla 131. Requisitos Funcionales en CU PM3.

Caso de Uso	Descripción
PM4	Modificar cuenta
Requisito Funcional	Descripción
RF.102	El sistema recibe los atributos de la entidad usuario.
RF.103	El sistema valida los atributos.
RF.94	(Ver en requisitos funcionales de PM1).
RF.104	Si la clase entidad usuario se persiste correctamente el sistema retorna un código de operación realizada.

Tabla 132. Requisitos Funcionales en CU PM4.

Caso de Uso	Descripción
PM5	Eliminar cuenta
Requisito Funcional	Descripción
RF.105	El sistema recibe un identificador de usuario para eliminar una entidad de este tipo.
RF.106	El sistema elimina de la base de datos la entidad.
RF.107	El sistema devuelve un código indicando que la operación de borrado se ha realizado con éxito.

Tabla 133. Requisitos Funcionales en CU PM5.

Caso de Uso	Descripción
PM6	Obtener puntos de interés
Requisito Funcional	Descripción
RF.108	El sistema consulta todos los puntos de interés dados de alta en la base de datos.
RF.109	El sistema retorna una respuesta con todos los puntos.

Tabla 134. Requisitos Funcionales en CU PM6.

Caso de Uso	Descripción
PM7	Obtener contenido multimedia
Requisito Funcional	Descripción
RF.110	El sistema recibe el nombre de imagen e identificador de directorio
RF.111	El sistema identifica según el identificador de directorio si se trata de un punto multimedia o un punto de interés y obtiene la imagen.
RF.112	El sistema envía una respuesta con la imagen.

Tabla 135. Requisitos Funcionales en CU PM7.

ANEXO IV

Estructura del CD

El contenido del CD del proyecto está organizado en directorios para agrupar de una forma organizada su contenido.

Los directorios son los siguientes:

Directorio	Descripción	Ficheros
Despliegue	Contiene los ficheros necesarios para el despliegue del proyecto y el código fuente.	ddl.sql Glassfish-web.xml GPSSM.rar GPSSM-SHNAPSHOT-1.0.war
Diagramas	Contiene diagramas que por su gran tamaño no pueden ser visualizados en una sola página.	Arquitectura lógica.pdf ER.png Gantt.pdf Uml Clases GPSSM.rar
Memoria	Contiene el fichero de memoria del proyecto.	Memoria Plataforma de Gestion de Itinerarios GPSSM - DAVID BURGOS.pdf

Tabla 136. Contenido del CD.