



UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior (Jaén)

Trabajo Fin de Grado

**Análisis, diseño e
implementación de una
aplicación web de gestión de
autoescuelas: backend +
frontend Android**

Alumno: Alberto Carrillo Ortega

Tutores: Prof. D. Andrés Molina Aguilar
Prof. D. Ángel Luis García Fernández

Dpto: Informática

Septiembre, 2019



Universidad de Jaén
Escuela Politécnica Superior de Jaén
Departamento de Informática

Don Andrés Molina Aguilar y Don Ángel Luis García Fernández, tutores del Proyecto Fin de Grado titulado: “Análisis, diseño e implementación de una aplicación web de gestión de autoescuelas: backend + frontend Android ”, que presenta Alberto Carrillo Ortega, autorizan su presentación para defensa y evaluación en la Escuela Politécnica Superior de Jaén.

Jaén, Septiembre de 2019

El alumno:

Alberto Carrillo Ortega

Los tutores:

Andrés Molina Aguilar
Ángel Luis García Fernández

Agradecimientos

Querría empezar por agradecerle su labor educativa a el cuerpo docente en general, y más concretamente a los tutores de este proyecto, Andrés y Ángel, por tener la paciencia de lidiar con mis numerosos defectos y ayudarme cuando lo he necesitado.

También agradecer a la Autoescuela Zona L por trabajar con nosotros durante este curso educativo en nuestro pequeña aventura empresarial.

Y por último. Especiales agradecimientos a mi compañero Javier por ser un gran amigo y el culpable de que me metiese en este embrollo que es la ingeniería informática. De verdad, muchas gracias. ¡Pero no te confíes! Recuerda que quedan muchos experimentos por desarrollar y necesito ayudantes.

Ficha del Trabajo Fin de Título

Titulación	Grado en Ingeniería Informática
Modalidad	Proyecto de Ingeniería
Especialidad	Ingeniería del Software
Mención	Sin Mención
Idioma	Español
Tipo	Específico
TFG en equipo	Sí
Fecha de asignación	09/04/2019
Descripción corta	Se pretende implementar una aplicación para gestionar diversos aspectos relacionado con la actividad profesional y académica desarrollada en una autoescuela. Consistirá en un sistema con arquitectura cliente - servidor en la que se desarrollará una aplicación del lado del servidor y una aplicaciones del lado del cliente para equipos móviles (Android) 1. Obtención de requisitos, organización del proyecto y correcta documentación del mismo. 2. Análisis y diseño de la todos los componentes del sistema (aplicación del lado del servidor y cliente). 3. Implementación, control y corrección de errores de la aplicación del lado del servidor. 4. Implementación, control y corrección de errores de la aplicación cliente para equipos móviles (Android)

NORMAS APLICADAS EN ESTE DOCUMENTO	
LOCALES	
TFT-UJA:2017	Normativa de Trabajos Fin de Grado, Fin de Máster y otros Trabajos Fin de Título de la Universidad de Jaén (Normativa marco UJA aprobada en Consejo de Gobierno)
TFT-EPSJ:2017	Normativa sobre Trabajos Fin de Grado y Fin de Máster en la Escuela Politécnica Superior de Jaén (Normativa EPSJ aprobada en Junta de Escuela)
TFT-EPSJ	Criterios de evaluación y normas de estilo para TFG y TFM de la Escuela Politécnica Superior de Jaén
NACIONALES E INTERNACIONALES	
ISO 2145:1978	Documentación - Numeración de divisiones y subdivisiones en documentos escritos
UNE 50132:1994	Traducción de la ISO 2145

NORMAS UTILIZADAS COMO BASE O REFERENCIA	
NACIONALES	
UNE 157001:2014	Criterios generales para la elaboración formal de los documentos que constituyen un proyecto técnico
UNE 157801:2007	Criterios generales para la elaboración de proyectos de sistemas de información
GUÍA PMBOK	Guías y buenas prácticas para la gestión de proyectos proporcionada por el <i>Project Management Institute</i>
<i>Estas normas se han utilizado como base o referencia para la inclusión de algunos contenidos y definiciones sobre elaboración de proyectos, entendiendo como proyecto la documentación consensuada entre una empresa y un cliente, que da lugar al perfeccionamiento de un contrato para la elaboración de una obra o la prestación de un servicio. Por consiguiente, no debe esperarse la aplicación de estas normas en cuanto a la completitud de los contenidos ni a la organización de los mismos.</i>	

Sumario

1 ESPECIFICACIÓN DEL PROYECTO.....	16
1.1 Introducción.....	16
1.2 Objeto del proyecto.....	17
1.3 Antecedentes y estado del arte.....	18
1.3.1 WinAutoGest.....	18
1.3.2 Control L.....	19
1.3.3 Conclusión.....	19
1.4 Descripción de la situación actual.....	20
1.4.1 Descripción del entorno actual.....	20
1.4.2 Resumen de las principales deficiencias encontradas.....	21
1.5 Requisitos iniciales.....	21
1.6 Alcance.....	22
1.6.1 Hipótesis y Restricciones.....	23
1.7 Análisis de riesgos.....	23
1.8 Ciclo de vida del Proyecto y Organización.....	25
1.8.1 Elección del ciclo de vida del proyecto.....	25
1.8.2 Metodología de desarrollo.....	27
1.8.3 Reglas de Operativa del equipo.....	28
1.8.3.1 Resolución de problemas ligados al desarrollo del proyecto.....	28
1.8.3.2 Toma de decisiones.....	28
1.8.3.3 Resolución de conflictos internos del equipo.....	28
1.8.3.4 Reuniones.....	28
1.9 Material y métodos.....	29
1.9.1 Recursos Materiales.....	30
1.9.2 Herramientas Software.....	30
1.9.2.1 Apache NetBeans.....	30
1.9.2.2 Visual Paradigm.....	30
1.9.2.3 Android Studio.....	31
1.9.2.4 LibreOffice.....	31
1.9.2.5 Git y GitLab.....	31
1.9.3 Tecnologías Utilizadas.....	32
1.9.3.1 OpenJDK.....	32
1.9.3.2 Spring Framework.....	32
1.9.3.3 JavaFX.....	32
1.9.3.4 Retrofit y OkHttp.....	33
1.9.3.5 PostgreSQL.....	33
1.9.4 Métodos.....	34
1.9.4.1 El Manifiesto Ágil.....	34
1.9.4.2 PMBOK.....	34
1.10 Estimación de tamaño y esfuerzo.....	35
1.11 Presupuesto.....	40
1.11.1 Gastos derivados del personal.....	40
1.11.2 Materiales.....	42
1.11.2.1 Materiales Hardware.....	42
1.11.2.2 Materiales Software.....	42
1.11.3 Costes indirectos.....	43
1.11.4 Presupuesto final.....	43

1.12 Normas y referencias.....	44
1.12.1 Disposiciones legales y normas aplicadas.....	44
2 INICIO DEL PROYECTO.....	45
2.1 Planificación inicial del proyecto.....	45
2.2 Análisis inicial del sistema.....	47
2.3 Diseño Arquitectónico preliminar.....	48
2.3.1 Arquitectura del sistema.....	48
2.3.2 Servidor.....	51
2.3.3 Cliente Móvil.....	51
3 DESARROLLO DEL PROYECTO.....	53
3.1 Primera Iteración.....	53
3.1.1 Historias de usuario.....	53
3.1.2 Análisis.....	55
3.1.3 Planificación de la iteración.....	56
3.1.3.1 Proceso de documentación previa.....	57
3.1.4 Diseño.....	60
3.1.4.1 Servidor.....	60
3.1.4.2 Cliente Móvil.....	64
3.1.5 Resumen de progreso.....	67
3.1.6 Entrega y reunión final de la iteración.....	69
3.2 Segunda Iteración.....	70
3.2.1 Historias de usuario.....	70
3.2.2 Análisis.....	73
3.2.3 Planificación de la iteración.....	77
3.2.4 Diseño.....	79
3.2.4.1 Cliente Móvil.....	80
3.2.5 Resumen de progreso.....	81
3.2.6 Entrega y reunión final de la iteración.....	83
3.3 Tercera Iteración.....	84
3.3.1 Historias de usuario.....	84
3.3.2 Análisis.....	87
3.3.3 Planificación de la iteración.....	90
3.3.4 Diseño.....	93
3.3.4.1 Paginación.....	93
3.3.4.2 Servidor.....	95
3.3.4.3 Cliente Móvil.....	99
3.3.5 Resumen de progreso.....	101
3.3.6 Entrega y reunión final de la iteración.....	102
3.4 Cuarta Iteración.....	103
3.4.1 Historias de usuario.....	103
3.4.2 Análisis.....	104
3.4.3 Planificación de la iteración.....	106
3.4.4 Diseño.....	107
3.4.4.1 Servidor.....	108
3.4.4.2 Cliente Móvil.....	113
3.4.5 Resumen de progreso.....	115
3.4.6 Entrega y reunión final de Iteración.....	117
3.5 Quinta Iteración.....	118

3.5.1 Historias de usuario.....	118
3.5.2 Análisis.....	119
3.5.3 Planificación de la iteración.....	120
3.5.4 Diseño.....	121
3.5.4.1 Gestión de sesiones.....	121
3.5.4.2 Servidor.....	123
3.5.4.3 Cliente Móvil.....	125
3.5.5 Resumen de progreso.....	127
3.5.6 Entrega y reunión final de Iteración.....	129
4 RESULTADOS Y DISCUSIÓN.....	132
4.1 Resumen final del progreso.....	132
4.2 Entrega.....	134
5 CONCLUSIONES.....	135
6 ANEXOS.....	137
6.1 A1 – Diagramas y Grafos de navegación.....	137
6.2 A2 – Documentación relevante del proyecto.....	147
6.3 A3 – Instalación y configuración del sistemas.....	154
6.3.1 Instalación del servidor.....	154
6.3.1.1 Requisitos previos.....	154
6.3.1.2 Instalación.....	154
6.3.2 Instalación del Cliente Móvil.....	159
6.3.2.1 Requisitos previos.....	159
6.3.2.2 Instalación.....	159
6.4 A4 – Manuales de Usuario.....	160
6.4.1 Cliente Móvil.....	160
6.4.1.1 Introducción.....	161
6.4.1.2 Vistas y navegación.....	162
6.4.1.3 Gestión de Errores.....	176
6.5 A4 – Glosario y Abreviaturas.....	179
6.6 A6 – Guía Original del Trabajo de Fin de Grado.....	184
7 BIBLIOGRAFÍA Y FUENTES.....	185

Índice de figuras

Figura 1: Logo WinAutoGest.....	18
Figura 2: Logo Control L.....	19
Figura 3: Matriz de Probabilidad e Impacto (PMBOK).....	23
Figura 4: Diagrama WBS - Autoescuela Zona L.....	46
Figura 5: <i>Arquitectura Preliminar – Estructura de Capas y Particiones</i>	50
Figura 6: <i>Arquitectura Preliminar – Lógica de negocio</i>	51
Figura 7: <i>Arquitectura Preliminar – Cliente Móvil</i>	52
Figura 8: DA001 Añadir Alumno (H001).....	55
Figura 9: DA002 Modificar Alumno (H002).....	55
Figura 10: DA003 Borrar Alumno (H003).....	56
Figura 11: DA004 Búsqueda de alumnos por DNI (H004).....	56
Figura 12: <i>Diagrama de Clases – Alumnado, Lógica de negocio, Servidor</i>	61
Figura 13: <i>Diagrama de Clases – Iteración 1, Cliente Móvil</i>	66
Figura 14: Grafo Navegación – Iteración 1, Cliente Móvil.....	67
Figura 15: Resumen de Progreso – Iteración 1.....	68
Figura 16: Burndown Chart – Tareas Iteración 1.....	68
Figura 17: Burndown Chart – Iteración 1.....	69
Figura 18: DA005 Añadir Expediente (H005).....	73
Figura 19: DA006 Borrar Expediente (H006).....	74
Figura 20: DA007 Modificar Expediente (H007).....	74
Figura 21: DA008 Añadir Examen (H008).....	75
Figura 22: DA009 Borrar Examen (H009).....	75
Figura 23: DA010 Listar Expedientes (H010).....	76
Figura 24: DA011 Listar Exámenes (H011).....	76
Figura 25: Cliente Móvil: Grafo Navegación 2º Iteración.....	81
Figura 26: Resumen de Progreso – Iteración 2.....	82
Figura 27: Burndown Chart – Tareas Iteración 2.....	82
Figura 28: Burndown Chart – Iteración 2.....	83
Figura 29: DA012 Añadir Vehículo (H012).....	87
Figura 30: D013 Modificar Vehículo(H013).....	88
Figura 31: D014 Borrar Vehículo (H014).....	88
Figura 32: D015 Añadir Operación de Mantenimiento(H015).....	89
Figura 33: D016 Borrar Operación de Mantenimiento (H016).....	89
Figura 34: D017 Listar Vehículos (H017).....	90
Figura 35: D018 Listar <i>Operaciones de Mantenimiento</i> (H018).....	90
Figura 36: Diagrama de Clases Vehículos - Capa de Servicios.....	98
Figura 37: Diagrama de Clases Vehículos - Lógica de Negocio.....	99
Figura 38: Diagrama de Clases – Iteración 3, Cliente Móvil.....	100
Figura 39: Grafo Navegación – Iteración 3, Cliente Móvil.....	100
Figura 40: Resumen de progreso - Iteración 3.....	101
Figura 41: Burndown Chart – Tareas Iteración 3.....	102
Figura 42: Burndown Chart – Iteración 3.....	102
Figura 43: DA019 Generar Documento Contrato (H019).....	105
Figura 44: DA020 Generar Documento Expedientes (H020).....	105
Figura 45: DA021 Generar Documento Vehiculos (H021).....	106
Figura 46: Diagrama de Clases – Documentos: Capa de Servicio, Servidor.....	111
Figura 6: Diagrama de Clases - Módulo Documentos, Cliente Escritorio.....	113
Figura 47: Diagrama de Clases – Documentos: Lógica de Negocio, Servidor.....	113
Figura 48: Diagrama de Clases – Iteración 4, Cliente Móvil.....	115
Figura 49: Resumen de progreso - Iteración 4.....	116
Figura 50: Burndown Chart – <i>Iteración 4</i>	116

Figura 51: Burndown Chart – Iteración 4.....	116
Figura 52: DA032 Acceso Seguro a la Aplicación (H032).....	119
Figura 53: DA033 Listar Alumnos por Estado Expediente (H033).....	120
Figura 54: Diagrama de Secuencia - Funcionamiento básico JWT.....	122
Figura 55: Diagrama de Clases - Autenticación JWT, Servidor.....	124
Figura 56: Diagrama de Clases - Usuarios, Lógica de Negocio, Servidor.....	125
Figura 57: <i>Diagrama de Clases – Iteración 5, Cliente Móvil</i>	126
Figura 58: <i>Grafo de Navegación – Iteración 5, Cliente Móvil</i>	127
Figura 59: Resumen de progreso - Iteración 5.....	128
Figura 60: Burndown Chart – Tareas <i>Iteración 5</i>	129
Figura 61: Burndown Chart – Iteración 5.....	129
Figura 62: Resumen de progreso – Product backlog.....	133
Figura 63: Burndown Chart – Final de proyecto.....	134
Figura 64: <i>Diagrama de Clases – Iteración 1, Cliente Móvil</i>	137
Figura 65: Grafo Navegación – Iteración 1, Cliente Móvil.....	138
Figura 66: Cliente Móvil: Grafo Navegación 2º Iteración.....	139
Figura 67: Diagrama de Clases – Iteración 3, Cliente Móvil.....	140
Figura 68: Grafo Navegación - 3º Iteración, Cliente Móvil.....	141
Figura 69: Diagrama de Clases – Iteración 4, Cliente Móvil.....	142
Figura 70: <i>Diagrama de Clases – Iteración 5, Cliente Móvil</i>	143
Figura 71: <i>Grafo de Navegación – Iteración 5, Cliente Móvil</i>	144
Figura 72: Diagrama de Clases – Lógica de Neocio, Servidor.....	145
Figura 73: Diagrama de Clases – Capa de Servicios ,Servidor.....	146
Figura 74: Formularios Autoescuela – Contrato de enseñanza.....	147
Figura 75: Formulario Autoescuela – Expediente del alumno.....	148
Figura 76: Formulario Autoescuela – Contabilidad alumno.....	149
Figura 77: Formulario Autoescuela – Parte diario de clases.....	150
Figura 78: Documentos – Contrato de enseñanza.....	151
Figura 79: Documento – Ficha del alumno.....	152
Figura 80: Documento – Informe del vehículo.....	153
Figura 81: Barra de navegación.....	162
Figura 82: Barra de progreso.....	162
Figura 83: Vista <i>BuscarAlumno</i>	163
Figura 84: Vista <i>CrearAlumno</i>	164
Figura 85: Vista <i>CrearAlumno</i>	164
Figura 86: Vista <i>MostrarAlumno</i>	165
Figura 87: Vista <i>EditarAlumno</i>	166
Figura 88: Vista <i>EditarAlumno</i>	166
Figura 89: Vista <i>MostrarExpedientes</i>	167
Figura 90: Vista <i>MostrarExpedientes</i>	167
Figura 91: Vista <i>CrearExpediente</i>	168
Figura 92: Vista <i>CrearExpediente</i>	168
Figura 93: Vista <i>EditarExpediente</i>	169
Figura 94: Vista <i>MostrarExámenes</i>	170
Figura 95: Vista <i>MostrarExámenes</i>	170
Figura 96: Vista <i>CrearExamen</i>	171
Figura 97: Vista <i>MostrarVehiculos</i>	172
Figura 98: Vista <i>CrearVehículo</i>	173
Figura 99: Vista <i>CrearVehículo</i>	173
Figura 100: Vista <i>EditarVehículo</i>	174
Figura 101: Vista <i>MostrarMantenimientos</i>	175
Figura 102: Vista <i>CrearMantenimiento</i>	176
Figura 103: <i>Error de elemento no encontrado</i>	177
Figura 104: Error de conexión.....	177

Figura 105: Error Interno del Servidor.....178

Figura 106: Error Desconocido.....178

Índice de tablas

Tabla 1: Project Overview Statement.....	22
Tabla 2: Identificación y clasificación de riesgos.....	24
Tabla 3: Respuestas a los diferentes riesgos.....	24
Tabla 4: Cuadrantes PMLC.....	26
Tabla 5: Product Backlog - Versión Inicial.....	35
Tabla 6: Personal.....	41
Tabla 7: Coste de personal.....	41
Tabla 8: Coste del hardware.....	42
Tabla 9: Coste del Software.....	42
Tabla 10: Costes indirectos.....	43
Tabla 11: Presupuesto final.....	44
Tabla 12: H001 Añadir Alumno.....	53
Tabla 13: H002 Modificar Alumno.....	53
Tabla 14: H003 Borrar Alumno.....	54
Tabla 15: H004 Búsqueda por DNI.....	54
Tabla 16: Backlog de iteración – <i>Iteración 1</i>	58
Tabla 17: Especificación API REST - Alumnado.....	61
Tabla 18: H005 Añadir Expediente.....	70
Tabla 19: H006 Borrar Expediente.....	70
Tabla 20: H007 Modificar Expediente.....	71
Tabla 21: H008 Añadir examen.....	71
Tabla 22: H009 Borrar examen.....	72
Tabla 23: H010 Listar Expedientes.....	72
Tabla 24: H011 Listar Exámenes.....	73
Tabla 25: Backlog de Iteración – <i>Iteración 2</i>	77
Tabla 26: H012 Añadir Vehículo.....	84
Tabla 27: H013 Modificar Vehículo.....	84
Tabla 28: H014 Borrar Vehículo.....	85
Tabla 29: H015 Añadir Operación de mantenimiento.....	85
Tabla 30: H016 Borrar Operaciones de mantenimiento.....	85
Tabla 31: H017 Listar Vehículos.....	86
Tabla 32: H018 Listar Operaciones de mantenimiento.....	86
Tabla 33: <i>Backlog de Iteración – Iteración 3</i>	91
Tabla 34: Especificación API REST - Vehículos.....	95
Tabla 35: H019 Generar Documento Contrato.....	103
Tabla 36: H020 Generar Documento Expediente.....	103
Tabla 37: H021 Generar Documento Vehículo.....	104
Tabla 38: <i>Backlog de Iteración - Iteración 4</i>	106
Tabla 39: Especificación API REST - Documentos.....	110
Tabla 40: H032 Acceso Seguro a la Aplicación.....	118
Tabla 41: H033 Listar Alumnos por Estado Expediente.....	118
Tabla 42: Backlog de Iteración – <i>Iteración 5</i>	120
Tabla 43: JSON Web Token - Ejemplo.....	123

1 ESPECIFICACIÓN DEL PROYECTO

1.1 Introducción

Tras cuatro años de formación en un grado orientado hacia la Ingeniería Informática, se considera al alumno cualificado para ejercer este trabajo profesionalmente. Este Trabajo de Fin de Grado plantea la aplicación del conocimiento aprendido desarrollando un proyecto cercano a la realidad, con un cliente real y casos de uso cercanos a la realidad. Casos de uso que se esperan tras los estudios si se pretende desarrollar un sistema software.

En este Trabajo de Fin de Grado se pretenden reflejar habilidades adquiridas en varios aspectos de la Ingeniería Informática, no solo meramente técnicos, como puede ser el uso de varias tecnologías o el diseño del sistema software, sino también sociales, como la obtención de requisitos en un entorno real, la comunicación con el cliente o la gestión, el trabajo y planificación de recursos en un entorno colaborativo. En definitiva, demostrar el nivel de adquisición de las diferentes competencias adquiridas a lo largo del grado y durante la experiencia universitaria en general.

Propusimos este proyecto porque creemos que la complejidad a la que hace frente la ingeniería del software no reside únicamente en la resolución de problemas técnicos. También reside en el ámbito social de un proyecto, pues en la práctica, la mayoría de proyectos necesitan de una colaboración estrecha y organizada entre personas para alcanzar un resultado de calidad. Asimismo, consideramos que la mejor forma de afianzar los conocimientos adquiridos durante el curso e iniciar nuestra vida en el mundo laboral, no es sino el poner estos conocimientos a prueba en un proyecto tan realista como el que hemos propuesto.

Antes de realizar la propuesta teníamos una idea clara en mente, el desarrollo en equipo de una aplicación en un entorno real. Para esto recorrimos una buena parte de la ciudad de Jaén en busca de autoescuelas que estuviesen dispuestas a colaborar con nosotros. En gran parte de los casos, dichas autoescuelas o bien ya disponían de un software (aunque algo desactualizado), o el responsable no estaba presente. Cuando nos reunimos con la autoescuela *Zona L* y nos dieron una

respuesta afirmativa estábamos dispuestos a realizar una aplicación de escritorio capaz de hacer frente a las alternativas del mercado. Es decir, crear un producto competitivo a pesar de ser solo dos desarrolladores. Sin embargo, nos encontramos con varios obstáculos en el camino. A la hora de presentar la propuesta, nos encontramos que era un requisito diferenciar nuestras responsabilidades dentro del Trabajo de Fin de Grado. A raíz de esto, nuestra idea inicial de realizar un solo cliente se vio descartada por la necesidad de añadir una nueva aplicación para dispositivos Android. Esta idea no nos agradaba puesto que no solo no existía una necesidad real por parte de la autoescuela *Zona L* de una aplicación móvil, sino que además de imposibilitar nuestra cooperación en un mismo cliente, ahora tendríamos que desarrollar dos clientes por separado cuyas tecnologías desconocíamos por completo. Dividiendo nuestros esfuerzos a la hora de desarrollar características competitivas para el cliente.

A pesar de eso, la parte del servidor sí la podíamos hacer en equipo y estábamos expectantes ante el reto que realizar dos clientes nativos. Una vez se aprobó la propuesta nos pusimos manos a la obra.

Este Trabajo de Fin de Grado forma parte de un proyecto en equipo junto con el Trabajo de Fin de Grado *Análisis, diseño e implementación de una aplicación web de gestión de autoescuelas: backend+ frontend ordenador* del alumno Javier Martínez Cavanillas. El análisis, diseño e implementación del servidor así como la gestión del proyecto serán comunes a los dos trabajos, mientras que el análisis, diseño e implementación del cliente Android pertenece de forma exclusiva a este Trabajo de Fin de Grado.

1.2 Objeto del proyecto

El objetivo principal del proyecto consiste en implementar una aplicación que permita gestionar una autoescuela en un entorno real, en base a los requisitos especificados por la misma. Para ello, contamos con la estrecha colaboración de la *Autoescuela Zona L* de Jaén, que se compromete a actuar como cliente durante el transcurso del trabajo de fin de grado, ayudándonos durante el proceso y guiándonos hacia la consecución de una aplicación de cierta calidad en el tiempo disponible para el trabajo.

En líneas generales, la aplicación permitirá gestionar los diferentes alumnos registrados en la autoescuela, el trascurso general de su educación en la autoescuela, los vehículos registrados dentro de la autoescuela y los profesores que imparten las clases en esta. No obstante, los requisitos y su evolución serán definidos más adelante en este documento.

1.3 Antecedentes y estado del arte

En el mercado hay gran variedad de aplicaciones destinadas a la gestión y mantenimiento de las operaciones realizadas en las autoescuelas, las cuales nos podrían servir de inspiración para el desarrollo de nuestra aplicación.

Sin embargo, gran cantidad de ellas están destinadas a la realización de exámenes tipo test para la evaluación teórica de los diferentes permisos de conducción, y aquellas aplicaciones que no lo son, suelen ser de pago o con versiones de prueba limitadas. A continuación, haremos un pequeño análisis de las características que poseen algunas de las aplicaciones que a nuestro parecer, son las más punteras y completas.

1.3.1 WinAutoGest

WinAutoGest es una aplicación para autoescuelas que cubre una amplia variedad de funciones. Entre ellas: la gestión de alumnos, los expedientes de los alumnos, ciertos impresos de la DGT, permisos de usuarios, contabilidad y una agenda para los profesores. La aplicación excede el alcance dispuesto para nuestro proyecto pero podemos aprender de sus funcionalidades. Además, estas herramientas se ofrecen en pequeños paquetes que se ofertan mediante mensualidades de 15, 20 y 25 euros.



Figura 1: Logo WinAutoGest

La aplicación principal funciona mediante un servicio web y una aplicación móvil complementaria. Nuestra aplicación posee características similares. Sin embargo, nuestra aplicación expone una interfaz REST que es utilizada por varios clientes ejecutados de forma nativa en Windows y Android. Al exponer una interfaz REST, las funcionalidades del servidor podrán ser utilizadas posteriormente por otro tipo de clientes o incluso otros servicios.

1.3.2 Control L

Al igual que la anterior, se vende en diferentes paquetes pero con funciones incrementales. La versión más barata se centra en el desarrollo de clases prácticas, mientras que conforme vamos subiendo en la categoría de la aplicación (versiones Advance y Premium), adquirimos los mecanismos más esenciales: Gestión de alumnos, documentación de DGT y demás utilidades que son de uso más administrativo. Fomentando así la compra de los planes más caros.

Esta aplicación es más restrictiva que la anterior, pero incluye aproximadamente las mismas utilidades. Y de la misma forma, gran parte de ellas requieren del pago de las suscripciones más caras, siendo la suscripción más alta la única en la que se ofrece un producto más personalizado a la autoescuela añadiendo logos y suministrando un servidor exclusivo.



Figura 2: Logo Control L

1.3.3 Conclusión

Es una práctica habitual entre las aplicaciones disponibles para gestión de autoescuelas la gestión de los alumnos y los expedientes de estos, así como en menor medida, la de los impresos relacionados al proceso educativo. Más allá de estas características esenciales, disponemos de más utilidades menos necesarias y

que suelen formar el grueso de los paquetes iniciales, incentivando la adquisición de suscripciones más caras.

En nuestro caso, nos centraremos en las necesidades más destacadas para la empresa contratante *Zona L*. Que coinciden con aquellas que se ofrecen en los paquetes más caros. Es decir, la gestión de los alumnos y sus expedientes.

1.4 Descripción de la situación actual

En nuestro caso específico, nuestra autoescuela cuenta con un equipo informático básico para permitir a sus alumnos practicar los tests del examen teórico a través de una plataforma informática, y realizar tareas sencillas de ofimática para la redacción de documentos.

1.4.1 Descripción del entorno actual

Actualmente la información relacionada con la autoescuela se gestiona de manera tradicional mediante impresos en papel, con la ayuda de documentos y hojas de cálculo en formato electrónico. La autoescuela dispone de dos ordenadores para administración y varios terminales para el uso de los alumnos.

La principal tarea a informatizar es la gestión de los alumnos de la autoescuela. Actualmente, cada alumno dispone de una ficha de usuario con la información personal básica y los pagos realizados.

Asociada a esta ficha se incluyen documentos como el contrato, la ficha de aceptación en la DGT y demás formularios asociados al alumno. La documentación con Tráfico depende de la DGT; actualmente se realiza mediante impresos, aunque próximamente se pretende implantar la presentación de dichos documentos de manera telemática. Cada ficha está organizada en distintos estantes según el estado de la formación del alumno (*estudiando para el examen teórico, estudiando para el examen práctico, apto o deshabilitado*) y el tipo de permiso contratado (AM, A1, A2, A, B, B96, B+E, C1, C1+E, C, D1, D1+E, D).

Los tests de aprendizaje para el examen teórico se realizan ya de manera informatizada mediante una plataforma existente. Por lo que por ahora no se contempla integrarlos en la aplicación.

1.4.2 Resumen de las principales deficiencias encontradas

La principal deficiencia en el sistema de la autoescuela *Zona L* es que al realizar la mayoría del proceso de forma escrita y en cierta medida desordenada. El recuperar información necesaria en una etapa posterior del aprendizaje (por ejemplo: datos necesarios para presentarse a algún examen) supone un gran esfuerzo y es frecuente que se traspapelen hojas, complicando más aún el proceso. Por ello se plantea la necesidad de informatizar todo el recorrido de los alumnos en la autoescuela así como los métodos de trabajo de la institución.

Otro aspecto importante, es que al realizar sus operaciones tanto de forma escrita como en formato digital, se carece de un sistema unificado y en ocasiones se ha de repetir trabajo pasando información de un formato a otro. Esta situación no podemos solventarla del todo debido a la forma de operar de la empresa, sin embargo nuestro proyecto ayudará a paliar las deficiencias encontradas y facilitar en buena medida el trabajo diario de los operadores.

1.5 Requisitos iniciales

Se pretende informatizar los datos referentes al alumno incluidos en la ficha, así como el historial de pagos realizados, permitiendo la posibilidad de imprimir las fichas de forma automática y localizarlas mediante el número de expediente, DNI o nombre del alumno. Se podrán listar los alumnos según el tipo de permiso que estén cursando, el año de matriculación y la fecha de superación del examen. Además, sería recomendable organizar los alumnos según su estado de formación actual (*teoría, practicas, superado*) e incorporar un sistema para marcar los alumnos que han abandonado las clases.

Los profesores podrán acceder y cumplimentar su parte diario de clase. En este documento se apuntan las clases prácticas que han dado y el número de horas totales de dicho profesor en ese día. Se hace referencia también al tipo de permiso de la clase y al vehículo utilizado.

En la parte de contabilidad se quieren almacenar los gastos e imprimir una hoja de gastos diarios. También sería interesante almacenar un registro de los pagos realizados con tarjeta.

La característica principal que debe tener la aplicación de cara al usuario es la sencillez. Se prefiere que disponga de pocas funciones que puedan realizarse de

forma simple e intuitiva, en contraposición con las aplicaciones actualmente en el mercado.

1.6 Alcance

Para establecer el alcance se ha elaborado un documento de alcance, el *Project Overview Statement* (Tabla 1). En este documento se establece la problemática, se define brevemente la oportunidad de negocio y los objetivos principales del proyecto. También quedan definidos los criterios de satisfacción que se emplearán para evaluar el estado final del proyecto y las principales hipótesis y restricciones que pensamos encontrarnos durante el desarrollo.

Tabla 1: *Project Overview Statement*

PROJECT OVERVIEW STATEMENT	Nombre de Proyecto Autoescuela TFG 1819	Coordinador de Proyecto Javier Martínez Cavanillas
Problema / Oportunidad El sistema de gestión de la autoescuela Zona L de Jaén se realiza de forma manual manejando documentos en papel. Se desea modernizar el sistema de forma que las operaciones realizadas se ejecute de una forma más rápida.		
Finalidad Diseñar e implementar un sistema informático que permita la gestión y desarrollo de las diferentes actividades llevadas a cabo en la autoescuela		
Objetivos 1. El Usuario podrá gestionar los alumnos dados de alta en la autoescuela 2. El Usuario dispondrá de un sistema para gestionar los vehículos de la autoescuela 3. El Usuario podrá gestionar el profesorado de la autoescuela 4. El Profesorado podrá dar parte de sus clases en el sistema 5. El Usuario podrá llevar la contabilidad general de la empresa en el sistema 6. El Usuario podrá rellenar e imprimir de manera automática los documentos actualmente asociados a la empresa		
Criterios de aceptación 1. La interfaz de usuario debe ser sencilla e intuitiva con una baja curva de aprendizaje 2. Se debe obtener un rendimiento superior al sistema tradicional 3. Debe ser fácilmente accesible para cualquier miembro autorizado de la autoescuela		
Hipótesis / Riesgos / Obstáculos 1. La estimación en coste y tiempo puede resultar incorrecta 2. Baja temporal de un miembro del equipo 3. Problema de integración entre las tecnologías del cliente y el servidor 4. Se pueden producir muchos cambios en el ámbito actual de aplicación		

5. Pérdida de información por fallo en el hardware 6. Dificultad para adaptarse a las tecnologías utilizadas			
Redactado por Javier Martínez Cavanillas	Fecha 19/5/2019	Aprobado por Javier Martínez Cavanillas	Fecha 19/5/2019

1.6.1 Hipótesis y Restricciones

El TFG se define como una asignatura de 12 créditos, lo que supone que la duración total del proyecto será de 300 horas, incluyendo todas las etapas del ciclo de vida, con la excepción del mantenimiento. Por consiguiente, la principal restricción aplicable es la limitación de la duración del trabajo.

El horario de trabajo y su disposición en el calendario deberá ser flexible en la medida de lo posible, debido a que el equipo de trabajo debe conciliar sus tareas en el Trabajo de Fin de Grado con el resto de asignaturas cursadas por los alumnos durante este año.

1.7 Análisis de riesgos

Tras la identificación de los principales riesgos para el proyecto, reflejados en el *Project Overview Statement*, se han priorizado según la guía ofrecida por el PMBOK (*Project Management Base Of Knowledge*), clasificándolos según la marca obtenida de la tabla que aparece en la Figura 3.

Probability and Impact Matrix										
Probability	Threats					Opportunities				
0.90	0.05	0.09	0.18	0.36	0.72	0.72	0.36	0.18	0.09	0.05
0.70	0.04	0.07	0.14	0.28	0.56	0.56	0.28	0.14	0.07	0.04
0.50	0.03	0.05	0.10	0.20	0.40	0.40	0.20	0.10	0.05	0.03
0.30	0.02	0.03	0.06	0.12	0.24	0.24	0.12	0.06	0.03	0.02
0.10	0.01	0.01	0.02	0.04	0.08	0.08	0.04	0.02	0.01	0.01
	0.05	0.10	0.20	0.40	0.80	0.80	0.40	0.20	0.10	0.05

Impact (ratio scale) on an objective (e.g., cost, time, scope or quality)

Each risk is rated on its probability of occurring and impact on an objective if it does occur. The organization's thresholds for low, moderate or high risks are shown in the matrix and determine whether the risk is scored as high, moderate or low for that objective.

Figura 3: Matriz de Probabilidad e Impacto (PMBOK)

Tras la clasificación de los riesgos, obtenemos la tabla Tabla 2:

Tabla 2: Identificación y clasificación de riesgos

Identificación de riesgos		Clasificación Cualitativa			
Riesgo	Categoría	Probabilidad	Impacto	Marca	Prioridad
Se pueden producir cambios frecuentes en el ámbito actual de la aplicación	Riesgo de requerimientos	0,9	0,2	0,18	1
Baja temporal de un miembro del equipo	Riesgo de personal	0,3	0,4	0,12	2
La estimación en coste y tiempo puede resultar incorrecta	Riesgo de estimación	0,7	0,2	0,14	3
Dificultad para adaptarse a las tecnologías utilizadas	Riesgo de personal	0,3	0,4	0,12	4
Posible pérdida de información por fallo en el hardware	Riesgo de herramientas	0,05	0,8	0,04	5
Problema de integración entre las tecnologías del cliente y el servidor	Riesgo de tecnologías	0,1	0,2	0,02	6

Según su prioridad, asignamos una respuesta al riesgo, identificamos un disparador que pueda indicarnos que el riesgo se está dando o que es probable que lo suframos en un futuro cercano, y asignamos una persona encargada de controlar cada riesgo durante el transcurso del proyecto. En la Tabla 3 se recogen los encargados de supervisar los diferentes riesgos y las respuestas planeadas.

Tabla 3: Respuestas a los diferentes riesgos

Respuestas		
Respuesta	Disparador	Responsable
Utilizar un modelo ágil para gestionar mejor los cambios en el ámbito, manteniendo una comunicación fluida con el cliente	Se añaden >7 historias de usuario extra al ámbito del proyecto	Alberto Carrillo Ortega
Utilizar la reserva del presupuesto destinada a riesgos para contratar un trabajador a tiempo parcial o delegar parte del trabajo	El trabajador solicita una baja por enfermedad o disminuye su ritmo de trabajo en más de un 40% debido a su enfermedad.	Javier Martínez Cavanillas

a terceros.		
Dejar un margen hasta la entrega del proyecto de un par de semanas en caso de tener que utilizar un iteración extra	Se produce un retraso en de más del 20% tras 2 iteración	Javier Martínez Cavanillas
Procurar escoger herramientas familiares a los desarrolladores	Se produce un retraso en de más del 20% tras 2 iteraciones y más de 5 informes de problemática relativos al uso de una herramienta	Javier Martínez Cavanillas
Copias de seguridad e información en la nube	-	Alberto Carrillo Ortega
Dedicar recursos extra a la documentación e implementación de adaptadores. En el peor caso, cambiar de tecnología	Se produce un retraso las tareas de integración entre aplicaciones durante 2 iteraciones consecutivas	Alberto Carrillo Ortega

1.8 Ciclo de vida del Proyecto y Organización

Antes de empezar con el proyecto, es conveniente establecer una estructura de organización del equipo y un método concreto de planificación durante el desarrollo. Por ello, analizamos las características del proyecto para escoger el flujo de trabajo que mejor se adapte a las necesidades del mismo y establecemos una serie de normas que estructuren el trabajo realizado.

1.8.1 Elección del ciclo de vida del proyecto

En la entrevista inicial con el cliente se obtuvo una visión general de los requisitos que formarán el ámbito del proyecto. Sin embargo, el cliente no sabe expresar con precisión las características del producto deseado, un ejemplo de esto es la demanda desde la primera entrevista de un prototipo o programa con funcionalidad básica para ver la estructura del software y proponer cambios, mejoras y nuevos requisitos.

A la hora de escoger un Modelo de ciclo de vida del proyecto o PMLC (*Project Management Life Cycle*), estos se pueden clasificar en cuatro cuadrantes (Tabla 4).

Tabla 4: Cuadrantes PMLC

		Solución	
		Definida	Sin definir
Objetivo	Sin definir	Modelo Emerxe (Proyectos de I+D)	Gestión de proyectos extrema
	Definido	Gestión de proyectos tradicional	Gestión de proyectos ágil

En nuestro proyecto, el objetivo final se encuentra definido. Sin embargo, hay muchas características de la solución final que no están definidas y no pueden serlo hasta que el desarrollo del proyecto arroje luz sobre las necesidades reales del usuario al utilizar el producto. Por lo tanto, es conveniente centrarse en una gestión de proyectos ágil. Una gestión de proyectos ágil se caracteriza por utilizar una especificación del proyecto menos formal al principio del mismo, que se va refinando mediante iteraciones en las que se hace entrega de parte del producto ya desarrollado al usuario, para obtener un feedback que permita ir completando la definición de la solución paulatinamente.

A la hora de escoger un ciclo de vida ágil, hay que tener en cuenta una serie de puntos clave que deben guiar el desarrollo del proyecto:

1. El objetivo global del proyecto sigue siendo el mismo especificado en la negociación inicial del ámbito y recogido en el *Project Overview Statement*. Aunque se contemplen cambios en los requisitos, estos deben ser coherentes con el ámbito y plazos establecidos.
2. La colaboración activa del cliente cobra un papel importante dentro del proyecto. Su implicación en el proyecto y la experiencia del usuario en su campo de trabajo supondrán la base para conseguir un producto de utilidad real para el cliente.
3. La negociación pasa a aplicarse ahora durante todo el desarrollo del proyecto, añadiendo complejidad adicional a la hora de reorganizar los recursos disponibles e introduciendo dinamismo a la hora de gestionar los requisitos, por lo que algunos serán descartados y otros nuevos añadidos.

1.8.2 Metodología de desarrollo

Debido a las conclusiones obtenidas y explicadas en el punto anterior, se ha decidido adoptar un ciclo de vida ágil, ya que los cambios en el ámbito y la necesidad de prototipado para guiar al cliente sobre las características de la aplicación resultan necesarias.

Para ello, se optará por un modelo de trabajo ágil inspirado en *Scrum* y *Kanban*, ajustado a las características especiales del equipo. (Bajo número de integrantes). Al estar conformado el equipo por solo 2 integrantes se asigna la responsabilidad de *Scrum Master* al Coordinador de Proyecto, mientras la responsabilidad de *Product Owner* se asignará al miembro restante. Ambos pertenecerán al equipo de desarrollo. Se dará una gran importancia a la participación en el proyecto del cliente.

El *Scrum Master* se encargará de supervisar el seguimiento de la planificación del proyecto, así como de reorganizar los recursos disponibles en cada momento.

El *Product Owner* será responsable de gestionar los distintos cambios en el ámbito, así como de asegurarse de que el cliente se encuentra satisfecho con el desarrollo del proyecto y las características del producto.

Las tareas de desarrollo del equipo en cada iteración serán estimadas en la planificación de la iteración a principio de las mismas y serán gestionadas por el equipo de desarrollo en conjunto con ayuda de un tablero *Kanban* virtual alojado en la plataforma *GitLab*.

Responsabilidades:

- *Scrum Master* / Coordinador de proyecto : Javier Martínez Cavanillas
- *Product Owner* : Alberto Carrillo Ortega
- Equipo de Desarrollo:
 - Alberto Carrillo Ortega
 - Javier Martínez Cavanillas

1.8.3 Reglas de Operativa del equipo

1.8.3.1 Resolución de problemas ligados al desarrollo del proyecto

El proceso de detección de problemas se realizará mediante las reuniones diarias de estado del equipo. Tras la detección de un problema se tomarán medidas para corregirlo, si el problema persiste o no se consigue solucionar de manera sencilla el coordinador de proyecto convocará una reunión de resolución de problemas.

1.8.3.2 Toma de decisiones

El coordinador de proyecto convocará una reunión de toma de decisiones para resolver un problema relacionado sobre la elección de una nueva tecnología o un desacuerdo a la hora de diseñar el sistema. Se discutirán las posibles soluciones durante el desarrollo de la misma, si no se alcanza un acuerdo tras finalizar esta se convocará una segunda reunión para proseguir el debate. En caso de que el acuerdo no llegue a consolidarse, la decisión corresponderá al coordinador de proyecto.

1.8.3.3 Resolución de conflictos internos del equipo

En caso de que se produzca algún conflicto entre los miembros del equipo, se recurrirá los tutores del Trabajo de Fin de Grado para actuar de moderadores durante la resolución del mismo.

1.8.3.4 Reuniones

Se realizaran los siguientes tipos de reuniones:

1. Reuniones Diarias: Reuniones breves de 10 – 15 minutos. En ellas se debatirá el estado actual de las tareas en proceso y se expondrán los posibles problemas que se presenten durante el desarrollo de las mismas, con el fin de detectar y corregir los posibles desvíos del proyecto.

Periodicidad: Una vez al día.

2. Reuniones de Estado Semanales: Reuniones de 20 minutos todas las semanas, preferiblemente los viernes. En dichas reuniones se resumirá el progreso realizado durante la semana y se revisará la planificación del proyecto.

Periodicidad: Una vez por semana.

3. Reuniones de Final de Iteración: Reuniones con el cliente para presentar los entregables desarrollados durante la iteración. Se realizarán pruebas de satisfacción del usuario, revisiones en el ámbito del proyecto y recepción de feedback.

Periodicidad: Tras finalizar cada iteración .

4. Reuniones de Decisión: Reuniones de 30 – 45 minutos usualmente. En ellas se discutirá entre los miembros del equipo la resolución de problemáticas asociadas a la elección entre varias alternativas.

Periodicidad: Cada vez que se presente la necesidad de tomar una decisión importante para el desarrollo del proyecto. El Coordinador de proyecto será el responsable de convocarlas.

5. Reuniones de Resolución de Problemas: Reuniones en las que se discutirá el plan a seguir para la resolución de obstáculos significativos para el desarrollo del proyecto. Su duración dependerá de la magnitud del problema a tratar.

Periodicidad: Cada vez que se presente un obstáculo significativo para el desarrollo del proyecto. El Coordinador de proyecto será el responsable de convocarlas.

6. Reuniones de Modificación del Ámbito: Reuniones de 20 – 30 minutos en las que se contemplará la inclusión de las propuestas del cliente en el ámbito del proyecto.

Periodicidad: Las reuniones de modificación del ámbito se celebrarán a consecuencia de una petición del cliente fuera de las reuniones de final de Iteración. La potestad de convocar estas reuniones recaerá en el *Product Owner*.

1.9 Material y métodos

En esta sección se describen los recursos materiales y herramientas software previstas a utilizar durante el transcurso del proyecto, proporcionando una breve descripción de ellas y señalando las diversas ventajas que nos han llevado a utilizarlas frente a las principales alternativas existentes en el mercado.

1.9.1 Recursos Materiales

Los recursos materiales son pocos; el equipo de desarrollo cuenta con dos equipos para el desarrollo del proyecto y un móvil para realizar las pruebas:

- Portátil MSI Apache Pro GE62 – i7 6700HQ, 16GB de memoria RAM, gráfica GTX 970M, 256 GB de almacenamiento SSD, 1TB de almacenamiento HDD.
- Portátil Asus Rog Strix Scar Edition GL703GM-E5016 – i7 8750H, 16 GB de memoria RAM, gráfica GTX 1060, 256 GB de almacenamiento SSD, 1TB de almacenamiento HDD.
- Teléfono Móvil Bq Aquaris X, Qualcomm Snapdragon 626, 3GB de memoria RAM, 32 GB de almacenamiento interno.

1.9.2 Herramientas Software

1.9.2.1 Apache NetBeans

Apache NetBeans 10 es un entorno de desarrollo integrado libre y gratuito, pensado principalmente para trabajar con Java. Apache NetBeans brinda diferentes herramientas a los desarrolladores como la gestión de proyectos, gestión de dependencias vía Maven, integración con sistemas de control de versiones, autocompletado y navegación por el código, herramientas de refactorización o soporte para tests unitarios.

Se ha escogido Apache NetBeans debido a la experiencia previa del equipo de desarrollo trabajando con este, la rápida puesta en marcha del proyecto y la integración y soporte que brinda para Spring Framework y JavaFX. Tanto el servidor como el cliente para sobremesa serán desarrollados en este entorno.

1.9.2.2 Visual Paradigm

Visual Paradigm es una herramienta CASE que soporta varios tipos de diagramas UML. Es una de las herramientas para modelado UML más completas del mercado, ofreciendo varias versiones (la mayoría de pago), que ofrecen diferentes características.

Utilizaremos Visual Paradigm en su versión Standard Edition, proporcionada por la Universidad de Jaén, a la hora de modelar los diferentes componentes del

sistema durante las fases de análisis y diseño, así como para elaborar los distintos diagramas UML presentados en este documento.

1.9.2.3 *Android Studio*

Android Studio es el entorno de desarrollo integrado oficial para la plataforma Android. Entre sus características podemos destacar la gestión de proyectos, control de versiones, autocompletado y navegación por el código, consola de desarrollador, herramientas de refactorización, plantillas para crear diseños comunes en aplicaciones Android o un editor de diseño de interfaz integrado.

Se usará Android Studio en el desarrollo del cliente enfocado a dispositivos móviles.

1.9.2.4 *LibreOffice*

LibreOffice es un paquete de software de oficina libre y de código abierto desarrollado por *The Document Foundation*. Ofrece un procesador de texto, editor de hojas de cálculo, editor de presentaciones, gestor de bases de datos, etc.

Se utilizará LibreOffice para elaborar la documentación y tablas de datos asociadas al proyecto.

1.9.2.5 *Git y GitLab*

Git es un sistema de control de versiones distribuido libre y de código abierto de alto rendimiento. Usaremos este sistema en conjunto con la plataforma GitLab para mantener los cambios y versiones del proyecto tanto localmente en nuestros equipos como en la nube de manera sencilla.

El usar estas plataformas nos otorga las siguientes ventajas a la hora de gestionar y desarrollar el proyecto:

- Control de versiones y registro de cambios realizado por cada usuario.
- Capacidad de almacenar y mantener diferentes entregas mediante etiquetas.
- Desarrollo colaborativo mediante la fusión de repositorios y repositorios sincronizados en la nube.
- Gestión centralizada de actividades mediante tableros *Kanban* en GitLab.

1.9.3 Tecnologías Utilizadas

1.9.3.1 OpenJDK

OpenJDK 12 es la implementación de la Máquina Virtual Java (JVM) que usaremos para el desarrollo del sistema. En contraposición con la implementación ofrecida por Oracle, esta es gratuita y libre, y es más accesible a la hora de instalarse en otros sistemas operativos que no sean de pago o no soporten el software privativo.

Trabajaremos con esta especificación de Java tanto en el servidor como en el cliente para ordenadores portátiles y de escritorio.

1.9.3.2 Spring Framework

A la hora de desarrollar el sistema del servidor, utilizaremos el framework Spring con Spring Boot para desarrollar de forma rápida y eficiente múltiples tareas de gestión que suponen una gran complejidad a la hora de la implementarlas desde cero. Algunas las funcionalidades que nos proporciona Spring son:

- Conexión e implementación REST mediante sistema nativo o conforme a la especificación estándar JAX RS, utilizando Apache Tomcat como contenedor de aplicaciones en el servidor.
- Contenedor de inversión de control (IoC) e inyección de dependencias.
- Soporte para programación orientada en aspectos (AOP).
- Configuración automática para el mapeado ORM mediante Hibernate utilizando la especificación JPA 2.0.
- Soporte para múltiples sistemas de caché.
- Spring Security, un paquete de seguridad avanzado.

1.9.3.3 JavaFX

A la hora de desarrollar la Interfaz Gráfica del cliente de escritorio se ha optado por utilizar OpenJFX 12, la versión de JavaFX para OpenJDK 12 como principal tecnología. Se ha elegido esta tecnología frente a alternativas como AWT o Java Swing debido a que JavaFX dispone de distintas ventajas frente a sus competidores:

- Capacidad de personalizar el aspecto gráfico de la aplicación de forma sencilla mediante hojas de estilo en cascada (CSS). Factor muy importante a la hora de proporcionar a la aplicación un color y aspecto en consonancia con la marca de la empresa y los gustos expresados por el cliente.
- Soporte para contenidos multimedia.
- API e interfaz de comunicación más coherente que Java Swing.
- Soporte para animaciones y efectos gráficos.

1.9.3.4 Retrofit y OkHttp

Para conectar con la interfaz REST expuesta por el servidor, utilizaremos un cliente HTTP para Java de código abierto conocido como OkHttp. De los mismos desarrolladores de OkHttp, utilizaremos la biblioteca Retrofit, que trabaja en consonancia con OkHttp para consumir de forma sencilla servicios REST. Las principales ventajas que Retrofit y OkHttp nos proporcionan son:

- Una manera sencilla y estructurada de consumir nuestra API REST.
- Cacheado de llamadas HTTP automático.
- Funciona tanto en aplicaciones de escritorio como en sistemas Android.

1.9.3.5 PostgreSQL

Como sistema de gestión de bases de datos utilizaremos PostgreSQL, un SGBD relacional orientado a objetos y de código abierto con un amplio recorrido. Las ventajas que nos hacen escoger este sistema frente a otros como OracleDB, MySQL o MongoDB son las siguientes:

- Modelo de base de datos relacional que asegura la integridad referencial.
- Asegura la integridad transaccional.
- Soporte para procedimientos y disparadores.
- Dispone de una herramienta gráfica de gestión y monitorización de Bases de datos, pgAdmin, que nos permite visualizar en todo momento el esquema generado, los datos introducidos, crear copias de seguridad, ver el historial de transacciones, etc. de forma rápida y sencilla.

- Dispone de un buen soporte para trabajar con Spring e Hibernate, proporcionando una fácil configuración y optimizaciones específicas.
- El equipo dispone de experiencia previa en el uso de PostgreSQL y sus herramientas asociadas.

1.9.4 Métodos

Como principal guía durante el desarrollo el proyecto seguirá los principios indicados en la metodología *Scrum* recogido en el *Manifiesto Ágil* y *The Scrum Guide* y las recomendaciones y directrices recogidas en el PMBOK.

1.9.4.1 El Manifiesto Ágil

Establecida la naturaleza y ciclo de vida del proyecto, se decide organizar el desarrollo del proyecto de acuerdo a los principios ágiles y la metodología *Scrum*, los principios del desarrollo ágil recogidos en el *Manifiesto Ágil* son los siguientes.

1. Se valoran más las personas e interacción que los procesos y herramientas.
2. Se valora más el software que funcione que la documentación exhaustiva.
3. Se valora más la colaboración con el cliente que la negociación de contrato.
4. Se valora más la respuesta a cambios que el seguimiento de un plan.

A la hora de establecer el método de trabajo, nos inspiraremos en la guía *Scrum* recogida en *The Scrum Guide*, disponible en la pagina *Scrum.org*. Si bien se establece *The Scrum Guide* como guía y recurso de consulta, el proyecto no tiene por qué seguir las recomendaciones establecidas en su totalidad.

1.9.4.2 PMBOK

En complemento con la metodología *Scrum*, a la hora de gestionar los recursos relativos al proyecto, realizar el análisis de riesgos, la planificación de sprints y el control de calidad, se tendrán en cuenta las pautas recomendadas en el *Project Management Base of Knowledge* (PMBOK), un conjunto de buenas prácticas para la gestión de proyectos reconocido por el *Project Management Institute* (PMI). Si bien se establece el PMBOK como guía y recurso de consulta, el proyecto no tiene por qué seguir las recomendaciones establecidas en su totalidad.

1.10 Estimación de tamaño y esfuerzo

Debido a la naturaleza del proyecto, el equipo debe estar preparado para cambios en el ámbito del mismo. No obstante, es necesario fijar un plazo para el desarrollo del proyecto, ya que de otro modo, nuevas características e ideas se añadirían constantemente a la lista de tareas, y se perdería la visión planteada inicialmente. Además, se hace necesario saber una aproximación del coste total del proyecto, para poder ofrecer al cliente un presupuesto competitivo y ajustado en la medida de lo posible a la realidad del proceso de desarrollo posterior.

Para obtener esta información, el primer paso será definir un conjunto de historias de usuario iniciales, ayudándonos del contenido definido en el apartado 1.5 *Requisitos iniciales*, asignar una prioridad a las mismas y estimar el esfuerzo necesario para completarlas. La tabla 5 muestra el *Product Backlog*, el conjunto de historias de usuario definidas para el proyecto al principio del mismo, este conjunto de historias está sujeto a modificaciones a lo largo del desarrollo.

Tabla 5: *Product Backlog - Versión Inicial*

PRODUCT BACKLOG		Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas	
ID	Tarea	Prioridad	Esfuerzo estimado (Puntos de Historia)	
H001	Añadir Alumno	Must Have	3	
H002	Modificar Alumno	Must Have	3	
H003	Borrar Alumno	Must Have	1	
H004	Busqueda por DNI	Must Have	1	
H005	Añadir expediente	Must Have	3	
H006	Borrar expediente	Must Have	1	
H007	Modificar expediente	Must Have	3	
H008	Añadir examen	Must Have	2	
H009	Borrar examen	Must Have	1	
H010	Listar Expedientes de un alumno	Must Have	2	
H011	Listar Exámenes de un expediente	Must Have	2	

H012	Añadir vehículo	Should Have	3
H013	Modificar vehículo	Should Have	3
H014	Borrar vehículo	Should Have	1
H015	Añadir Operación de mantenimiento	Should Have	1
H016	Borrar Operación de mantenimiento	Should Have	1
H017	Listar Vehículos	Should Have	2
H018	Listar Operaciones de Mantenimiento	Should Have	2
H019	Generar Documento Contrato	Should Have	5
H020	Generar Documento Expediente	Should Have	5
H021	Generar Documento Vehículo	Should Have	5
H022	Añadir Profesor	Could Have	3
H023	Borrar Profesor	Could Have	1
H024	Modificar Profesor	Could Have	3
H025	Añadir parte	Would Have	3
H026	Borrar parte	Would Have	1
H027	Modificar parte	Would Have	3
H028	Buscar partes	Would Have	2
H029	Añadir pago	Could Have	3
H030	Borrar pago	Could Have	1
H031	Mostrar pagos	Could Have	1

PRODUCT BACKLOG		Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas
ID	Como	Quiero	Para
H001	Administrador	Añadir Alumnos	Realizar los registros de forma digita
H002	Administrador	Modificar Alumnos	Corregir y actualizar los datos del registro digital
H003	Administrador	Borrar Alumnos	Eliminar del registro los alumnos deseados
H004	Administrador	Buscar Alumnos por DNI	Encontrar Alumnos de forma rápida
H005	Administrador	Añadir Expedientes	Gestionar el proceso educativo de mis estudiantes
H006	Administrador	Borrar Expedientes	Poder corregir errores y organizar la información de los expedientes
H007	Administrador	Modificar Expedientes	Poder corregir errores y actualizar la información de los expedientes
H008	Administrador	Añadir Exámenes	Gestionar las evaluaciones de mis estudiantes
H009	Administrador	Borrar Exámenes	Poder corregir errores y organizar la información de los expedientes
H010	Administrador	Listar Expedientes	Poder ver la información referente a los expedientes de un alumno
H011	Administrador	Listar Exámenes	Poder visualizar la información referente a los exámenes de un expediente
H012	Administrador	Añadir Vehículos	Añadir un vehículo a mis flotas

H013	Administrador	Modificar Vehículos	Corregir y actualizar los datos de mi flota
H014	Administrador	Borrar Vehículos	Eliminar de la flota los vehículos deseados
H015	Administrador	Añadir Operaciones de mantenimiento	Llevar el control de la gestión de un vehículo concreto
H016	Administrador	Borrar Operaciones de mantenimiento	Corregir errores durante el control de la gestión de un vehículo concreto
H017	Administrador	Listar Vehículos	Ver todos los vehículos de mi flota
H018	Administrador	Listar Operaciones de mantenimiento	Ver el historial de las operaciones realizadas a un vehículo concreto
H019	Administrador	Generar Documento Contrato	Agilizar los procesos administrativos de la autoescuela
H020	Administrador	Generar un Documento de un Expediente	Tener un documento con la información de expediente de un alumno
H021	Administrador	Generar un Documento de un Vehículo	Tener un documento con la información de un vehículo de la flota
H022	Administrador	Añadir Profesores	Añadir mis profesores al registro digital
H023	Administrador	Borrar Profesores	Eliminar del registro los profesores deseados
H024	Administrador	Modificar Profesores	Corregir y actualizar los datos de mis profesores registrados
H025	Profesor	Añadir parte	Añadir al registro mis actividades docentes
H026	Profesor	Borrar parte	Eliminar del registro los partes deseados

H027	Profesor	Modificar parte	Corregir y actualizar los datos de mis partes registrados
H028	Profesor	Buscar partes	Acceder rápidamente a un parte deseado
H029	Administrador	Añadir pago	Llevar un control de los ingresos de mi autoescuela
H030	Administrador	Borrar pago	Eliminar del registro los pagos deseados
H031	Administrador	Mostrar pagos	Corregir y actualizar los datos de mis pagos registrados

Las historias de usuario han sido priorizadas usando el método MoSCoW, clasificándolas en cuatro órdenes de prioridad diferentes (*Must Have*, *Should Have*, *Could Have*, *Would Have*).

La estimación se ha realizado entre los miembros del equipo mediante comparación. Se ha empezado dando un valor a una tarea que estimamos de tamaño medio, (*modificar alumno*) y le hemos asignado un total de tres puntos de historia; después, el equipo ha ido estimando mediante el método *Delphi* las historias de usuario restantes.

El método *Delphi* consiste en un proceso prospectivo iterativo, utilizado para obtener información esencialmente cualitativa, pero relativamente precisa. Consiste en conseguir un consenso basado en la discusión entre varios expertos. Dicho consenso se obtiene a través de la elaboración de un cuestionario; en nuestro caso la estimación de cada historia de usuario. Los diferentes miembros del equipo proponen una estimación de acuerdo a su criterio y se muestran los datos resultantes en un histograma. Tras ver los resultados, los miembros que hayan dado los valores extremos defienden su postura ante el resto. Una vez expuestos los argumentos de cada postura, se vuelve a realizar una nueva pasada en la que cada miembro propone una estimación. El proceso se repite hasta alcanzar un consenso, pero si no se llega a un consenso, se tomará como estimación la media del histograma.

Una vez estimadas todas las historias obtenemos que el esfuerzo inicial del proyecto supone un total de 71 puntos de historia. Un punto de historia es una unidad relativa; para que cobre valor real deberemos fijar un número de puntos de historia objetivo a realizar por iteración. Este número se puede ir ajustando a medida que se vaya desarrollando cada iteración hasta encontrar la velocidad real del equipo. Sin embargo, antes de empezar el proyecto no contamos con información relevante como para dar un valor real, por lo que habrá que estimar una velocidad inicial. A la hora de estimar esta velocidad debemos tener en cuenta:

1. Cada historia de usuario implica el diseño y la implementación de tres aplicaciones diferentes en paralelo: Cliente de sobremesa, cliente móvil y servidor, cada uno utilizando tecnologías distintas.
2. La duración de cada iteración será de dos semanas, sin contar la reunión con el cliente, que se realizará durante la semana de la iteración siguiente.

De acuerdo a los puntos anteriores y el desempeño del equipo durante anteriores proyectos realizados durante las prácticas en la universidad, concluimos que la velocidad media del equipo será de unos 15 puntos de historia por iteración.

Como hemos puntualizado al principio de este capítulo, el desarrollo ágil fija un plazo para el proyecto, permaneciendo el coste y el ámbito variables. Se decide dividir el proyecto en cinco iteraciones de dos semanas de duración, dando un total de diez semanas de duración del proyecto. Trabajando 5 horas diarias, supone 250 horas de trabajo, ajustándose a la restricción impuesta por los doce créditos correspondientes a este TFG.

1.11 Presupuesto

La cantidad y disponibilidad de recursos disponible para el proyecto es controlable y constante durante todo el transcurso del proyecto, por lo que la elaboración del Plan de Costes se simplifica bastante. No hará falta descomponer los diferentes recursos necesarios para cada historia de usuario, ya que los recursos disponibles serán los mismos durante todo el proyecto.

1.11.1 *Gastos derivados del personal*

El equipo de desarrollo se compone de dos miembros solamente, y debido al reducido tamaño del equipo, cada miembro cumple con varias funciones dentro del

proyecto incluyendo análisis, diseño, implementación y pruebas. Como guía a la hora de establecer el salario del personal, se ha tomado como referencia el convenio *XVII Convenio colectivo estatal de empresas de consultoría, y estudios de mercados y de la opinión pública* con código de convenio 99001355011983; publicado en el BOE a 6 de marzo de 2018. Este convenio está en vigor hasta el 31 de diciembre del 2019, prorrogándose a partir de dicha fecha por tácita aceptación mientras no inicie su vigencia el Convenio Colectivo Estatal que lo sustituya.

La Tabla 6 desglosa las funciones de cada miembro del equipo dentro del proyecto y su sueldo por hora.

Tabla 6: Personal

Miembro	Funciones	Sueldo (€/hora)
Javier Martínez Cavanillas	Coordinador de proyecto	15
	Analista	
	Diseñador	
	Programador Full Stack	
Alberto Carrillo Ortega	Control de calidad	14
	Analista	
	Diseñador	
	Programador Full Stack	

El plazo establecido para el proyecto es de un total de diez semanas, trabajando cinco horas al día. En la Tabla 7 se muestra el sueldo total tras concluir todo el proyecto.

Tabla 7: Coste de personal

Miembro	Coste (€/hora)	Horas de trabajo	Coste total
Javier Martínez Cavanillas	15	250	3750 €
Alberto Carrillo Ortega	14	250	3700 €
Total			7250 €

1.11.2 Materiales

1.11.2.1 Materiales Hardware

Para estimar el coste de los equipos se ha calculado la amortización de su precio durante las diez semanas de duración en base al coste en el momento de adquisición. Los equipos son de gama alta, por lo que se les estima una vida útil mínima de 4 años. Sin embargo, para el dispositivo móvil donde se van a realizar las pruebas se estima una vida útil de 3 años. La Tabla 8 resume el coste derivado de la amortización de los equipos durante el tiempo que dura el proyecto.

Tabla 8: Coste del hardware

Hardware					
Nombre	Precio	Unidades	Coste	Semanas	Total
MSI Apache Pro GE62 6QF	1.631,00 €	1	7,84 €	10	78,41 €
Asus Rog Strix Scar Edition GL703GM-E5016	1.100,00 €	1	5,29 €	10	52,88 €
Bq Aquaris X	232,00 €	1	1,12 €	10	11,15 €
Total					142,45 €

1.11.2.2 Materiales Software

En los materiales se ha conseguido minimizar costes abogando en la medida de lo posible por el uso de software libre y gratuito. El principal coste en este sentido lo supone la licencia de Visual Paradigm Standard Edition, proporcionada por la Universidad de Jaén. En la Tabla 9 se recoge el costo de las principales licencias software utilizadas.

Tabla 9: Coste del Software

Software			
Nombre	Unidades	Coste	Total
Windows 10	2	0,00 € (Incluido en el coste del equipo)	0,00 €

LibreOffice	2	0,00 €	0,00 €
PostgreSQL	2	0,00 €	0,00 €
Apache Netbeans	2	0,00 €	0,00 €
Visual Paradigm	2	60,00 €	120,00 €
Android Studio	1	0,00 €	0,00 €
Gitlab	2	0,00 €	0,00 €
Open JDK	2	0,00 €	0,00 €
Total			120,00 €

1.11.3 Costes indirectos

En esta sección se desglosan los costes derivados del consumo de luz y agua, servicio de limpieza, red de telefonía y acceso a internet. La Tabla 10 desglosa el coste indirecto supuesto por dichos gastos.

Tabla 10: Costes indirectos

Costes indirectos	
Concepto	Total
Agua	20,00 €
Luz	200,00 €
Servicio de limpieza	190,00 €
Internet	156,00 €
Total	566,00 €

1.11.4 Presupuesto final

Aplicando como beneficio un 20% del coste del proyecto y una reserva para contingencias del 8% obtenemos el presupuesto final (Tabla 11).

Tabla 11: Presupuesto final

Resumen		
Coste del personal	7.250,00 €	
Coste del hardware	145,45 €	
Coste del software	120,00 €	
Costes indirectos	626,00 €	
Reserva de contingencia	651,32 €	
Margen de beneficio	1.628,29 €	
Presupuesto		
Concepto	Precio sin IVA	Precio con IVA (21%)
Desarrollo aplicación de Gestión Autoescuelas Zona L	10.421,06 €	12.609,48 €

1.12 Normas y referencias

1.12.1 Disposiciones legales y normas aplicadas

El sistema albergará información personal de los alumnos registrados en la autoescuela, por lo que el proyecto debe desarrollarse teniendo en cuenta la normativa dispuesta en la *Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos Personales y garantía de los derechos digitales*. La aplicación debe proporcionar las características necesarias para cumplir con la normativa:

1. Protección de datos personales de carácter confidencial (contraseñas) mediante cifrado en la base de datos.
2. Asegurar la integridad, confidencialidad, disponibilidad y resiliencia de los datos en el ámbito de actuación correspondiente a la aplicación.
3. Derecho y capacidad a eliminar datos no deseados de la base de datos.
4. Comunicación segura entre el servidor y los respectivos clientes para el traspaso de información confidencial.

Una vez la aplicación cumpla con estos requisitos la responsabilidad final del tratamiento de los datos recae sobre la empresa gerente de los datos, que deberá

mantener un plan de trabajo con respecto a la transparencia, recolección, tratamiento y mantenimiento de datos personales acorde a la ley.

2 INICIO DEL PROYECTO

2.1 Planificación inicial del proyecto

Tras obtener el *Product Backlog* decidimos generar un diagrama WBS que nos ayude a tener una visión más concisa de las tareas a realizar durante el proyecto, al menos durante la etapa inicial del mismo. El diagrama WBS representa las diferentes historias de usuario agrupadas de manera semántica. Los nodos finales de este diagrama son llamados paquetes de trabajo, y deben ser tareas lo suficientemente concretas como para estimarse fácilmente, deben poder asignarse a una sola persona y tienen que tener asociado algún tipo de entregable, ya sea la entrega de código, un informe o la aprobación de un documento.

En un diagrama WBS ágil las historias de usuario iniciales del proyecto son aquellas que resultaron prioritarias y fueron aclaradas con detalle en la fase inicial de análisis y entrevistas con el cliente. Por tanto, disponemos de más información sobre las mismas y podemos desglosar en profundidad las tareas necesarias mientras que los últimas historias de usuario no quedan definidas con suficiente precisión por el cliente y no disponemos de información suficiente para desglosar las tareas en profundidad. Además, en una metodología ágil no conviene perder el tiempo en planificar tareas a largo plazo, puesto que el ámbito es cambiante y existe la posibilidad de que algunas historias de usuario del proyecto no se lleven a cabo o sean sustituidas por otros más útiles o atractivas para el cliente.

La Figura 4 muestra el diagrama WBS desglosando las historias de usuario en tareas mas pequeñas.

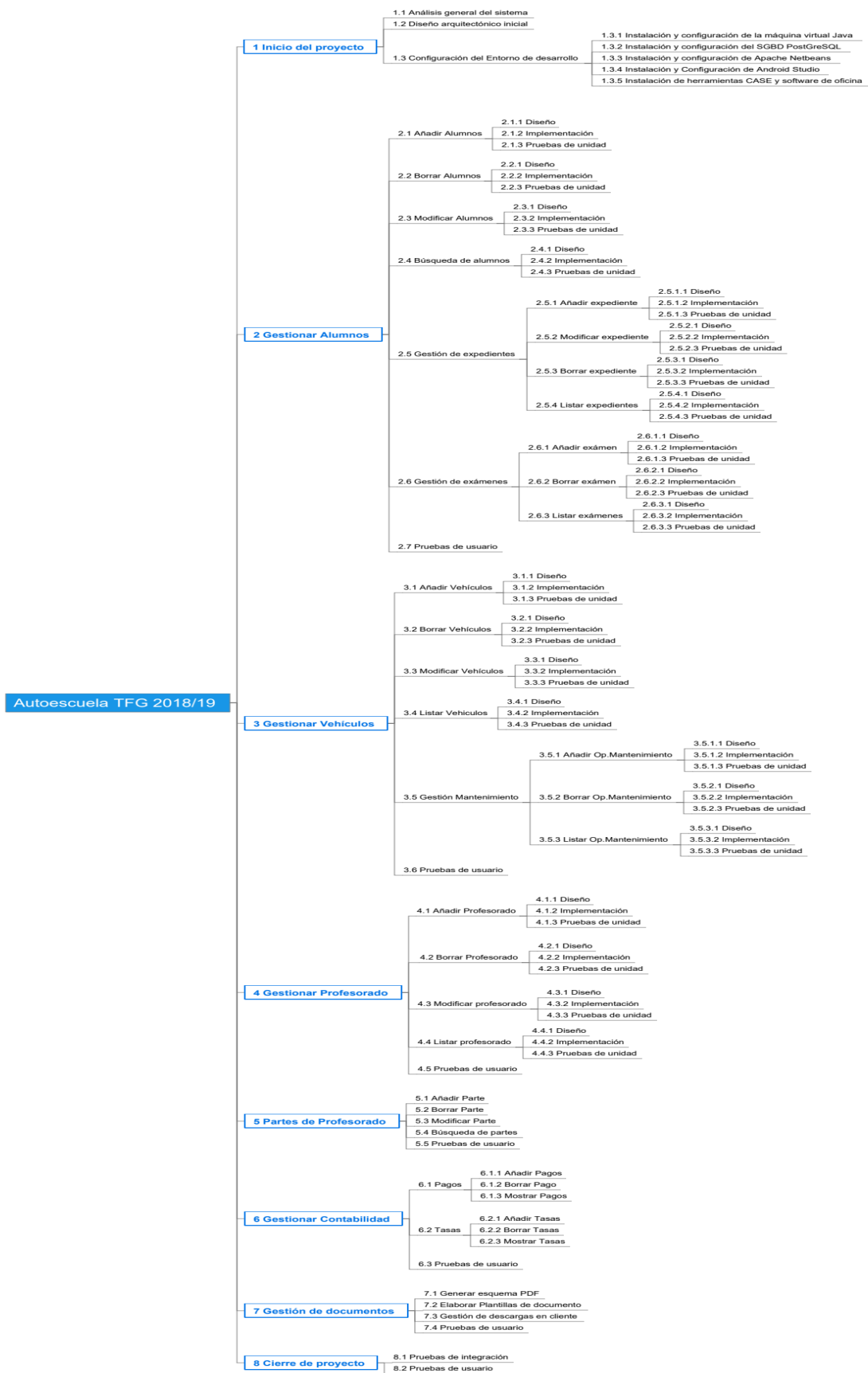


Figura 4: Diagrama WBS - Autoescuela Zona L

2.2 Análisis inicial del sistema

En la lógica sistema intervienen diferentes entidades comunes en un ambiente como el de una autoescuela. Estas entidades son traducidas y representadas mediante las diferentes clases del proyecto.

Estas son algunas de las entidades que caracterizan el sistema:

A) Alumno:

- i. Representa a los alumnos de la autoescuela. Contiene la información necesaria (DNI, nombre, teléfono...) así como otra menos necesaria (Nombre de los padres). Es una de las entidades más relevantes del sistema.
- ii. Se relaciona con la entidad expediente de modo que un alumno puede tener varios expedientes.

B) Expediente

- i. Representa un curso dentro de la autoescuela. Un curso viene a ser el proceso que sigue un alumno para obtener un tipo de carnet de conducir (A1, A2...) y contiene información relativa a este; por ejemplo, la fecha en la que se matriculó el alumno, si ha entregado alguna foto o los documentos correspondientes para el carnet en cuestión o el estado en el que se encuentra el expediente.
- ii. Se relaciona con la entidad alumno de modo que un expediente puede ser de un solo alumno.
- iii. Se relaciona con la entidad examen de modo que un expediente puede tener varios exámenes.

C) Examen

- i. Representa los exámenes que se realizan durante el proceso de obtención de un carnet (Teórico, Práctico, de Maniobras). Es una entidad simple que solo se caracteriza por su fecha, tipo y si el alumno lo completó con éxito o no.
- ii. Se relaciona con la entidad expediente de modo que un examen solo puede pertenecer a un expediente.

D) Vehículo

- i. Simboliza los diferentes automóviles que conforman la flota de la autoescuela. Son descritos por información relevante a ellos tal como su matrícula, su marca y modelo. Además también almacenan datos útiles como la fecha en la que ha pasado la ITV o en la que esta caduca.
- ii. Se relaciona con la entidad operaciones de mantenimiento de modo que un vehículo puede tener varias operaciones de mantenimiento.

E) Operación de mantenimiento

- i. Simboliza las diferentes operaciones que experimenta un vehículo de la flota durante su pertenencia en la autoescuela. Contiene información sobre la fecha en la que ocurrieron y en qué consisten. Ejemplos de operaciones de mantenimiento van desde un simple cambio de aceite hasta reparaciones más serias del automóvil.
- ii. Se relaciona con la entidad vehículo de modo que una operación de mantenimiento solo puede pertenecer a un vehículo.

2.3 Diseño Arquitectónico preliminar

El diseño del sistema irá evolucionando a medida que se desarrollen las diferentes iteraciones que lo componen. No obstante, la naturaleza de la aplicación, lenguajes y marcos de trabajo utilizados permiten fijar una serie de criterios arquitectónicos generales. En este capítulo se expondrá el tipo de arquitectura en la que se enmarca nuestra aplicación y los principios utilizados a la hora de su desarrollo.

2.3.1 Arquitectura del sistema

El sistema conforma una arquitectura cliente – servidor, donde los clientes se encargarán de mostrar la información mientras que el servidor se hará cargo de la lógica de negocio asociada y la gestión de datos.

La aplicación se desarrollará basándose en los principios de la programación orientada a objetos (OOP). Entre las razones por las cuales tomamos esta decisión podemos destacar:

- Es la más utilizada actualmente en la mayoría de proyectos software del mercado.
- Facilita el proceso de documentación y comprensión a la hora de un futuro mantenimiento por parte de terceros
- El equipo de desarrollo tiene experiencia trabajando según este estilo arquitectónico.

Así mismo, los diferentes objetos del sistema se estructurarán en una arquitectura de capas y particiones formada por 5 capas. Cada capa se encargará de una función diferente dentro del sistema, desacoplando en la medida de lo posible los diferentes componentes que lo conforman. Parte de estos componentes forman parte de las bibliotecas y tecnologías externas utilizadas en el desarrollo y detalladas anteriormente.

1. **Capa de Presentación:** se encarga de controlar la interfaz visible por el usuario. Es la única implementada en el lado de los clientes. Pueden considerarse dos vistas dentro de esta capa, vista para clientes de sobremesa y vista para dispositivos móviles.
2. **Capa de Servicios:** alojada en el servidor, se encarga de establecer la comunicación entre los clientes y el servidor estableciendo una API basada en REST.
3. **Capa de Lógica de Negocio:** aquí se implementará la lógica principal de la aplicación, tratamiento de datos, autenticación, etc. Las características de la aplicación se implementarán de manera modular, permitiendo que los diferentes módulos puedan trabajar de manera independiente en la medida de lo posible.
4. **Capa de Persistencia:** Implementa el mapeado ORM entre los objetos utilizados en la capa de negocio y las tablas de datos utilizadas en la base de datos. Esta capa está gestionada por *Hibernate*.
5. **Sistema de Gestión de Bases de datos (SGBD):** Organiza, clasifica y mantiene la integridad entre los distintos datos almacenados por el sistema. Utilizaremos PostgreSQL para realizar esta función.

La Figura 5 muestra un diagrama de la arquitectura preliminar del sistema.

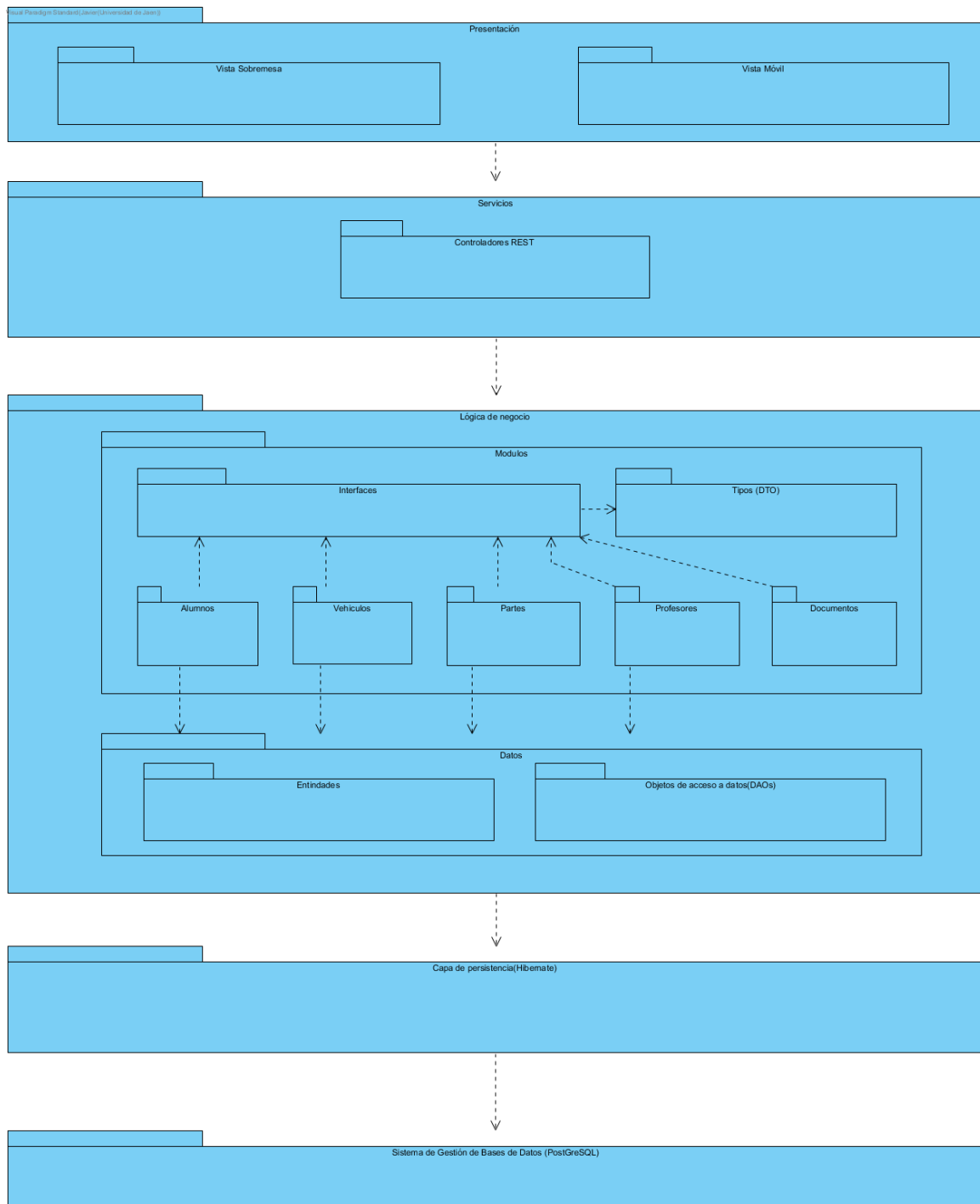


Figura 5: Arquitectura Preliminar – Estructura de Capas y Particiones

2.3.2 Servidor

En el servidor se implementarán las capas de acceso a servicios, lógica de negocio, persistencia y almacenamiento de datos. Las capas más relevantes de cara a la implementación propia del proyecto son la lógica de negocio y la de servicios. Cada controlador REST de la capa de servicios accederá a uno o más módulos en la lógica de negocio y a su vez estos accederán a los diferentes objetos de negocio que conforman el modelo y sus relaciones. La gestión de concurrencia e integridad transaccional se realizará en los módulos, proporcionando una mayor flexibilidad a la hora de realizar operaciones más complejas sobre los objetos.

La Figura 6 muestra un pequeño esquema de la capa de negocio del servidor.

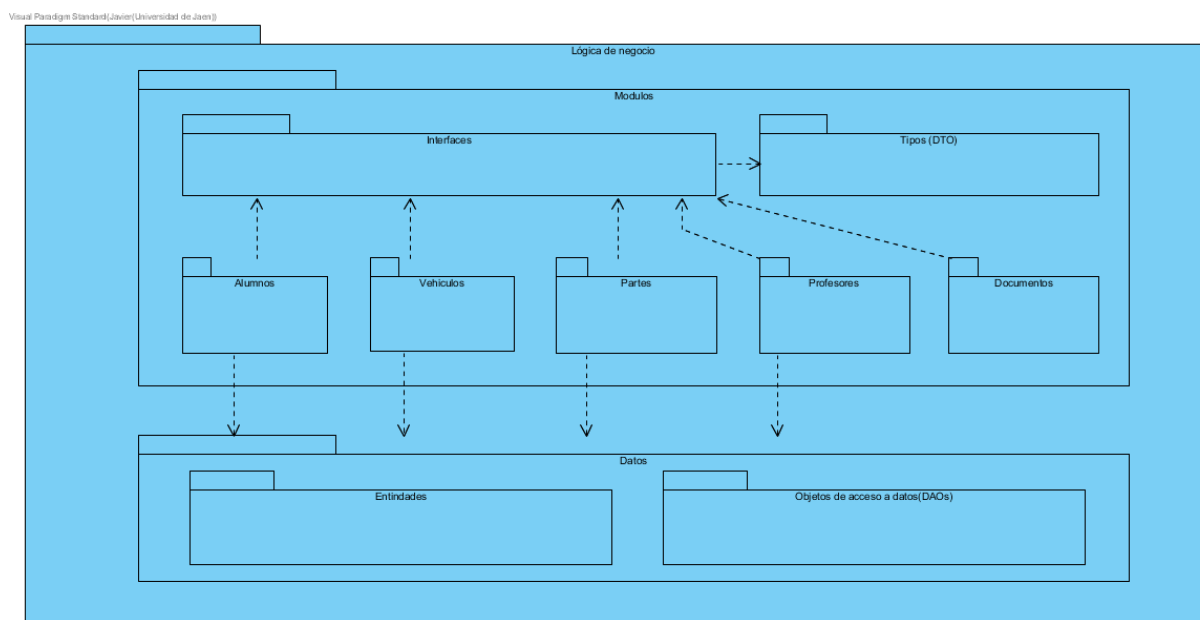


Figura 6: Arquitectura Preliminar – Lógica de negocio

2.3.3 Cliente Móvil

El cliente móvil consistirá en una aplicación Android que sirva como interfaz con el servidor. Se realizará siguiendo el paradigma Android de *Single Activity App*. El cual consiste en que solo habrá una actividad pero se subdivide en elementos más pequeños llamados *Fragment* (fragmentos).

Estos fragmentos serán modulares y cada uno (o cada pequeño subgrupo de ser necesario) estarán destinados a resolver un solo caso de uso o en su defecto un grupo reducido y muy relacionado de estos. En otras palabras, La actividad

mantiene un entorno de trabajo común y cada fragmento realiza una interacción y una labor concreta e independiente.

Así mismo debido a la necesidad de aprovechar recursos en el dispositivo móvil las llamadas a el servidor se harán a mediante métodos estáticos localizados en una clase abstracta (*ApiLink*). Lo cual nos garantiza el poder realizar las peticiones necesarias desde cualquier fragmento sin preocuparnos por acoplamiento ni instanciación de clases auxiliares. Esta clase usará a si mismo otra clase llamada *ErrorDialogo* que implementa la interfaz *Interceptor* y obtendrá los posibles errores que nos devuelva el servidor para más tarde informar al usuario de ellos.

La navegación entre fragmentos nos la proporcionará Android pero para ello será necesario ciertos atributos de la actividad principal.

En la Figura 7 se muestra un esquema ilustrando lo anteriormente mencionado.

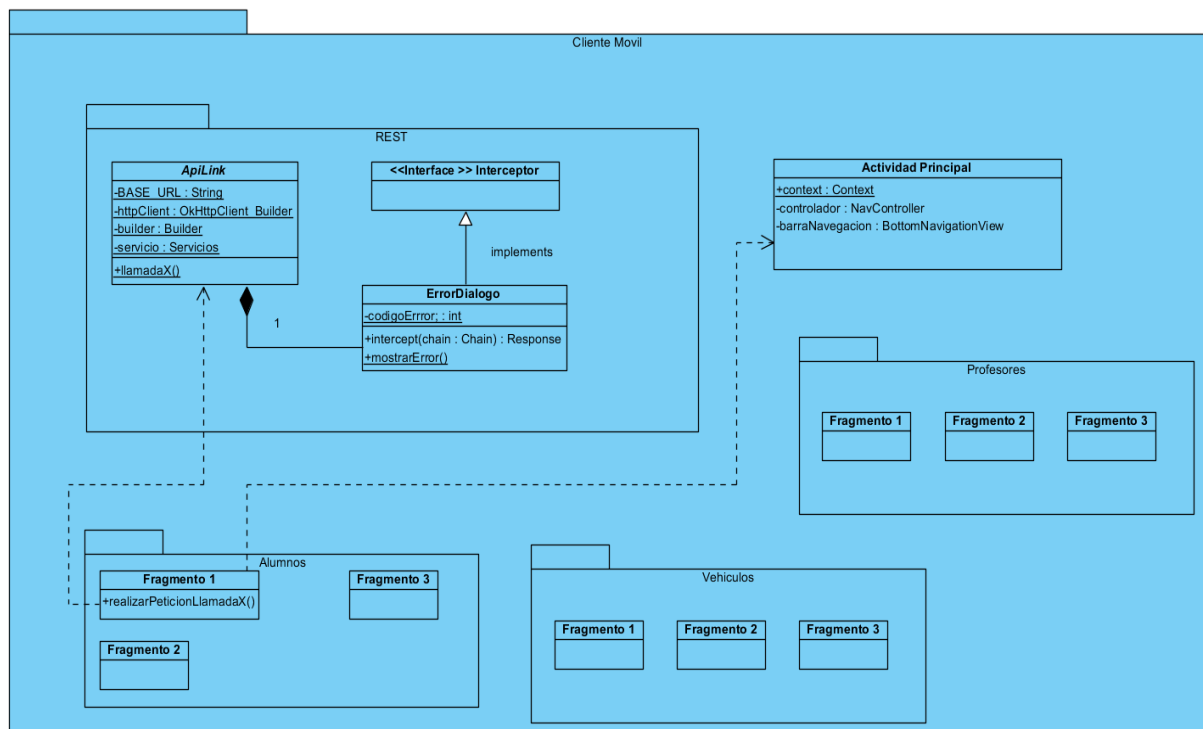


Figura 7: Arquitectura Preliminar – Cliente Móvil

3 DESARROLLO DEL PROYECTO

3.1 Primera Iteración

Tras la puesta a punto de la estructura base y el debido proceso de documentación sobre las nuevas tecnologías utilizadas, la primera iteración se compone de la inclusión de los alumnos en el sistema.

3.1.1 Historias de usuario

En las tablas Tabla 12, Tabla 13, Tabla 14 y Tabla 15 se muestran las historias de usuario de esta iteración.

Tabla 12: H001 Añadir Alumno

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H001	Estimación 3	Prioridad Must Have	
Como Administrador	Quiero Añadir Alumnos	Para Realizar los registros de forma digital	
Criterios de satisfacción - Dado que se han introducido correctamente los datos del alumno, se cree una entrada en el registro			
Redactado por Alberto Carrillo Ortega	Fecha 23/05/2019	Aprobado por Javier Martínez Cavanillas	Fecha 23/05/2019

Tabla 13: H002 Modificar Alumno

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H002	Estimación 3	Prioridad Must Have	
Como Administrador	Quiero Modificar Alumnos	Para Corregir y actualizar los datos del registro digital	
Criterios de satisfacción - Dado que se han modificado correctamente los datos del alumno, se modifique su entrada en el registro			
Redactado por Alberto Carrillo Ortega	Fecha 23/05/2019	Aprobado por Javier Martínez Cavanillas	Fecha 23/05/2019

Tabla 14: H003 Borrar Alumno

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H003	Estimación 0.5	Prioridad Could Have	
Como Administrador	Quiero Borrar Alumnos	Para Eliminar del registro los alumnos deseados	
Criterios de satisfacción - Dado que se ha borrado el alumno correctamente, se borre su entrada del registro y los expedientes relacionados con el			
Redactado por Alberto Carrillo Ortega	Fecha 23/05/2019	Aprobado por Javier Martínez Cavanillas	Fecha 23/05/2019

Tabla 15: H004 Búsqueda por DNI

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H004	Estimación 1	Prioridad Must Have	
Como Administrador	Quiero Buscar Alumnos por DNI	Para Encontrar Alumnos de forma rápida	
Criterios de satisfacción - Dado que se ha introducido un DNI existente, se muestre la entrada correspondiente a el alumno con ese DNI			
Redactado por Alberto Carrillo Ortega	Fecha 23/05/2019	Aprobado por Javier Martínez Cavanillas	Fecha 23/05/2019

3.1.2 Análisis

En las figuras: Figura 8, Figura 9, Figura 10 y Figura 11 se muestran los diagramas de actividad referentes a las historias de usuario del apartado anterior.

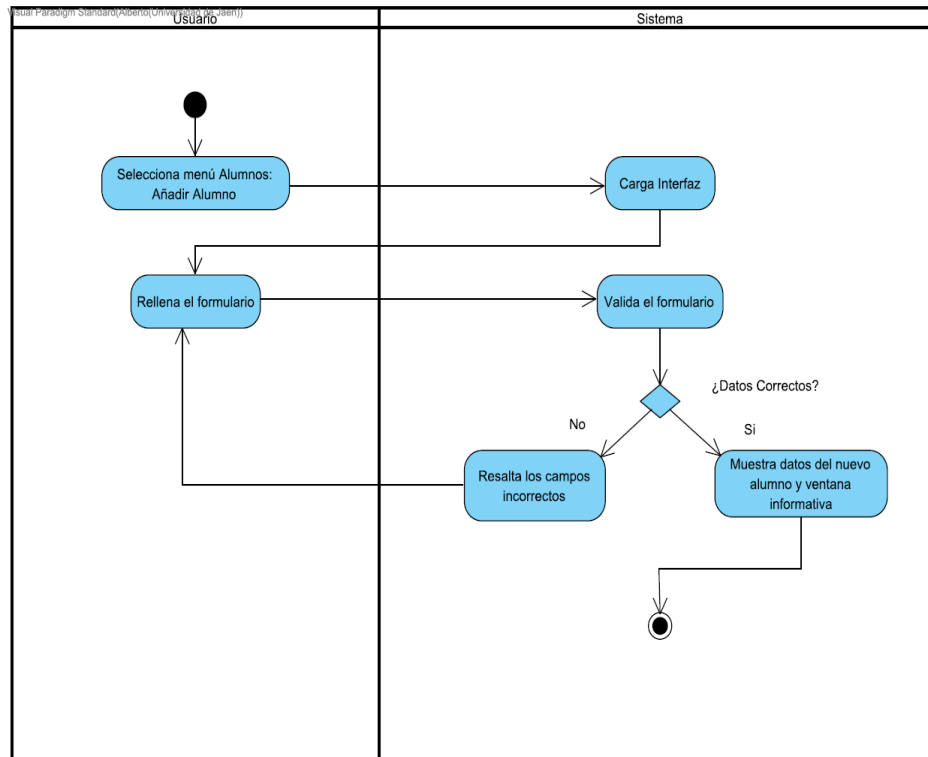


Figura 8: DA001 Añadir Alumno (H001)

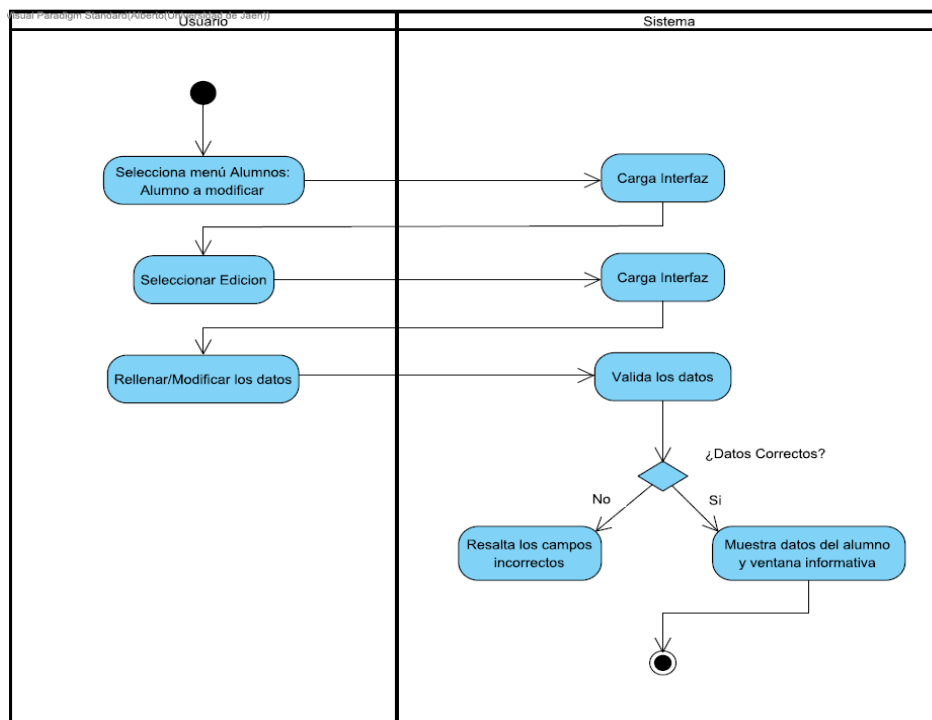


Figura 9: DA002 Modificar Alumno (H002)

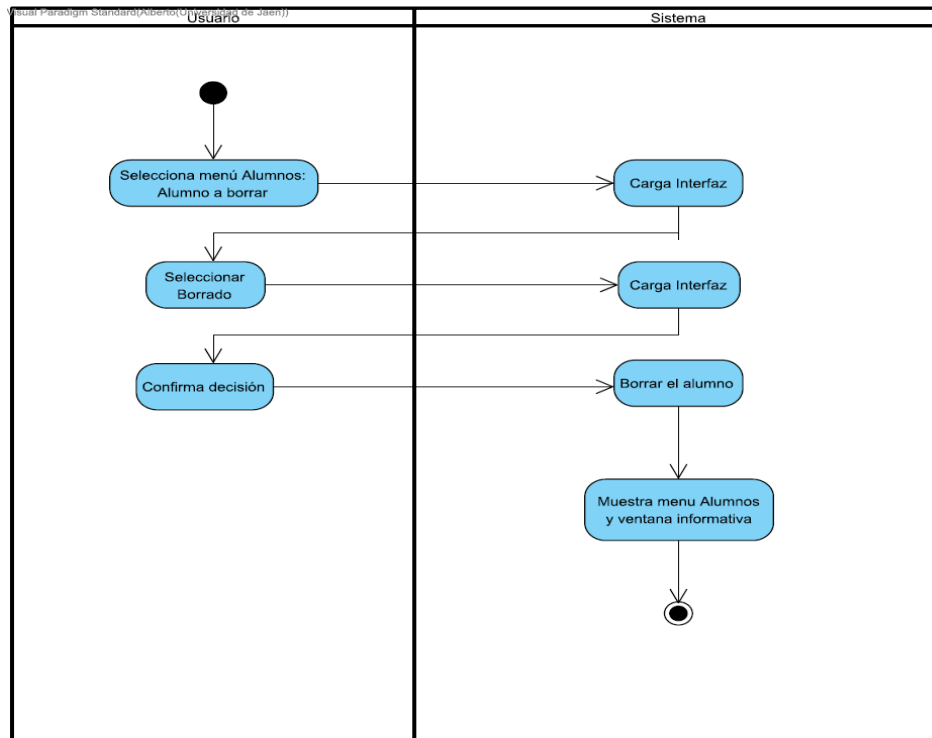


Figura 10: DA003 Borrar Alumno (H003)

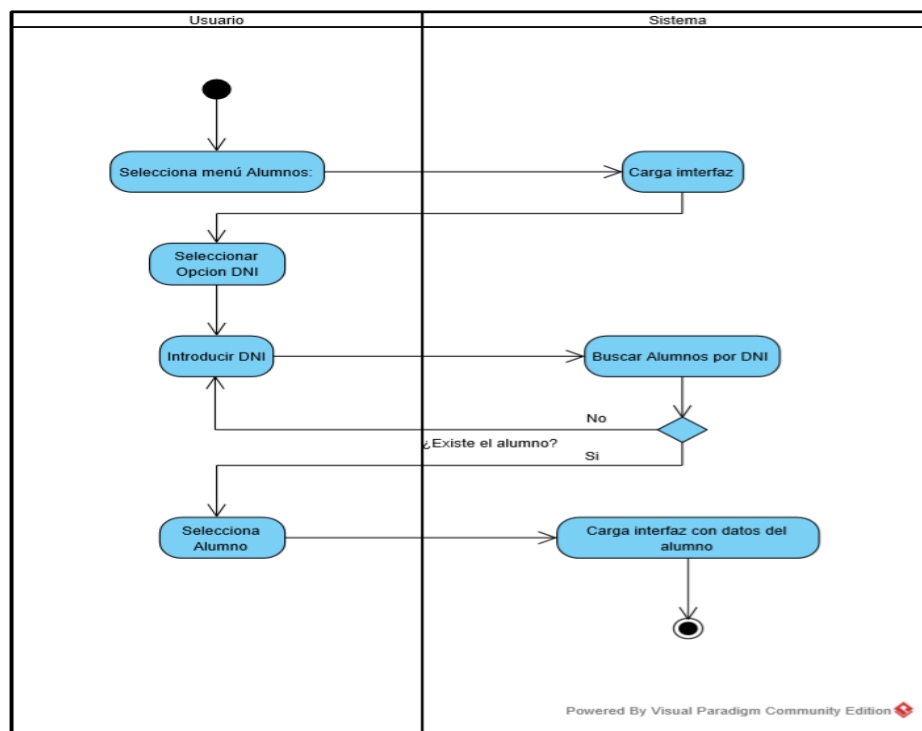


Figura 11: DA004 Búsqueda de alumnos por DNI (H004)

3.1.3 Planificación de la iteración

Esta iteración es un poco diferente a los demás, ya que contiene tareas asociadas a la etapa de inicio del proyecto y no se pueden asociar a una historia de

usuario concreta, puesto que contribuyen al desarrollo de todas. Estas tareas son las citadas a continuación:

1. Análisis general del sistema: detallado en el apartado 2.1 de esta memoria.
2. Diseño arquitectónico: detallado en el apartado 2.2 de esta memoria.
3. Instalación y configuración de la máquina virtual Java: La máquina virtual OpenJDK no dispone de un instalador y necesita ser configurada manualmente en el equipo.
4. Instalación y configuración del SGBD PostgreSQL: Instalación de PostgreSQL, instalación de pgAdmin 4 y creación de un servidor base para el uso de la aplicación.
5. Instalación y configuración de Apache NetBeans: Instalación de Apache NetBeans 10 en los equipos y configuración del mismo para utilizar la versión de la maquina virtual deseada. Configuración común de los encabezados del proyecto e información de licencia. Configuración de las variables de entorno necesarias para ejecutar el proyecto JavaFX.
6. Instalación y Configuración de Android Studio
7. Instalación de herramientas CASE y software de oficina: Instalación y activación de la licencia Visual Paradigm Standard Edition en los equipos. Instalación del la suite ofimática LibreOffice y otras herramientas de desarrollo como Git.

3.1.3.1 *Proceso de documentación previa*

En el diagrama WBS no se ha reflejado el proceso de documentación de los miembros sobre las diferentes tecnologías asociadas al proyecto como una tarea, puesto que resulta difícil estimar el esfuerzo dedicado por cada miembro a ello y no genera ningún entregable. Sin embargo, es conveniente destacar que el equipo ha invertido una parte de su tiempo durante esta iteración a informarse sobre las prácticas adecuadas y correcto uso de dichas tecnologías, sobre todo aquellas con las que el equipo no está familiarizado anteriormente; JavaFX y Android.

En la Tabla 16 se muestra un esquema del *Backlog de Iteración* de la primera iteración.

Tabla 16: Backlog de iteración – Iteración 1

BACKLOG DE ITERACIÓN	Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas
Iteración Iteración 1 – Alumnos		Puntos de Historia 15
ID	Tarea	Proyecto
Inicio de proyecto		
T0001	Análisis general del sistema	
T0002	Diseño Arquitectónico del sistema	
T0003	Instalación y configuración de la máquina virtual Java	
T0004	Instalación y configuración del SGBD PostgreSQL	
T0005	Instalación y configuración de Apache Netbeans	
T0006	Instalación y Configuración de Android Studio	
T0007	Instalación de herramientas CASE y software de oficina	
H001 Añadir Alumno		
T0101	Diseño añadir alumno	Servidor
T0102	Diseño añadir alumno	Cliente de escritorio
T0103	Diseño añadir alumno	Cliente Móvil
T0104	Implementación añadir alumno	Servidor
T0105	Implementación añadir alumno	Cliente Móvil
T0106	Implementación añadir alumno	Cliente de escritorio
H002 Modificar Alumno		
T0107	Diseño modificar alumno	Servidor
T0108	Diseño modificar alumno	Cliente de escritorio
T0109	Diseño modificar alumno	Cliente Móvil
T0110	Implementación modificar alumno	Servidor

T0111	Implementación modificar alumno	Cliente Móvil
T0112	Implementación modificar alumno	Cliente de escritorio
H003 Borrar Alumno		
T0113	Diseño borrar alumno	Servidor
T0114	Diseño borrar alumno	Cliente de escritorio
T0115	Diseño borrar alumno	Cliente Móvil
T0116	Implementación borrar alumno	Cliente de escritorio
T0117	Implementación borrar alumno	Cliente Móvil
T0118	Implementación borrar alumno	Servidor
H004 Búsqueda por DNI		
T0119	Diseño búsqueda por DNI	Servidor
T0120	Diseño búsqueda por DNI	Cliente de escritorio
T0121	Diseño búsqueda por DNI	Cliente Móvil
T0122	Implementación búsqueda de alumno por DNI	Servidor
T0123	Implementación búsqueda de alumno por DNI	Cliente de escritorio
T0124	Implementación búsqueda de alumno por DNI	Cliente Móvil
H005 Añadir Expediente		
T0125	Diseño añadir expediente	Servidor
T0126	Diseño añadir expediente	Cliente de escritorio
T0127	Diseño añadir expediente	Cliente Móvil
T0128	Implementación añadir expediente	Servidor
T0129	Implementación añadir expediente	Cliente de escritorio
T0130	Implementación añadir expediente	Cliente Móvil
H006 Borrar Expediente		
T0131	Diseño borrar expediente	Servidor
T0132	Diseño borrar expediente	Cliente de escritorio

T0133	Diseño borrar expediente	Cliente Móvil
T0134	Implementación borrar expediente	Servidor
T0135	Implementación borrar expediente	Cliente de escritorio
T0136	Implementación borrar expediente	Cliente Móvil
H007 Modificar Expediente		
T0137	Diseño modificar expediente	Servidor
T0138	Diseño modificar expediente	Cliente de escritorio
T0139	Diseño modificar expediente	Cliente Móvil
T0140	Implementación modificar expediente	Servidor
T0141	Implementación modificar expediente	Cliente de escritorio
T0142	Implementación modificar expediente	Cliente Móvil
T0143	Pruebas de unidad	Servidor

3.1.4 Diseño

Las tareas de diseño se han ido realizándose de forma incremental. Sin embargo, para un mejor entendimiento del diseño de la aplicación en este apartado se recoge el trabajo de las tareas de diseño pertenecientes a esta iteración y la siguiente, teniendo en cuenta las relaciones completas con elementos del sistema como son los expedientes y los exámenes, pertenecientes al diseño de las historias de **H001, H002, H003, H004, H005, H006, H007, H008, H009, H010 y H011**.

3.1.4.1 Servidor

El modelo del servidor sigue un sistema sencillo. Disponemos de tres tipos de objetos de negocio: *Alumno*, que contiene la información personal del alumno registrado, *Expediente*, un alumno podrá poseer varios expedientes, en los que se albergará información sobre la fecha de apertura del expediente, estado y exámenes asociados; por último, *Examen*, que representará la fecha, tipo y calificación del examen.

Aquí en Figura 12 se muestra el esquema de el módulo de alumnos.

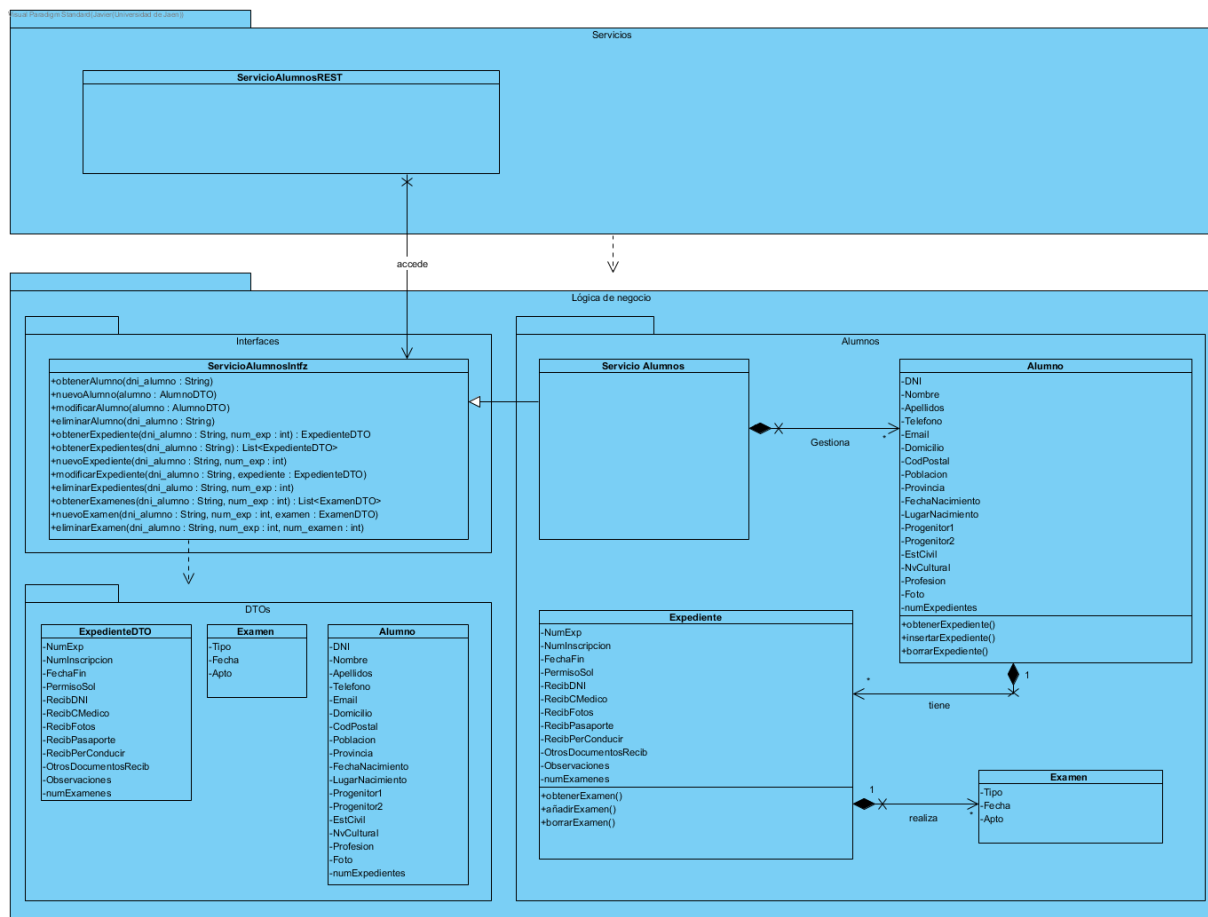


Figura 12: Diagrama de Clases – Alumnado, Lógica de negocio, Servidor

El controlador *ServicioAlumnosREST* expone al exterior la interfaz que se describe en la Tabla 17.

Tabla 17: Especificación API REST - Alumnado

ESPECIFICACIÓN API REST	Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas
<i>Obtener Alumno</i>	URL Recurso	Método
	<i>/alumnos/{dni_alumno}</i>	GET
	Parámetros	Tipo
	dni_alumno : String	Path
<i>Registrar Alumno</i>	URL Recurso	Método
	<i>/alumnos/{dni_alumno}</i>	POST
	Parámetros	Tipo
	dni_alumno : String	Path

	alumno : AlumnoDTO	RequestBody
<i>Modificar Alumno</i>	URL Recurso	Método
	<i>/alumnos/{dni_alumno}</i>	PUT
	Parámetros	Tipo
	dni_alumno : String	Path
<i>Borrar Alumno</i>	URL Recurso	Método
	<i>/alumnos/{dni_alumno}</i>	DELETE
	Parámetros	Tipo
	dni_alumno : String	Path
<i>Obtener Expedientes</i>	URL Recurso	Método
	<i>/alumnos/{dni_alumno}/expedientes</i>	GET
	Parámetros	Tipo
	dni_alumno : String	Path
<i>Añadir Expediente</i>	URL Recurso	Método
	<i>/alumnos/{dni_alumno}/expedientes/{num_exp}</i>	POST
	Parámetros	Tipo
	dni_alumno : String	Path
	num_exp : int	Path
	Expediente : ExpedienteDTO	RequestBody
<i>Modificar Expediente</i>	URL Recurso	Método
	<i>/alumnos/{dni_alumno}/expedientes/{num_exp}</i>	PUT
	Parámetros	Tipo
	dni_alumno : String	Path
	num_exp : int	Path
	Expediente : ExpedienteDTO	RequestBody
<i>Borrar Expediente</i>	URL Recurso	Método
	<i>/alumnos/{dni_alumno}/expedientes/{num_exp}</i>	DELETE
	Parámetros	Tipo

	dni_alumno : String	Path
	num_exp : int	Path
<i>Obtener Exámenes</i>	URL Recurso	Método
	/alumnos/{dni_alumno}/expedientes/{num_exp}/exámenes	GET
	Parámetros	Tipo
	dni_alumno : String	Path
	num_exp : int	Path
<i>Añadir Examen</i>	URL Recurso	Método
	/alumnos/{dni_alumno}/expedientes/{num_exp}/exámenes/{num_examen}	POST
	Parámetros	Tipo
	dni_alumno : String	Path
	num_exp : int	Path
	num_examen : int	Path
	Examen : ExamenDTO	RequestBody
<i>Modificar Examen</i>	URL Recurso	Método
	/alumnos/{dni_alumno}/expedientes/{num_exp}/exámenes/{num_examen}	PUT
	Parámetros	Tipo
	dni_alumno : String	Path
	num_exp : int	Path
	num_examen : int	Path
	Examen : ExamenDTO	RequestBody
<i>Modificar Examen</i>	URL Recurso	Método
	/alumnos/{dni_alumno}/expedientes/{num_exp}/exámenes/{num_examen}	DELETE
	Parámetros	Tipo
	dni_alumno : String	Path

	num_exp : int	Path
	num_examen : int	Path

3.1.4.2 Cliente Móvil

En esta primera iteración me centré en realizar un diseño extensible a todos los fragmentos necesarios para el desarrollo del módulo de alumnado. El cual abarca desde la creación de alumnos hasta el borrado de exámenes.

La mayoría de los fragmentos siguen este esquema:

- 1) En *onCreateView()* se carga la vista almacenada en su XML (cada fragmento tiene un XML que contiene su vista con el formato *frag_nombre_fragmento*) y puebla los campos de esta con los datos necesarios (Dtos de las diferentes clases) obtenidos a través de el controlador de navegación.
- 2) El usuario rellena los campos y pulsa el botón que realiza la petición.
- 3) La aplicación recoge los datos de los campos rellenos por el usuario, genera la clase o clases necesarias en el contexto dado y realiza una petición al servidor.
- 4) En este punto bloqueamos la interacción del usuario con la aplicación hasta la llegada de la respuesta asíncrona del servidor. De esto se encarga la clase *BarraProgreso*, que muestra un indicador al usuario de que ha de esperar.
- 5) Recibimos la respuesta del servidor a través de una interfaz. En este caso he optado por este esquema debido a que tuve problemas con los hilos de Android y esta fue la solución que me más versatilidad me proporcionó.
- 6) Procesamos la respuesta, actuamos de forma apropiada ya sea navegando a una vista diferente o informando al usuario de el error producido si este fuese el caso. Al navegar entre vistas es común el traspaso de datos de una vista a otra. Esto se realiza mediante el controlador de navegación y Android nos proporciona una forma sencilla de ejecutarlo.

Esté será el caso estándar para la mayoría de los fragmentos de la aplicación exceptuando los fragmentos donde se mostrará listas de elementos los cuales

siguen un esquema parecido pero con una pequeña diferencia. Al cargar la vista bloqueo la interacción con el usuario y realizo una petición para obtener los elementos que pueblan la lista, al obtener los elementos libero la interacción una vez más.

No considero necesario explicar que función cumple cada fragmento pues creo que sus nombres son suficientemente descriptivos. Denotar que los fragmentos de *MostrarAlumno*, *MostrarExpediente*, etc. Además de realizar la tarea de mostrar los datos de un alumno o listar sus expedientes también engloban entre sus funcionalidades las tareas de borrado de alumno o expediente. Este patrón se extiende para el resto de fragmentos de la aplicación que se centren en mostrar información no editable.

Las llamadas al servidor se realizan mediante la clase *ApiLink* cual cuenta con todos los parámetros necesarios para realizar estas llamadas y la respuestas se transmiten a el resto de la aplicación mediante interfaces personalizadas para cada paquete. La clase *ErrorDialogo* interceptará los errores que nos envíe el servidor y se usará mediante la interfaz para informar al usuario (Hecho así una vez más por problemas con la concurrencia en Android)

A continuación en la Figura 13 se muestra el diagrama de clases de la aplicación una vez terminada la iteración. En este punto todo lo referente a expedientes y exámenes está declarado pero no implementado.

Todos los diagramas y grafos referentes al cliente móvil, tanto de esta iteración como de las siguientes, se encuentran en el anexo primero para una más clara visualización.



66



Figura 14: Grafo Navegación – Iteración 1, Cliente Móvil

3.1.5 Resumen de progreso

El progreso en esta iteración ha resultado moderadamente lento, se ha producido un retraso en la realización y diseño de la interfaz a la hora de modificar alumno en el cliente de escritorio. La tarea será aplazada para la siguiente iteración en el que se añadirán los expedientes y los exámenes.

En la Figura 15 se muestra un resumen del progreso durante la iteración. Y en la Figura 16 y Figura 17 se ilustra el *Burdown Chart* de la iteración tanto por tareas realizadas como por puntos de historia completados.

ID	Tarea	Proyecto	Tareas Inicio	3/06/2019 Dia 1	4/06/2019 Dia 2	5/06/2019 Dia 3	6/06/2019 Dia 4	7/06/2019 Dia 5	10/06/2019 Dia 6	11/06/2019 Dia 7	12/06/2019 Dia 8	13/06/2019 Dia 9	14/06/2019 Dia 10	Progreso
Inicio de proyecto														
T0001	Análisis general del sistema		1	1										100.0%
T0002	Diseño Arquitectónico del sistema		1	1										100.0%
T0003	Instalación y configuración de la máquina virtual Java		1	1										100.0%
T0004	Instalación y configuración del SGBD PostgreSQL		1	1										100.0%
T0005	Instalación y configuración de Apache Netbeans		1	1										100.0%
T0006	Instalación y Configuración de Android Studio		1	1										100.0%
T0007	Instalación de herramientas CASE y software de oficina		1	1										100.0%
H001 Añadir Alumno														
T0101	Diseño añadir alumno	Servidor	1				1							100.0%
T0102	Diseño añadir alumno	Cliente de escritorio	1				1							100.0%
T0103	Diseño añadir alumno	Cliente Móvil	1				1							100.0%
T0104	Implementación añadir alumno	Servidor	1					1						100.0%
T0105	Implementación añadir alumno	Cliente Móvil	1					1						100.0%
T0106	Implementación añadir alumno	Cliente de escritorio	1					1	1					100.0%
H002 Modificar Alumno														
T0107	Diseño modificar alumno	Servidor	1				1							100.0%
T0108	Diseño modificar alumno	Cliente de escritorio	1				1							100.0%
T0109	Diseño modificar alumno	Cliente Móvil	1					1						100.0%
T0110	Implementación modificar alumno	Servidor	1						1					100.0%
T0111	Implementación modificar alumno	Cliente Móvil	1						1					100.0%
T0112	Implementación modificar alumno	Cliente de escritorio	1							1				100.0%
H003 Borrar Alumno														
T0113	Diseño borrar alumno	Servidor	1							1				100.0%
T0114	Diseño borrar alumno	Cliente de escritorio	1							1				100.0%
T0115	Diseño borrar alumno	Cliente Móvil	1							1				100.0%
T0116	Implementación borrar alumno	Cliente de escritorio	1								1			100.0%
T0117	Implementación borrar alumno	Cliente Móvil	1									1		100.0%
T0118	Implementación borrar alumno	Servidor	1								1			100.0%
H004 Búsqueda por DNI														
T0119	Diseño búsqueda por DNI	Servidor	1									1		100.0%
T0120	Diseño búsqueda por DNI	Cliente de escritorio	1									1		100.0%
T0121	Diseño búsqueda por DNI	Cliente Móvil	1										1	100.0%
T0122	Implementación búsqueda de alumno por DNI	Servidor	1										1	100.0%
T0123	Implementación búsqueda de alumno por DNI	Cliente de escritorio	1										1	100.0%
T0124	Implementación búsqueda de alumno por DNI	Cliente Móvil	1										1	100.0%
H005 Añadir Expediente														
T0125	Diseño añadir expediente	Servidor	1											0.0%
T0126	Diseño añadir expediente	Cliente de escritorio	1											0.0%
T0127	Diseño añadir expediente	Cliente Móvil	1											0.0%
T0128	Implementación añadir expediente	Servidor	1											0.0%
T0129	Implementación añadir expediente	Cliente de escritorio	1											0.0%
T0130	Implementación añadir expediente	Cliente Móvil	1											0.0%
H006 Borrar Expediente														
T0131	Diseño borrar expediente	Servidor	1											0.0%
T0132	Diseño borrar expediente	Cliente de escritorio	1											0.0%
T0133	Diseño borrar expediente	Cliente Móvil	1											0.0%
T0134	Implementación borrar expediente	Servidor	1											0.0%
T0135	Implementación borrar expediente	Cliente de escritorio	1											0.0%
T0136	Implementación borrar expediente	Cliente Móvil	1											0.0%
H007 Modificar Expediente														
T0137	Diseño modificar expediente	Servidor	1											0.0%
T0138	Diseño modificar expediente	Cliente de escritorio	1											0.0%
T0139	Diseño modificar expediente	Cliente Móvil	1											0.0%
T0140	Implementación modificar expediente	Servidor	1											0.0%
T0141	Implementación modificar expediente	Cliente de escritorio	1											0.0%
T0142	Implementación modificar expediente	Cliente Móvil	1											0.0%
T0143	Pruebas de unidad	Servidor	1											0.0%
				50	48	45	43	38	35	32	29	26	23	20
				50	45	40	35	30	25	20	15	10	5	0
				15	15	15	15	15	15	9	8	8	8	7
				15	13.5	12	10.5	9	7.5	6	4.5	3	1.5	0

Figura 15: Resumen de Progreso – Iteración 1

Burndown Chart

Iteración 1

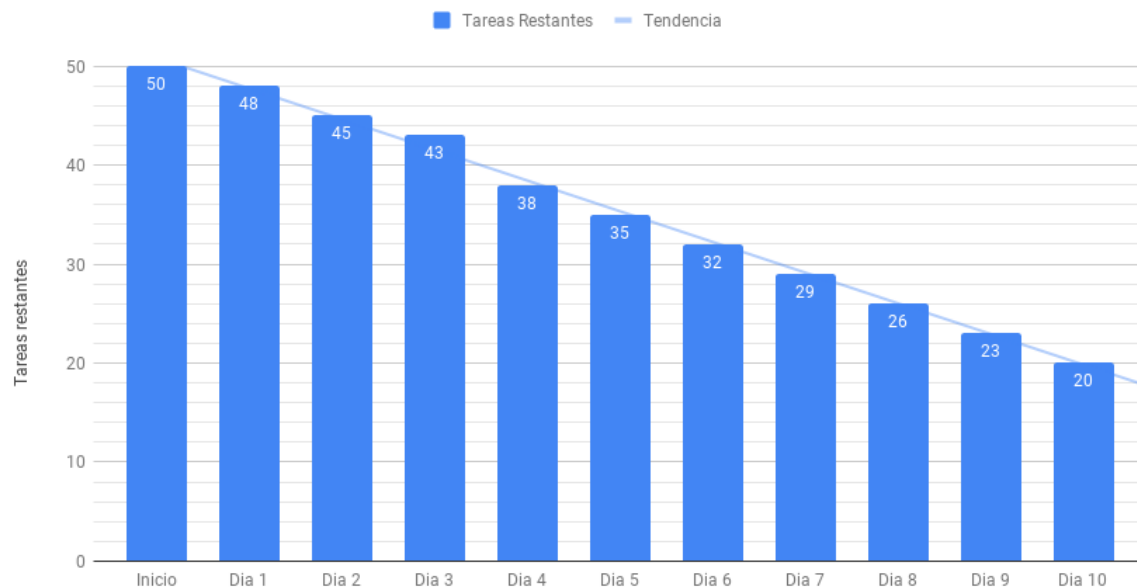


Figura 16: Burdown Chart – Tareas Iteración 1

Burndown Chart

Iteración 1

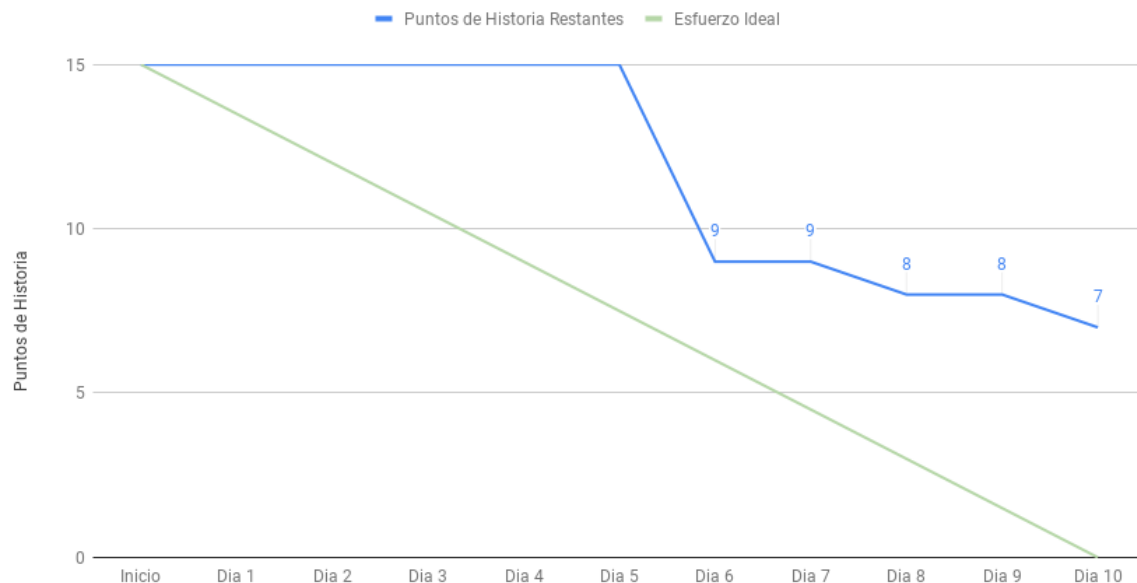


Figura 17: Burndown Chart – Iteración 1

3.1.6 Entrega y reunión final de la iteración

El primer prototipo se enseñó en funcionamiento al cliente en la oficina de la empresa mostrando la versión de desarrollo implementada durante la iteración. La valoración del cliente fue satisfactoria, a pesar de la pobre apariencia visual del mismo.

Como mejoras para el cliente de sobremesa, propone forzar la entrada de datos a mayúsculas para mejorar la visibilidad de los campos en la ficha de alumno y posteriores, así como aumentar el tamaño general de fuente en la interfaz, se decide aumentar el tamaño de fuente base de la aplicación.

En lo referente al cliente móvil se aconsejan ciertos cambios en la interfaz para mejorar su apariencia. Estos cambios están centrados a la forma en la que se muestran los campos referentes al alumno. Que a juicio del cliente son confusos y mejorables. Todas estos comentarios se tendrán en cuenta para su mejora en las siguientes iteraciones.

En la segunda iteración se pretende aplicar esas recomendaciones al cliente de escritorio y, ir empezando a aplicar mejoras visuales a la interfaz mediante hojas

de estilo e iconos e implementar las funcionalidades relacionadas con la gestión de expedientes y exámenes.

3.2 Segunda Iteración

Después de la primera iteración, proseguimos con la implementación del diseño elaborado anteriormente, incluyendo las clases expediente y examen y desarrollando las historias de usuario relativas a estas clases del dominio.

3.2.1 Historias de usuario

En las tablas Tabla 18, Tabla 19, Tabla 20, Tabla 21, Tabla 22, Tabla 23 y Tabla 24 se muestran las historias de usuario de esta iteración.

Tabla 18: H005 Añadir Expediente

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H005	Estimación 3	Prioridad Must Have	
Como Administrador	Quiero Añadir Expedientes	Para Gestionar el proceso educativo de mis estudiantes	
Criterios de satisfacción - Dado que se han introducido correctamente los datos del expediente, se cree una entrada en el registro			
Redactado por Alberto Carrillo Ortega	Fecha 18/06/2019	Aprobado por Javier Martínez Cavanillas	Fecha 18/06/2019

Tabla 19: H006 Borrar Expediente

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H006	Estimación 0.5	Prioridad Could Have	
Como Administrador	Quiero Borrar Expedientes	Para Poder corregir errores y organizar la información de los expedientes	
Criterios de satisfacción - Dado que se ha borrado el expediente correctamente, se borre su entrada del registro y los exámenes relacionados el			
Redactado por Alberto Carrillo Ortega	Fecha 18/06/2019	Aprobado por Javier Martínez Cavanillas	Fecha 18/06/2019

Tabla 20: H007 Modificar Expediente

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H007	Estimación 3	Prioridad Could Have	
Como Administrador	Quiero Modificar Expedientes	Para Poder corregir errores y actualizar la información de los expedientes	
Criterios de satisfacción - Dado que se han modificado correctamente los datos del expediente, se modifique su entrada en el registro			
Redactado por Alberto Carrillo Ortega	Fecha 18/06/2019	Aprobado por Javier Martínez Cavanillas	Fecha 19/06/2019

Tabla 21: H008 Añadir examen

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H008	Estimación 2	Prioridad Must Have	
Como Administrador	Quiero Añadir Exámenes	Para Gestionar las evaluaciones de mis estudiantes	
Criterios de satisfacción - Dado que se han introducido correctamente los datos del examen, se cree una entrada en el registro			
Redactado por Alberto Carrillo Ortega	Fecha 19/06/2019	Aprobado por Javier Martínez Cavanillas	Fecha 19/06/2019

Tabla 22: H009 Borrar examen

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H009	Estimación 0.5	Prioridad Could Have	
Como Administrador	Quiero Borrar exámenes	Para Poder corregir errores y organizar la información de los expedientes	
Criterios de satisfacción - Dado que se ha borrado el examen correctamente, se borre su entrada del registro			
Redactado por Alberto Carrillo Ortega	Fecha 18/06/2019	Aprobado por Javier Martínez Cavanillas	Fecha 18/06/2019

Tabla 23: H010 Listar Expedientes

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H010	Estimación 0.5	Prioridad Must Have	
Como Administrador	Quiero Listar Expedientes	Para Poder ver la información referente a los expedientes de un alumno	
Criterios de satisfacción - Dado que se ha seleccionado un alumno valido, se muestran las entrada correspondiente a los expedientes de ese alumno			
Redactado por	Fecha	Aprobado por	Fecha

Alberto Carrillo Ortega	18/06/2019	Javier Martínez Cavanillas	18/06/2019
-------------------------	------------	----------------------------	------------

Tabla 24: H011 Listar Exámenes

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H011	Estimación 0.5	Prioridad Must Have	
Como Administrador	Quiero Listar Exámenes	Para Poder visualizar la información referente a los exámenes de un expediente	
Criterios de satisfacción - Dado que se ha seleccionado un expediente valido, se muestran las entrada correspondiente a los exámenes de ese expediente			
Redactado por Alberto Carrillo Ortega	Fecha 18/06/2019	Aprobado por Javier Martínez Cavanillas	Fecha 18/06/2019

3.2.2 Análisis

En las figuras Figura 18, Figura 19, Figura 20, Figura 21, Figura 22, Figura 23 y Figura 24 se muestran los diagramas de actividad referentes a las historias de usuario del apartado anterior.

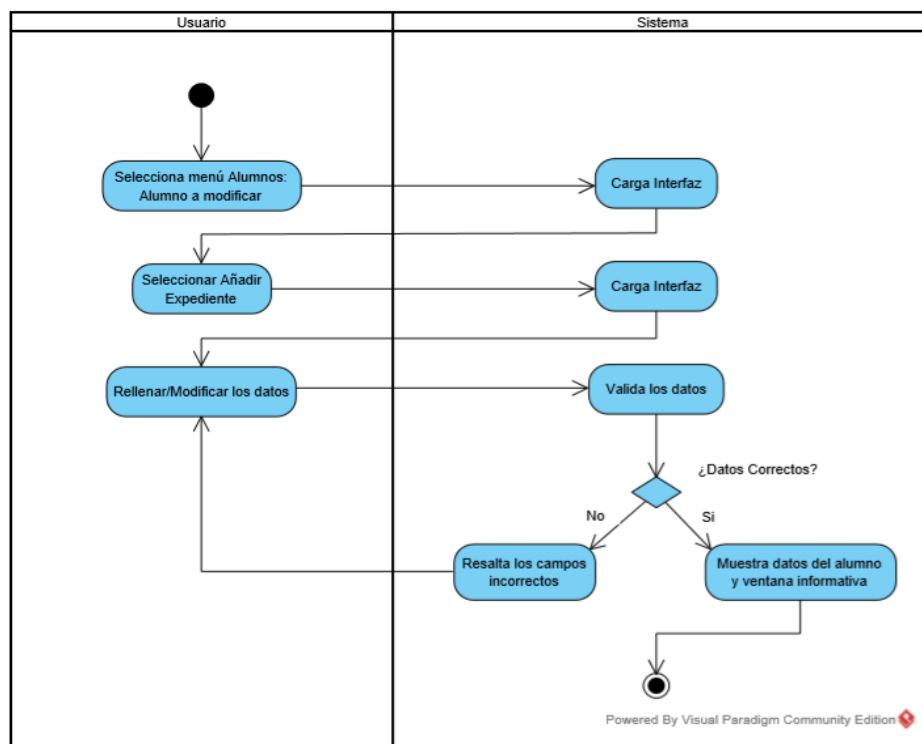


Figura 18: DA005 Añadir Expediente (H005)

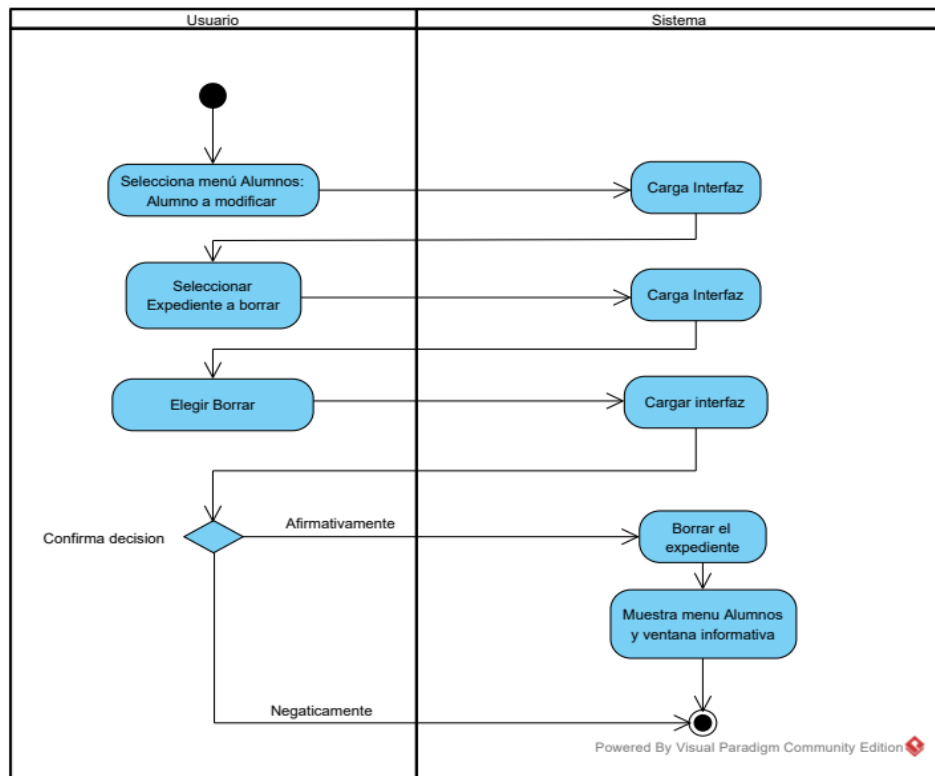


Figura 19: DA006 Borrar Expediente (H006)

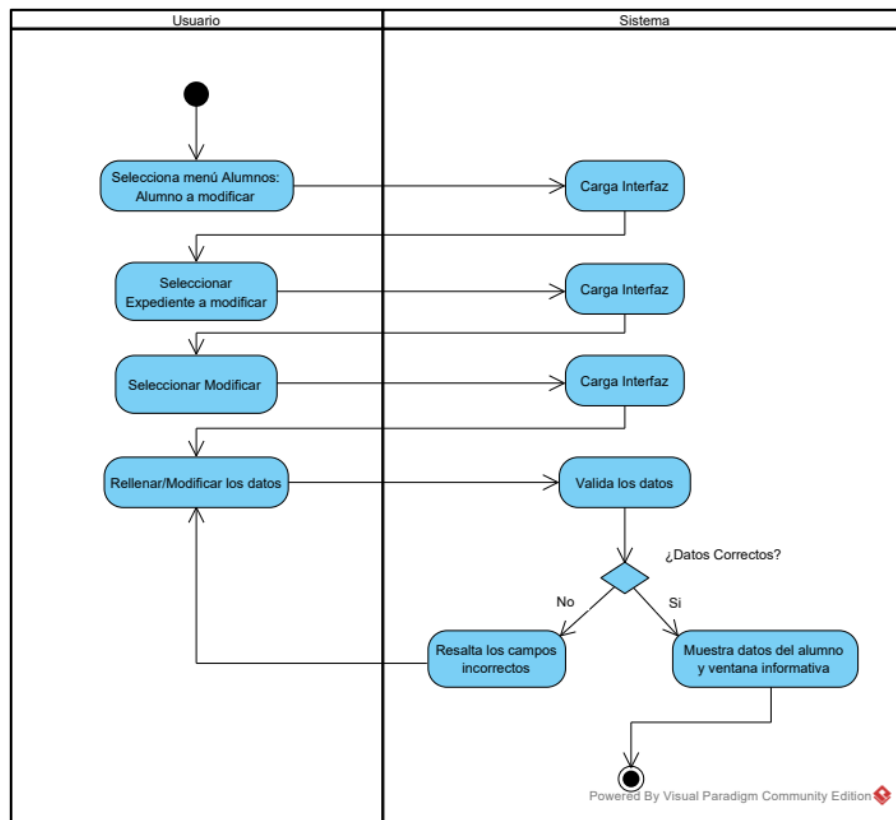


Figura 20: DA007 Modificar Expediente (H007)

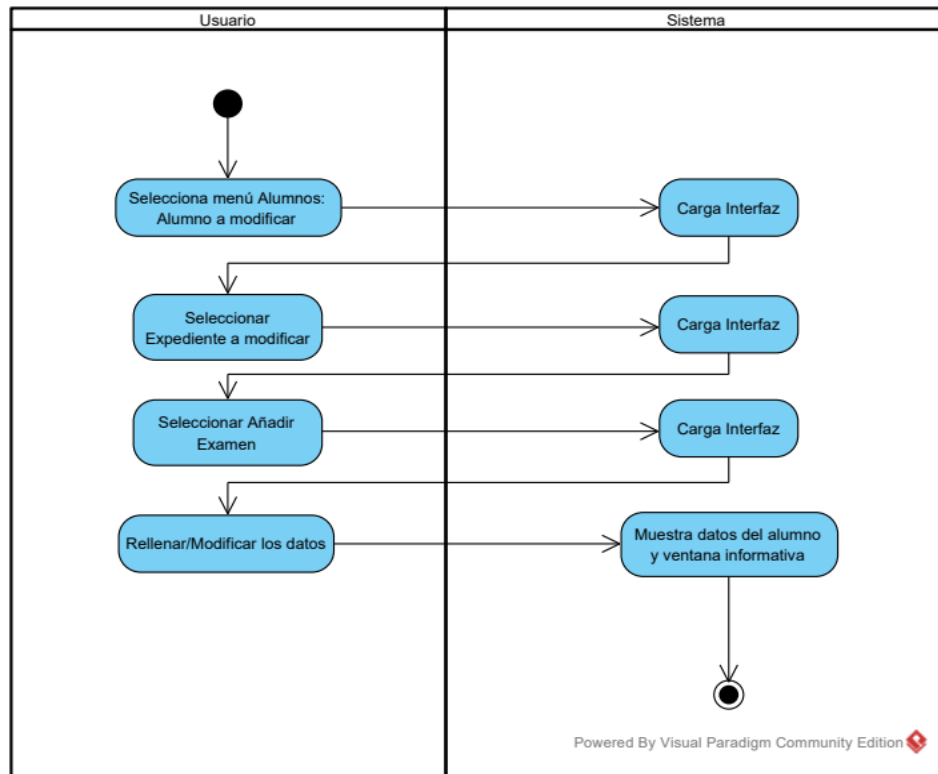


Figura 21: DA008 Añadir Examen (H008)

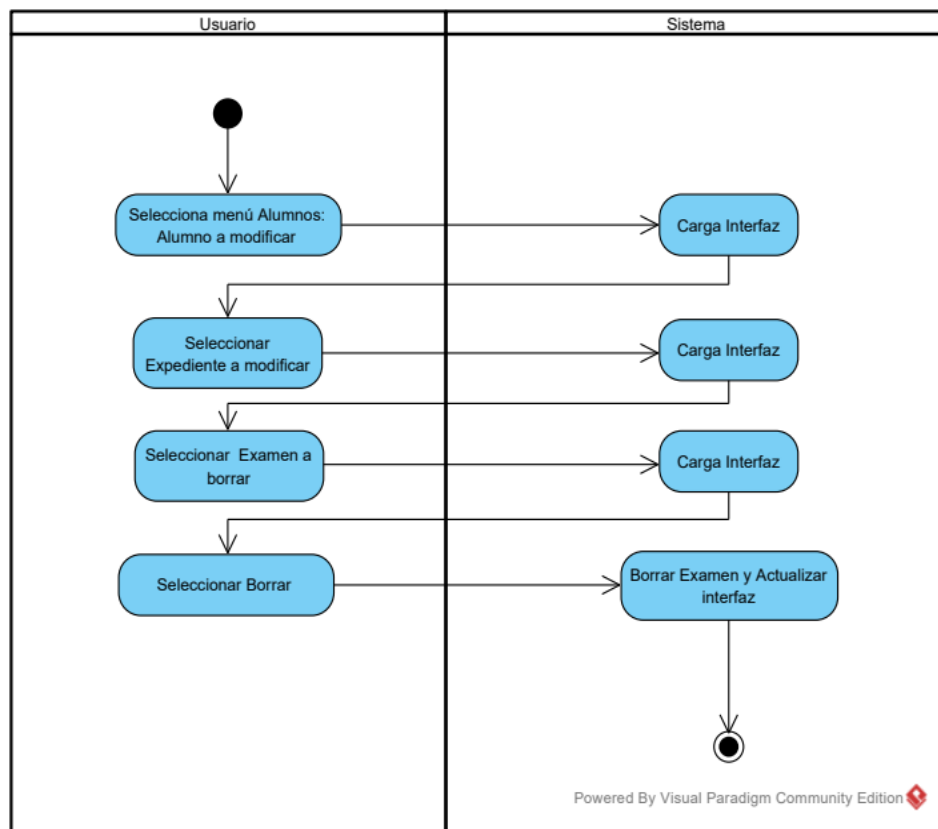


Figura 22: DA009 Borrar Examen (H009)

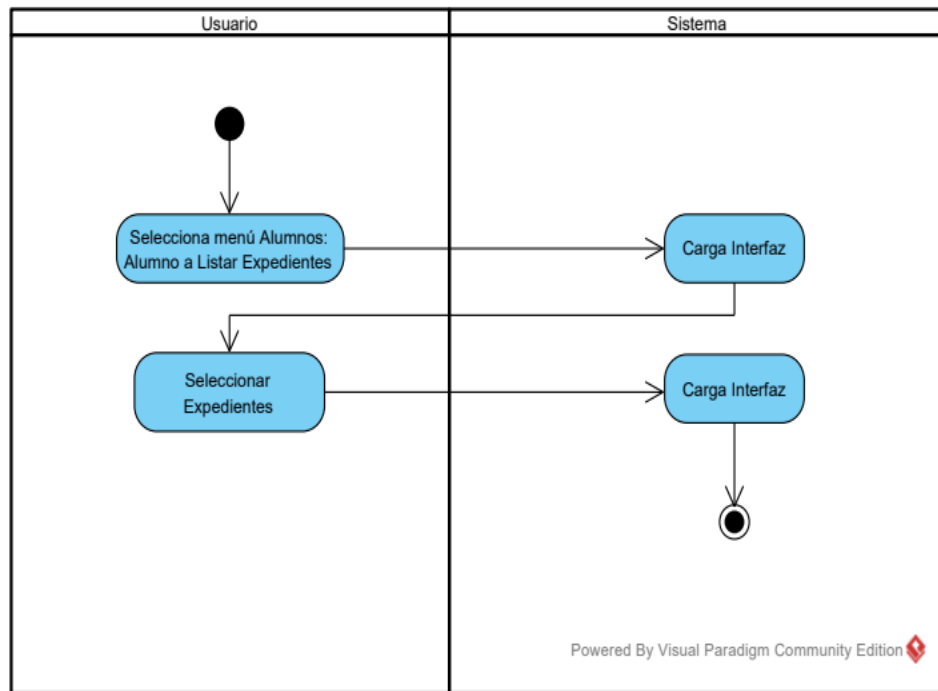


Figura 23: DA010 Listar Expedientes (H010)

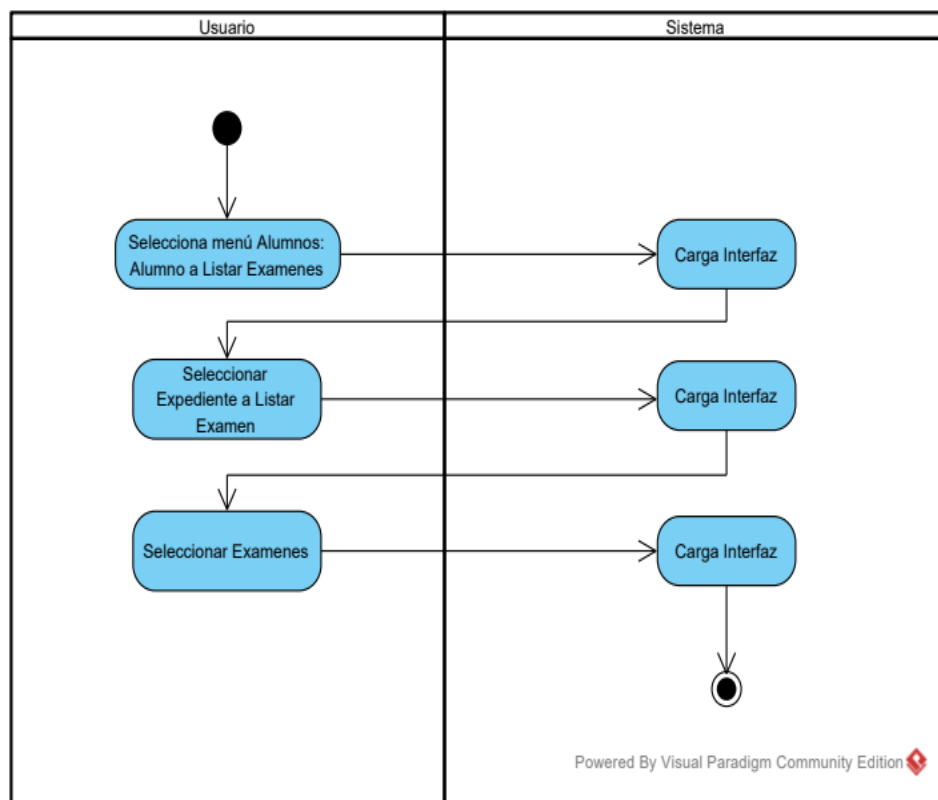


Figura 24: DA011 Listar Exámenes (H011)

3.2.3 Planificación de la iteración

En esta iteración la prioridad será realizar las tareas que no pudieron completarse en la anterior iteración. Es decir, las tareas de modificar expediente, añadir expediente, etc. Afortunadamente se prevé un incremento en el ritmo de trabajo del equipo, pues en la primera iteración se dedicó un tiempo considerable al desarrollo de un diseño extensible a todo el modulo de alumnado.

Una vez completadas las tareas atrasadas el equipo continuará el desarrollo de forma normal.

En la Tabla 25 se muestra un esquema del *Backlog de Iteración* de la segunda iteración.

Tabla 25: Backlog de Iteración – Iteración 2

BACKLOG DE ITERACIÓN		Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas
Iteración Iteración 2 – Alumnos Parte II			Puntos de Historia 15
ID	Tarea	Proyecto	
H002 Modificar Alumno			
T0112	Implementación modificar alumno	Cliente de escritorio	
H005 Añadir Expediente			
T0125	Diseño añadir expediente	Servidor	
T0126	Diseño añadir expediente	Cliente de escritorio	
T0127	Diseño añadir expediente	Cliente Móvil	
T0128	Implementación añadir expediente	Servidor	
T0129	Implementación añadir expediente	Cliente de escritorio	
T0130	Implementación añadir expediente	Cliente Móvil	
H006 Borrar Expediente			
T0131	Diseño borrar expediente	Servidor	
T0132	Diseño borrar expediente	Cliente de escritorio	
T0133	Diseño borrar expediente	Cliente Móvil	

T0134	Implementación borrar expediente	Servidor
T0135	Implementación borrar expediente	Cliente de escritorio
T0136	Implementación borrar expediente	Cliente Móvil
H007 Modificar Expediente		
T0137	Diseño modificar expediente	Servidor
T0138	Diseño modificar expediente	Cliente de escritorio
T0139	Diseño modificar expediente	Cliente Móvil
T0140	Implementación modificar expediente	Servidor
T0141	Implementación modificar expediente	Cliente de escritorio
T0142	Implementación modificar expediente	Cliente Móvil
T0143	Pruebas de unidad	Servidor
H008 Añadir examen		
T0201	Diseño añadir examen	Servidor
T0202	Diseño añadir examen	Cliente de escritorio
T0203	Diseño añadir examen	Cliente Móvil
T0204	Implementación añadir examen	Servidor
T0205	Implementación añadir examen	Cliente de escritorio
T0206	Implementación añadir examen	Cliente Móvil
H009 Borrar examen		
T0207	Diseño borrar examen	Servidor
T0208	Diseño borrar examen	Cliente de escritorio
T0209	Diseño borrar examen	Cliente Móvil
T0210	Implementación borrar examen	Servidor
T0211	Implementación borrar examen	Cliente de escritorio
T0212	Implementación borrar examen	Cliente Móvil
H011 Listar Exámenes		

T0213	Diseño Listar exámenes	Servidor
T0214	Diseño Listar exámenes	Cliente de escritorio
T0215	Diseño Listar exámenes	Cliente Móvil
T0216	Implementación Listar exámenes	Servidor
T0217	Implementación Listar exámenes	Cliente de escritorio
T0218	Implementación Listar exámenes	Cliente Móvil
H010 Listar Expedientes		
T0219	Diseño Listar expedientes	Servidor
T0220	Diseño Listar expedientes	Cliente Móvil
T0221	Diseño Listar expedientes	Cliente de escritorio
T0222	Implementación Listar expedientes	Servidor
T0223	Implementación Listar expedientes	Cliente Móvil
T0224	Implementación Listar expedientes	Cliente de escritorio
Otros		
T0225	Elaborar hoja de estilos CSS	Cliente de escritorio
T0226	Aumentar tamaño de letra	Cliente de escritorio
T0227	Forzar cambios de texto a mayúscula	Cliente de escritorio
T0228	Pruebas de Unidad	Servidor
T0229	Validación de campos	Cliente de escritorio

3.2.4 Diseño

El diseño de esta iteración se explicó en la anterior iteración para mejor entendimiento en el punto

3.2.4.1 *Cliente Móvil*

En esta iteración no hubo nada que diseñar pero aprovecharé para profundizar aún mas en los fragmentos que muestran información. Los fragmentos *MostrarExpedientes* y *MostrarExámenes* están estrechamente ligados a las clases *PostAdapterExpedientes* y *PostAdapterExámenes* respectivamente. Estos fragmentos poseen un elemento Android llamado RecyclerView. Este elemento es el que se encarga de mostrar todos los expedientes y exámenes en forma de lista..

Para ello es necesaria la implementación de las clases *PostAdapterExpedientes/Exámenes*. Las clases *PostAdapterExpedientes/Exámenes* son las encargadas de vincular los expedientes y exámenes a las listas. Además para la posterior interacción los *PostAdapters* definen cada uno su interfaz *ItemListenerExpedientes/Exámenes* mediante la cual los fragmentos *MostrarExpedientes/Exámenes*, los cuales implementan la dicha interfaz, reciben el elemento con el que el usuario ha interactuado y en función de la interacción se invoca un método u otro. Esto da mucha libertad y flexibilidad a la hora de programar.

Por ultimo decir que cada elemento de la lista es una vista en si misma. Esta vista está almacenada en una CardView (Otro elemento XML de Android para este tipo de situaciones que consiste en una vista normalmente pequeña y compacta) con el formato card_elemento. Por ejemplo los exámenes vinculan sus datos con los campos de card_examen y los expedientes con card_expediente. Durante la vinculación de los datos añadimos la funcionalidad a los botones de CardView para interactuar mediante la interfaz listener previamente mencionada.

A continuación en la Figura 25 adjunto el estado del grafo de navegación al finalizar la iteración.

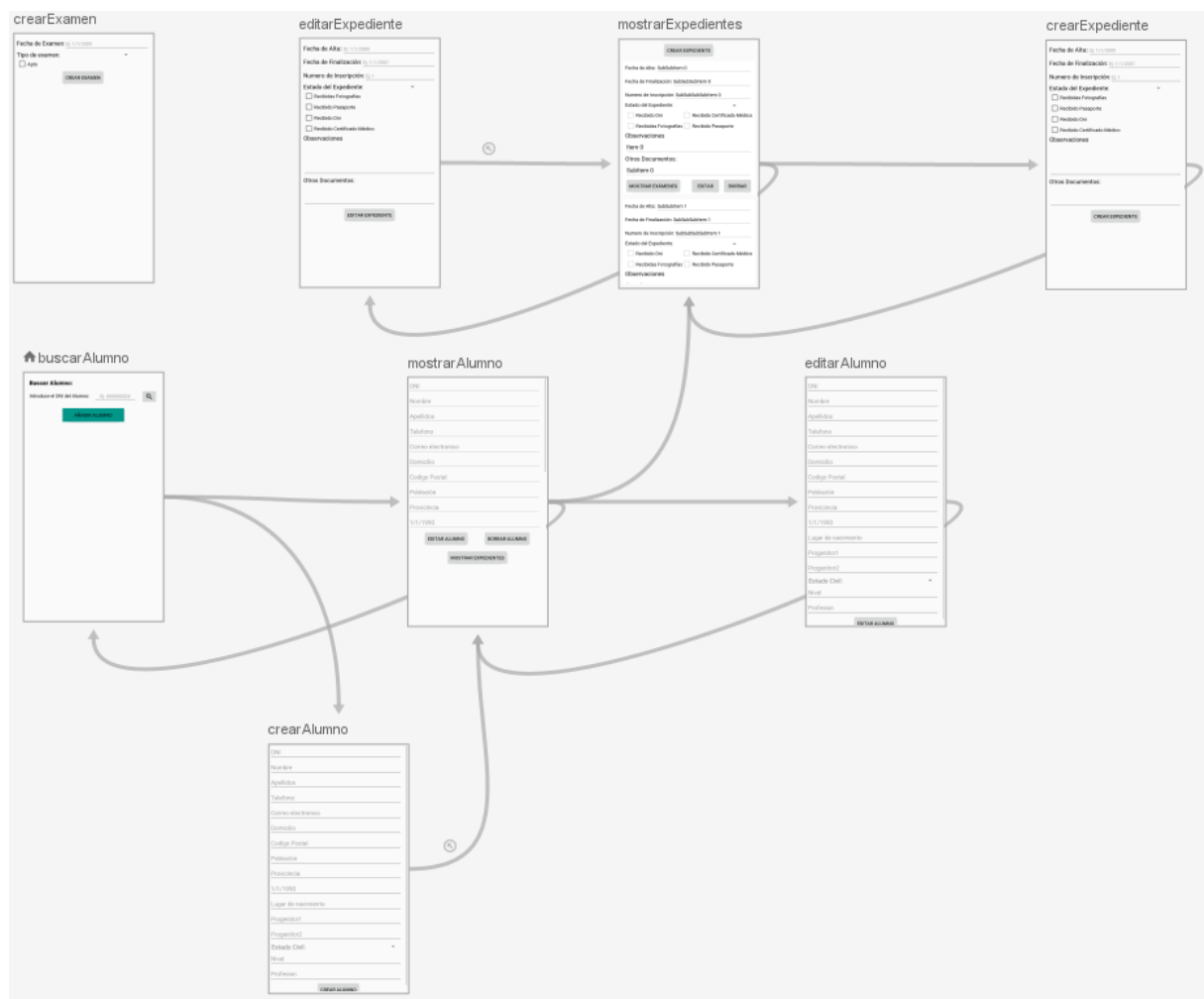


Figura 25: Cliente Móvil: Grafo Navegación 2ª Iteración

3.2.5 Resumen de progreso

En esta iteración conseguimos recuperar la totalidad de tareas atrasadas y nos situamos solamente con dos tareas sin poder completar a tiempo: Listar exámenes en la implementación de los clientes, tanto en el de escritorio como en el móvil. Aún no se han podido seguir las recomendaciones de mejoras visuales en el cliente móvil, pero se han tenido en cuenta para el desarrollo del resto de tareas.

En la Tabla 25 se muestra un resumen del progreso durante la iteración. Y en la Figura 27 se ilustra el *Burdown Chart* de la iteración.

ID	Tarea	Proyecto	Tareas Inicio	17/06/2019 Dia 1	18/06/2019 Dia 2	19/06/2019 Dia 3	20/06/2019 Dia 4	21/06/2019 Dia 5	24/06/2019 Dia 6	25/06/2019 Dia 7	26/06/2019 Dia 8	27/06/2019 Dia 9	28/06/2019 Dia 10	Progreso
H002 Modificar Alumno														
T0112	Implementación modificar alumno	Cliente de escritorio	1	1										100.0%
H005 Añadir Expediente														
T0125	Diseño añadir expediente	Servidor	1	1										100.0%
T0126	Diseño añadir expediente	Cliente de escritorio	1	1										100.0%
T0127	Diseño añadir expediente	Cliente Móvil	1	1										100.0%
T0128	Implementación añadir expediente	Servidor	1		1									100.0%
T0129	Implementación añadir expediente	Cliente de escritorio	1		1									100.0%
T0130	Implementación añadir expediente	Cliente Móvil	1		1									100.0%
H006 Borrar Expediente														
T0131	Diseño borrar expediente	Servidor	1		1									100.0%
T0132	Diseño borrar expediente	Cliente de escritorio	1		1									100.0%
T0133	Diseño borrar expediente	Cliente Móvil	1		1									100.0%
T0134	Implementación borrar expediente	Servidor	1		1									100.0%
T0135	Implementación borrar expediente	Cliente de escritorio	1		1									100.0%
T0136	Implementación borrar expediente	Cliente Móvil	1		1									100.0%
H007 Modificar Expediente														
T0137	Diseño modificar expediente	Servidor	1		1									100.0%
T0138	Diseño modificar expediente	Cliente de escritorio	1		1									100.0%
T0139	Diseño modificar expediente	Cliente Móvil	1		1									100.0%
T0140	Implementación modificar expediente	Servidor	1		1									100.0%
T0141	Implementación modificar expediente	Cliente de escritorio	1		1									100.0%
T0142	Implementación modificar expediente	Cliente Móvil	1		1									100.0%
T0143	Pruebas de unidad	Servidor	1		1									100.0%
H008 Añadir examen														
T0201	Diseño añadir examen	Servidor	1						1					100.0%
T0202	Diseño añadir examen	Cliente de escritorio	1						1					100.0%
T0203	Diseño añadir examen	Cliente Móvil	1						1					100.0%
T0204	Implementación añadir examen	Servidor	1						1					100.0%
T0205	Implementación añadir examen	Cliente de escritorio	1						1					100.0%
T0206	Implementación añadir examen	Cliente Móvil	1						1					100.0%
H009 Borrar examen														
T0207	Diseño borrar examen	Servidor	1						1					100.0%
T0208	Diseño borrar examen	Cliente de escritorio	1						1					100.0%
T0209	Diseño borrar examen	Cliente Móvil	1						1					100.0%
T0210	Implementación borrar examen	Servidor	1						1					100.0%
T0211	Implementación borrar examen	Cliente de escritorio	1						1					100.0%
T0212	Implementación borrar examen	Cliente Móvil	1						1					100.0%
H011 Listar Exámenes														
T0213	Diseño Listar exámenes	Servidor	1							1				100.0%
T0214	Diseño Listar exámenes	Cliente de escritorio	1							1				100.0%
T0215	Diseño Listar exámenes	Cliente Móvil	1							1				100.0%
T0216	Implementación Listar exámenes	Servidor	1								1			100.0%
T0217	Implementación Listar exámenes	Cliente de escritorio	1									1		0.0%
T0218	Implementación Listar exámenes	Cliente Móvil	1										1	0.0%
H010 Listar Expedientes														
T0219	Diseño Listar expedientes	Servidor	1							1				100.0%
T0220	Diseño Listar expedientes	Cliente Móvil	1							1				100.0%
T0221	Diseño Listar expedientes	Cliente de escritorio	1								1			100.0%
T0222	Implementación Listar expedientes	Servidor	1									1		100.0%
T0223	Implementación Listar expedientes	Cliente Móvil	1										1	100.0%
T0224	Implementación Listar expedientes	Cliente de escritorio	1										1	100.0%
Otros														
T0225	Elaborar hoja de estilos CSS	Cliente de escritorio	1									1		100.0%
T0226	Aumentar tamaño de letra	Cliente de escritorio	1						1					100.0%
T0227	Forzar cambios de texto a mayúscula	Cliente de escritorio	1			1								100.0%
T0228	Pruebas de Unidad	Servidor	1											100.0%
T0229	Validación de campos	Cliente de escritorio	1		0.3					0.3	0.4	0.5	0.5	100.0%
Tareas Restantes				50	46	38,7	34,7	31,7	24,7	15,7	10,4	6	3,5	3
Esfuerzo Ideal				50	45	40	35	30	25	20	15	10	5	0
Puntos de Historia Restantes				15	14	11	10	10	7	4	4	2	2	0
Esfuerzo Ideal				15	13,5	12	10,5	9	7,5	6	4,5	3	1,5	0

Figura 26: Resumen de Progreso – Iteración 2

Burndown Chart

Iteración 2



Figura 27: Burndown Chart – Tareas Iteración 2

Burndown Chart

Iteración 2

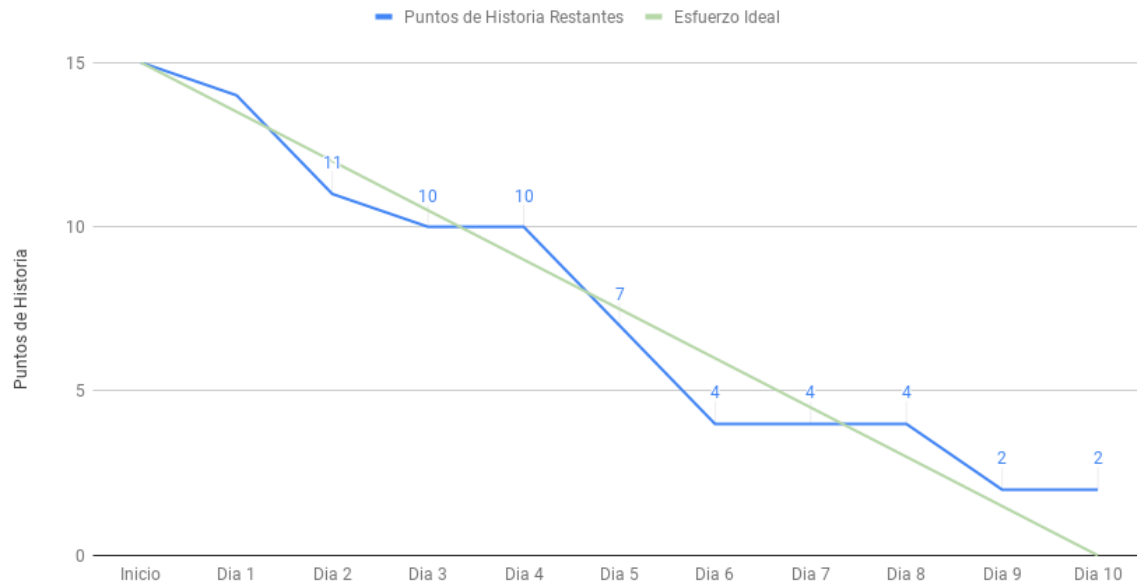


Figura 28: Burndown Chart – Iteración 2

3.2.6 Entrega y reunión final de la iteración

Se muestra la ejecución del módulo de alumnos completo al cliente. El usuario prueba las distintas funcionalidades disponibles e indica un par de correcciones, estas están relacionadas con la nomenclatura utilizada por la autoescuela en términos relacionados con el tipo de examen y los permisos correspondientes, por lo que propone modificar un par de términos de la interfaz para las futuras entregas.

En lo referente al cliente móvil el cliente está satisfecho con el formato de las nuevas tareas. Y comprueba el funcionamiento satisfactorio de la aplicación. Sugiere validación de campos para informar al usuario de sus errores al rellenar los campos.

3.3 Tercera Iteración

En la tercera iteración se sigue el ritmo previsto al inicio y se inicia el desarrollo de las tareas relacionadas con los vehículos de la autoescuela. Continuamos con las tareas atrasadas y desarrollando las historias de usuario de esta iteración.

3.3.1 Historias de usuario

En las tablas Tabla 26, Tabla 27, Tabla 28, Tabla 29, Tabla 30, Tabla 31 y Tabla 32 se muestran las historias de usuario de esta iteración.

Tabla 26: H012 Añadir Vehículo

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H012	Estimación 2	Prioridad Must Have	
Como Administrador	Quiero Añadir Vehículos	Para Añadir un vehículo a mis flotas	
Criterios de satisfacción - Dado que se han introducido correctamente los datos del vehículo, se cree una entrada en el registro			
Redactado por Alberto Carrillo Ortega	Fecha 25/06/2019	Aprobado por Javier Martínez Cavanillas	Fecha 25/06/2019

Tabla 27: H013 Modificar Vehículo

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H013	Estimación 2	Prioridad Could Have	
Como Administrador	Quiero Modificar Vehículos	Para Corregir y actualizar los datos de mi flota	
Criterios de satisfacción - Dado que se han modificado correctamente los datos del vehículo, se modifique su entrada en el registro			
Redactado por Alberto Carrillo Ortega	Fecha 25/06/2019	Aprobado por Javier Martínez Cavanillas	Fecha 25/06/2019

Tabla 28: H014 Borrar Vehículo

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H014	Estimación 0.5	Prioridad Could Have	
Como Administrador	Quiero Borrar Vehículos	Para Eliminar de la flota los vehículos deseados	
Criterios de satisfacción - Dado que se ha borrado el vehículo correctamente, se borre su entrada del registro y las operaciones de mantenimiento relacionados con el			
Redactado por Alberto Carrillo Ortega	Fecha 25/06/2019	Aprobado por Javier Martínez Cavanillas	Fecha 25/06/2019

Tabla 29: H015 Añadir Operación de mantenimiento

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H015	Estimación 1	Prioridad Must Have	
Como Administrador	Quiero Añadir Operaciones de mantenimiento	Para Llevar el control de la gestión de un vehículo concreto	
Criterios de satisfacción - Dado que se han introducido correctamente los datos la operación de mantenimiento, se cree una entrada en el registro			
Redactado por Alberto Carrillo Ortega	Fecha 25/06/2019	Aprobado por Javier Martínez Cavanillas	Fecha 25/06/2019

Tabla 30: H016 Borrar Operaciones de mantenimiento

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H016	Estimación 0.5	Prioridad Could Have	

Como Administrador	Quiero Borrar Operaciones de mantenimiento	Para Corregir errores durante el control de la gestión de un vehículo concreto	
Criterios de satisfacción - Dado que se ha borrado la operación de mantenimiento correctamente, se borre su entrada del registro			
Redactado por Alberto Carrillo Ortega	Fecha 25/06/2019	Aprobado por Javier Martínez Cavanillas	Fecha 25/06/2019

Tabla 31: H017 Listar Vehículos

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H017	Estimación 3	Prioridad Must Have	
Como Administrador	Quiero Listar Vehículos	Para Ver todos los vehículos de mi flota	
Criterios de satisfacción - Dado que hay vehículos en el registro, se devuelvan las entradas correspondientes a estos			
Redactado por Alberto Carrillo Ortega	Fecha 25/06/2019	Aprobado por Javier Martínez Cavanillas	Fecha 25/06/2019

Tabla 32: H018 Listar Operaciones de mantenimiento

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19		Coordinador de Proyecto Javier Martínez Cavanillas
ID H018	Estimación 1	Prioridad Must Have	
Como Administrador	Quiero Listar Operaciones de mantenimiento	Para Ver el historial de las operaciones realizadas a un vehículo concreto	
Criterios de satisfacción - Dado que se ha seleccionado un vehículo valido, se devuelvan las entrada correspondiente a las operaciones de mantenimiento de ese vehículo			
Redactado por Alberto Carrillo Ortega	Fecha 25/06/2019	Aprobado por Javier Martínez Cavanillas	Fecha 25/06/2019

3.3.2 Análisis

En las figuras Figura 29, Figura 30, Figura 31, Figura 32, Figura 33, Figura 34 y Figura 35 se muestran los diagramas de actividad referentes a las historias de usuario del apartado anterior.

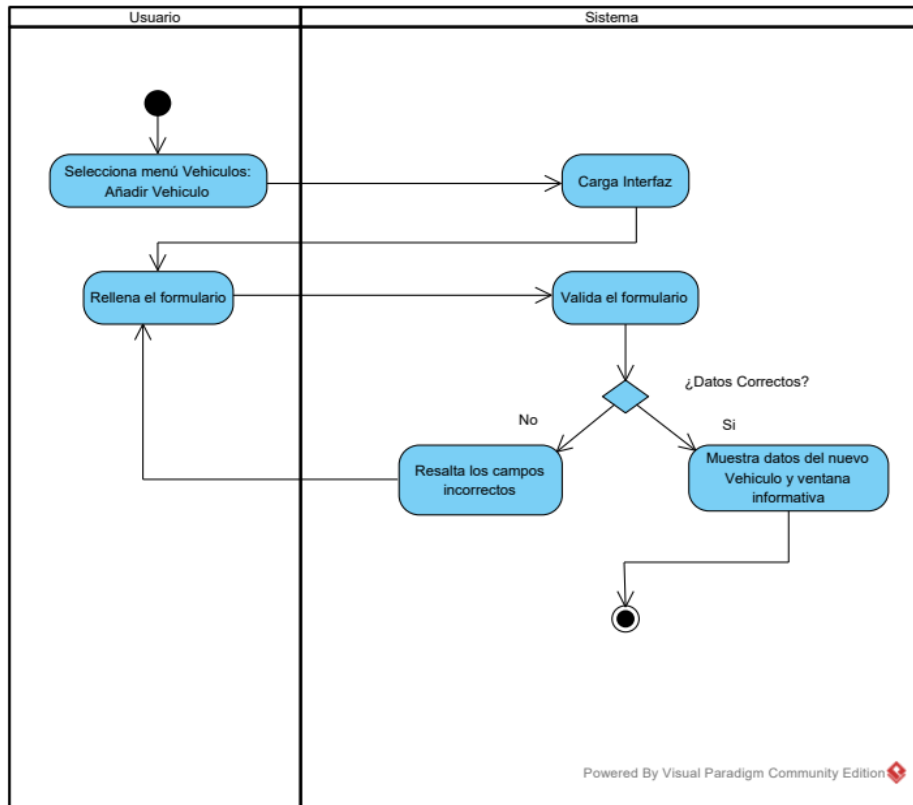


Figura 29: DA012 Añadir Vehículo (H012)

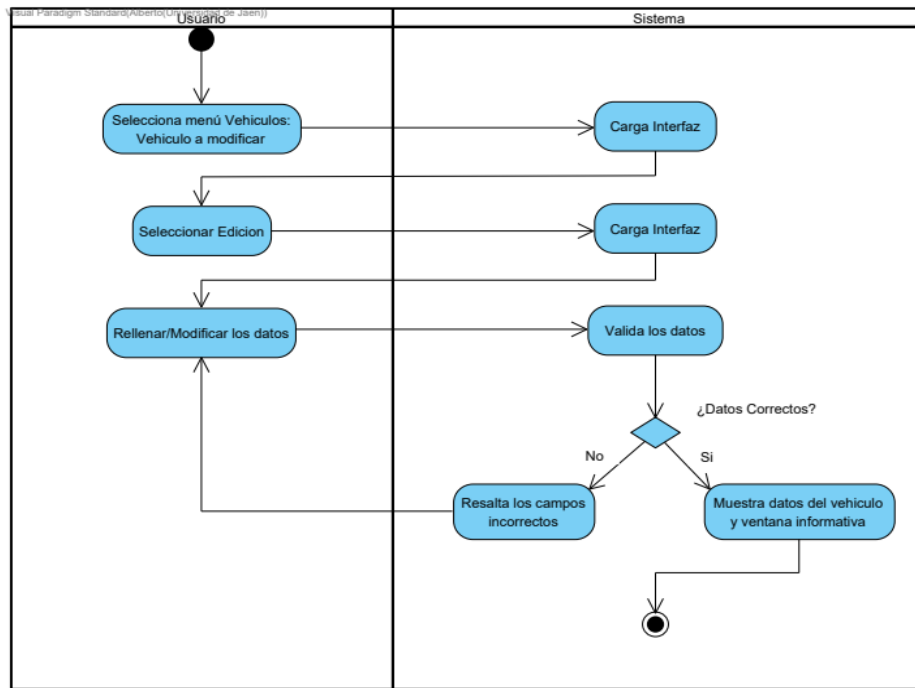


Figura 30: D013 Modificar Vehículo(H013)

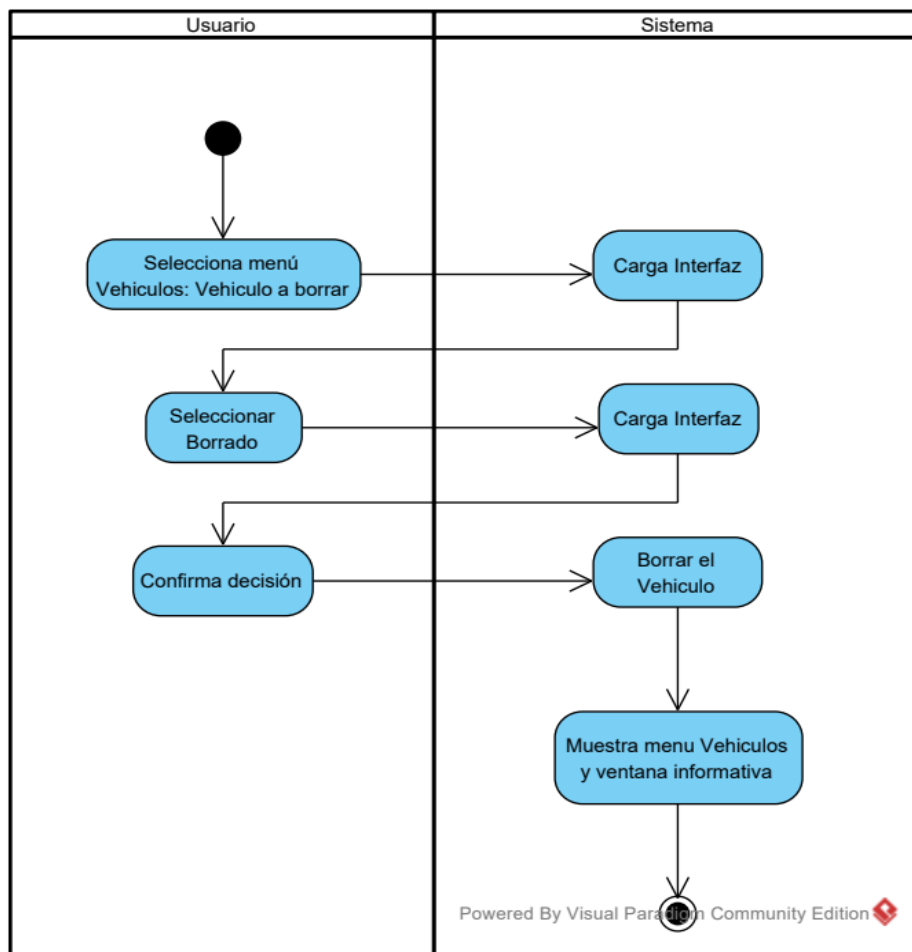


Figura 31: D014 Borrar Vehículo (H014)

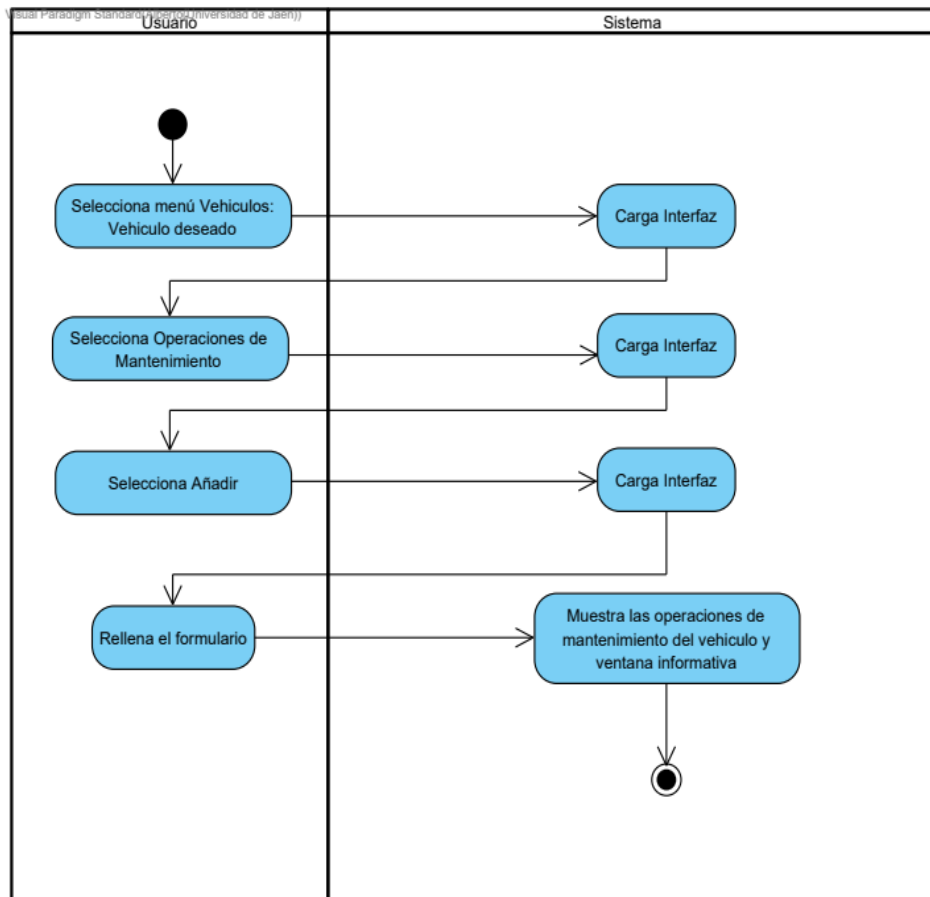


Figura 32: D015 Añadir Operación de Mantenimiento(H015)

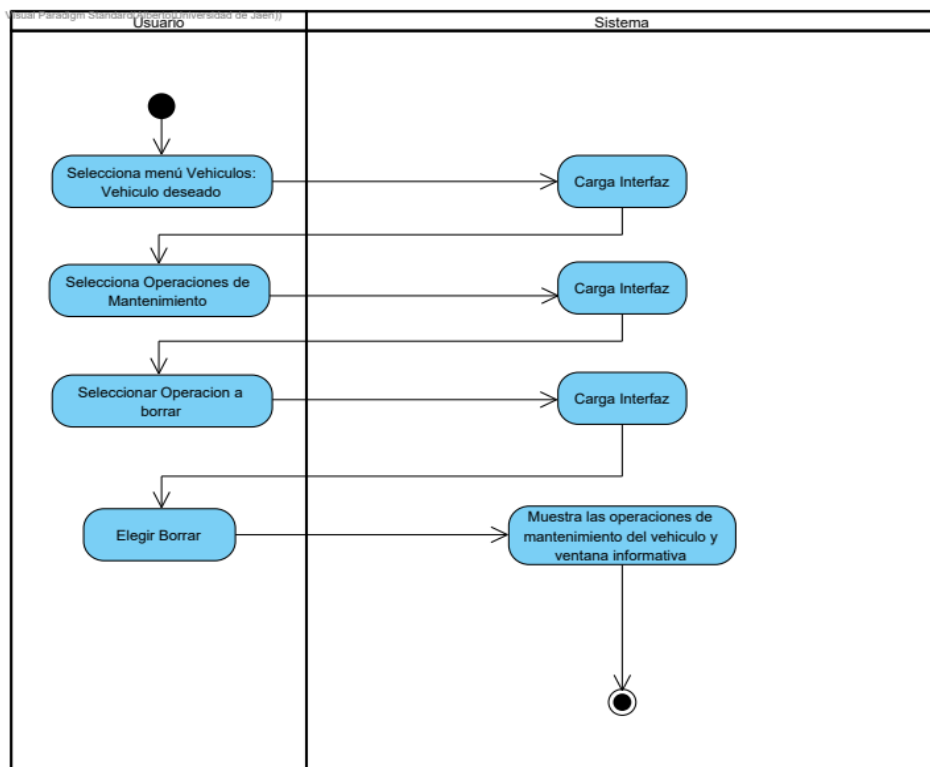


Figura 33: D016 Borrar Operación de Mantenimiento (H016)

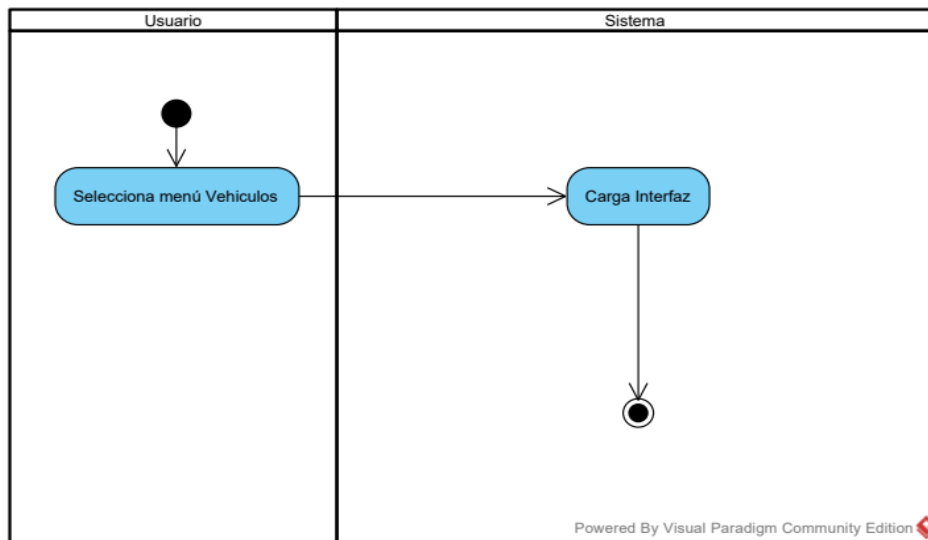


Figura 34: D017 Listar Vehículos (H017)

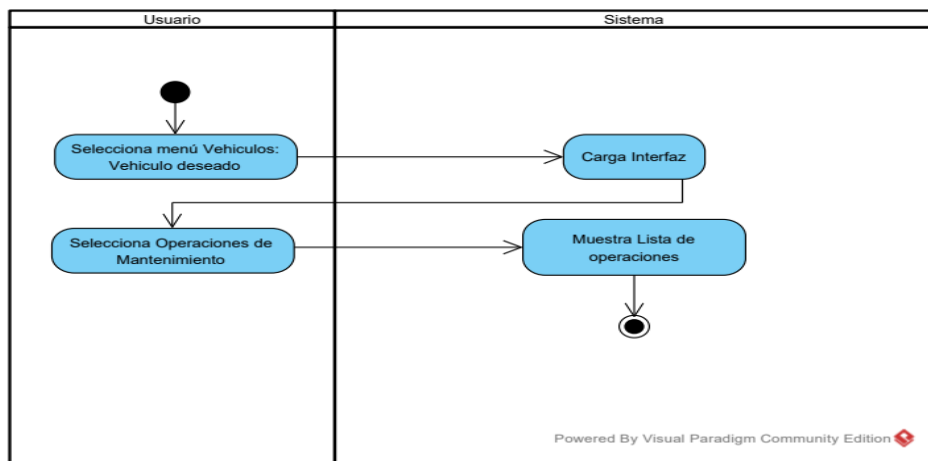


Figura 35: D018 Listar Operaciones de Mantenimiento(H018)

3.3.3 Planificación de la iteración

Para esta iteración el equipo habrá de diseñar e implementar los elementos referentes al módulo de vehículos y finalizar las tareas atrasadas en la iteración anterior.

Para el desarrollo del módulo de vehículos se espera que la experiencia adquirida en el módulo de alumnos sea extensible a este otro módulo reduciendo así en gran medida el tiempo de desarrollo que se dio en la primera iteración. A pesar de ello se habrá de dedicar un periodo de tiempo para documentación previa sobre paginación para las peticiones a la API.

En la Tabla 33 se muestra un esquema del *Backlog de Iteración* de la tercera iteración.

Tabla 33: *Backlog de Iteración – Iteración 3*

BACKLOG DE ITERACIÓN		Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas
Iteración Iteración 3 – Vehículos			Puntos de Historia 15
ID	Tarea		Proyecto
H011 Listar Exámenes			
T0217	Implementación Listar exámenes		Cliente de escritorio
T0218	Implementación Listar exámenes		Cliente Móvil
H012 Añadir Vehículo			
T0301	Diseño Añadir Vehículo		Servidor
T0302	Diseño Añadir Vehículo		Cliente de escritorio
T0303	Diseño Añadir Vehículo		Cliente Móvil
T0304	Implementación Añadir Vehículo		Servidor
T0305	Implementación Añadir Vehículo		Cliente de escritorio
T0306	Implementación Añadir Vehículo		Cliente Móvil
H013 Modificar Vehículo			
T0307	Diseño Modificar Vehículo		Servidor
T0308	Diseño Modificar Vehículo		Cliente de escritorio
T0309	Diseño Modificar Vehículo		Cliente Móvil
T0310	Implementación Modificar Vehículo		Servidor
T0311	Implementación Modificar Vehículo		Cliente de escritorio
T0312	Implementación Modificar Vehículo		Cliente Móvil
H014 Borrar Vehículo			
T0313	Diseño Borrar Vehículo		Servidor
T0314	Diseño Borrar Vehículo		Cliente de escritorio
T0315	Diseño Borrar Vehículo		Cliente Móvil

T0316	Implementación Borrar Vehículo	Servidor
T0317	Implementación Borrar Vehículo	Cliente de escritorio
T0318	Implementación Borrar Vehículo	Cliente Móvil
H015 Añadir Operación de mantenimiento		
T0319	Diseño Añadir Op. Mantenimiento	Servidor
T0320	Diseño Añadir Op. Mantenimiento	Cliente de escritorio
T0321	Diseño Añadir Op. Mantenimiento	Cliente Móvil
T0322	Implementación Añadir Op. Mantenimiento	Servidor
T0323	Implementación Añadir Op. Mantenimiento	Cliente de escritorio
T0324	Implementación Añadir Op. Mantenimiento	Cliente Móvil
H016 Borrar Operaciones de mantenimiento		
T0325	Diseño Borrar Op. Mantenimiento	Servidor
T0326	Diseño Borrar Op. Mantenimiento	Cliente de escritorio
T0327	Diseño Borrar Op. Mantenimiento	Cliente Móvil
T0328	Implementación Borrar Op. Mantenimiento	Servidor
T0329	Implementación Borrar Op. Mantenimiento	Cliente de escritorio
T0330	Implementación Borrar Op. Mantenimiento	Cliente Móvil
H017 Listar Vehículos		
T0331	Diseño Listar Vehículos	Servidor
T0332	Diseño Listar Vehículos	Cliente de escritorio
T0333	Diseño Listar Vehículos	Cliente Móvil
T0334	Implementación Listar Vehículos	Servidor
T0335	Implementación Listar Vehículos	Cliente de escritorio
T0336	Implementación Listar Vehículos	Cliente Móvil
H018 Listar Operaciones de mantenimiento		
T0337	Diseño Listar Op. Mantenimiento	Servidor
T0338	Diseño Listar Op. Mantenimiento	Cliente de escritorio
T0339	Diseño Listar Op. Mantenimiento	Cliente Móvil

T0340	Implementación Listar Op. Mantenimiento	Servidor
T0341	Implementación Listar Op. Mantenimiento	Cliente de escritorio
T0342	Implementación Listar Op. Mantenimiento	Cliente Móvil
T0343	Pruebas de Unidad	Servidor

3.3.4 Diseño

3.3.4.1 *Paginación*

La principal diferencia en cuanto a la API de este módulo con respecto al anterior es que varias llamadas están destinadas a devolver listas de información que pueden llegar a ser muy extensas; por tanto, deberemos diseñar algún sistema de paginación que nos permita recabar la información de forma parcial y acotada para no generar una carga excesiva sobre la conexión.

A la hora de diseñar la paginación debemos saber que existen diferentes puntos de vista a la hora de solucionar este problema. A continuación, vamos a explorar algunos de estos métodos, y estudiar sus ventajas y desventajas para posteriormente elegir cuál implementar.

1. Paginación Límite – Sesgo

Es el método más sencillo de implementar; sin embargo, tiene una serie de desventajas. Consiste en utilizar dos parámetros para acotar la consulta: límite y sesgo. El primero indica el número de líneas que se quieren obtener, y el segundo el número de línea por la cual se empiezan a devolver los datos.

La petición tendría esta forma:

```
GET /objetos?sesgo=40&limite=10
```

En lenguaje SQL, la consulta resultante sería como la siguiente:

```
SELECT * FROM tabla ORDER BY id LIMIT 10 OFFSET 40
```

Donde 10 sería el límite y 40 la posición de inicio de la consulta. Este método presenta dos importantes desventajas:

1. Bajo rendimiento con sesgos altos: A la hora de aplicar la directiva OFFSET, el sistema de gestión de bases de datos debe recorrer las primeras filas de la consulta una a una contando hasta llegar al número indicado, por lo que se vuelve una operación costosa con grandes sesgos.
2. Deriva de página: Aún más grave que el primero es la deriva de página, un problema de inconsistencia que provoca que si se añaden nuevas filas antes del sesgo, un usuario puede volver a recibir elementos repetidos al solicitar la siguiente página.

Este método es adecuado si estas seguro que los datos del sistema no van a crecer demasiado y no es importante que las peticiones puedan ser inconsistentes. El sistema de repositorios de Spring dispone de una API para implementar este método de forma automática.

2. Paginación por clave

La paginación por clave utiliza campos de la última entrada de la página para obtener la siguiente.

La petición tendría esta forma:

```
GET /objetos?fechaInicio=2018-01-03&limite=10
```

Un ejemplo de consulta SQL sería:

```
SELECT * FROM tabla WHERE fecha <= '2018-01-03' ORDER BY  
fecha, id LIMIT 10
```

Este método no sufre del problema de consistencia que tiene la paginación Límite – Sesgo y posee un rendimiento constante sin tener en cuenta el número de página. Este método tiene sin embargo otras desventajas:

1. Alto acoplamiento entre el mecanismo de paginación y el filtrado, obligando al cliente a utilizar filtros siempre en las consultas
2. Solo podemos acceder a páginas contiguas.

3. Seek Method o método de búsqueda

Es una variante de la paginación por clave que solventa los problemas de acoplamiento derivados de esta. Solicita a la API siempre un id, y después busca y extrae los campos necesarios para realizar la consulta.

La petición tendría esta forma:

```
GET /objetos?idInicio=5&limite=10
```

El ejemplo en lenguaje SQL sería el siguiente:

Si ordenamos unicamente por id

```
SELECT * FROM tabla WHERE id > 5 LIMIT 10
```

Si ordenamos por otro campo deberemos realizar dos consultas

```
SELECT email AS email_inicio FROM tabla WHERE id = 5;
```

```
SELECT * FROM tabla WHERE email >= email_inicio ORDER BY  
email, id LIMIT 10;
```

Este es el método más complejo de implementar, pero proporciona unos buenos resultados a la hora de paginar a la vez que es intuitivo para el consumidor de la API. Hemos decidido implementar la paginación siguiendo este método frente al Límite – Sesgo, principalmente para deshacernos de los problemas de inconsistencia entre páginas en listas ordenadas por fecha y referencia como las operaciones de mantenimiento de un coche, y pensando en que el sistema quede lo más preparado posible para posibles modificaciones futuras que incluyan listados extensos de información.

3.3.4.2 Servidor

Siguiendo con el patrón de diseño anterior diseñaremos el módulo correspondiente a la gestión y registro de la flota de vehículos de la autoescuela.

El controlador *ServicioVehiculosREST* expone al exterior la interfaz que se describe en la Tabla 34.

Tabla 34: Especificación API REST - Vehículos

ESPECIFICACIÓN API REST	Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas
----------------------------	--	---

Obtener Vehículos	URL Recurso	Método
	<i>/vehiculos/</i>	GET
	Parámetros	Tipo
	inicio : String	RequestParam
	tamPag : int	RequestParam
Obtener Vehículo	URL Recurso	Método
	<i>/vehiculos/{matricula}</i>	GET
	Parámetros	Tipo
	matricula : String	Path
Nuevo Vehículo	URL Recurso	Método
	<i>/vehiculos/{matricula}</i>	POST
	Parámetros	Tipo
	matricula : String	Path
	vehiculo : VehiculoDTO	RequestBody
Modificar Vehículo	URL Recurso	Método
	<i>/vehiculos/{matricula}</i>	PUT
	Parámetros	Tipo
	matricula : String	Path
	vehiculo : VehiculoDTO	RequestBody
Borrar Vehículo	URL Recurso	Método
	<i>/vehiculos/{matricula}</i>	DELETE
	Parámetros	Tipo
	matricula : String	Path
Obtener Mantenimientos	URL Recurso	Método
	<i>/vehiculos/{matricula}/mantenimientos/</i>	GET
	Parámetros	Tipo
	matricula : String	Path
	inicio : String	RequestParam

	tamPag : int	RequestParam
Nuevo Mantenimiento	URL Recurso	Método
	/vehiculos/{matricula}/mantenimientos/	POST
	Parámetros	Tipo
	matricula : String	Path
	mantenimiento: OpMantenimientoDTO	RequestBody
Modificar Mantenimiento	URL Recurso	Método
	/vehiculos/{matricula}/mantenimientos/{num_mantenimiento}	PUT
	Parámetros	Tipo
	matricula : String	Path
	num_mantenimiento : int	Path
	mantenimiento: OpMantenimientoDTO	RequestBody
Borrar Mantenimiento	URL Recurso	Método
	/vehiculos/{matricula}/mantenimientos/{num_mantenimiento}	PUT
	Parámetros	Tipo
	matricula : String	Path
	num_mantenimiento : int	Path

Aquí en la Figura 36 se muestra el esquema de diagrama de clases del modulo de vehículos. Y en la Figura 37 el diagrama de la lógica de negocio del módulo.

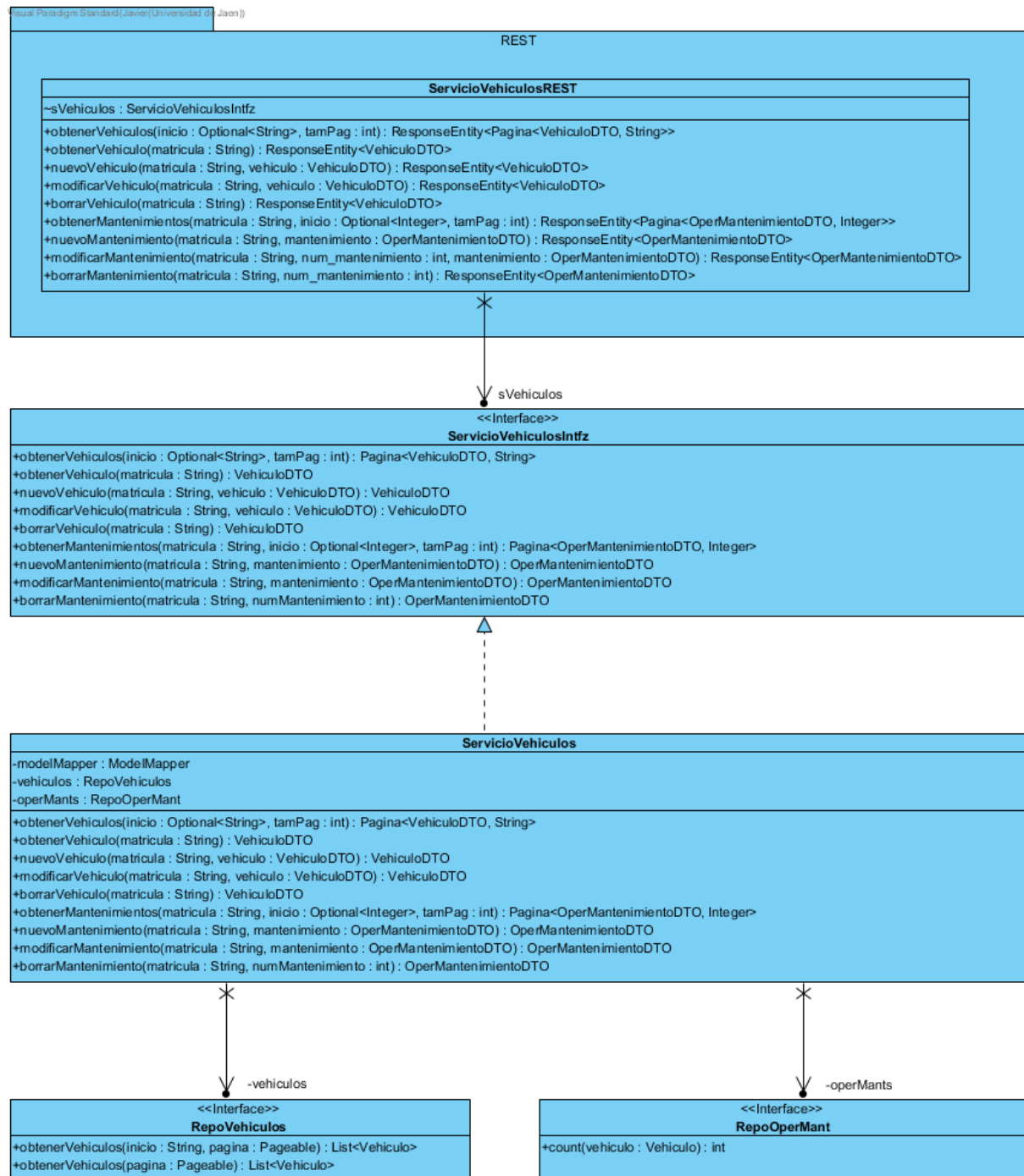


Figura 36: Diagrama de Clases Vehículos - Capa de Servicios

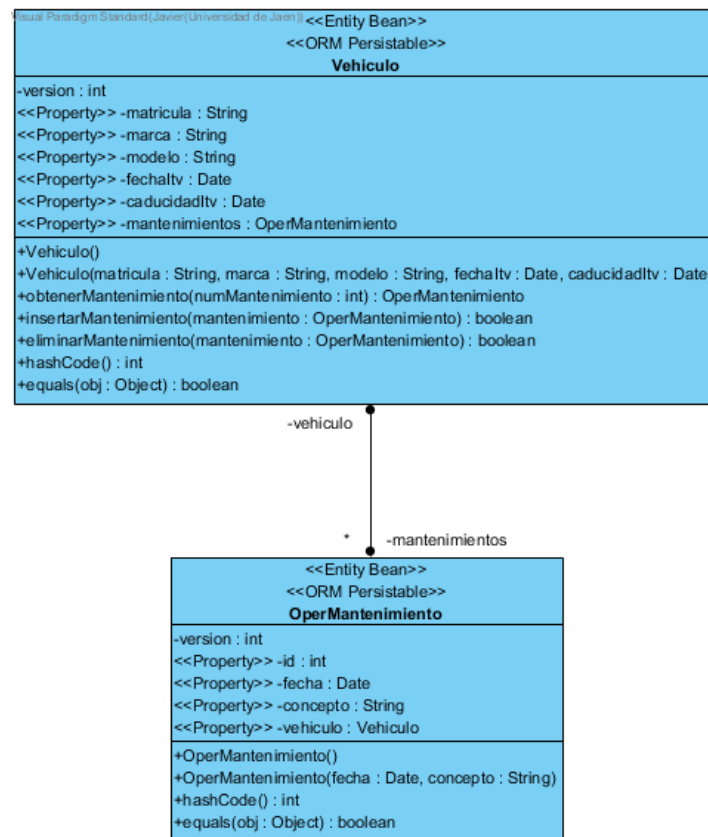


Figura 37: Diagrama de Clases Vehículos - Lógica de Negocio

3.3.4.3 Cliente Móvil

En esta iteración he diseñado los fragmentos para crear, editar, mostrar y borrar vehículos, así mismo como los apropiados para las operaciones de mantenimiento de estos, siguiendo el esquema de diseño que apliqué en el resto de la aplicación, he podido extenderlo sin la necesidad de cambios notables.

En el paquete REST he añadido la clase pagina que proviene del servidor y que será necesaria para ciertas peticiones. La Figura 38 refleja el diagrama de clases al final de la iteración.



El diagrama de flujo de la interfaz de usuario (UI) para el sistema de gestión de exámenes se estructura de la siguiente manera:

- Inicio:** El usuario puede acceder directamente a **crearAlumno** o **crearVehiculo**.
- Gestión de Alumnos:**
 - crearAlumno** y **editarAlumno** llevan a **mostrarAlumno**.
 - mostrarAlumno** tiene un botón de búsqueda que lleva a **buscarAlumno**.
 - mostrarAlumno** y **editarAlumno** llevan a **mostrarExpediente**.
 - mostrarAlumno** y **editarAlumno** llevan a **mostrarExamen**.
- Gestión de Vehículos:**
 - crearVehiculo** y **editarVehiculo** llevan a **listaVehiculos**.
 - listaVehiculos** tiene un botón de búsqueda que lleva a **buscarAlumno**.
 - listaVehiculos** lleva a **mostrarExamen**.
- Gestión de Expedientes y Exámenes:**
 - mostrarExpediente** y **mostrarExamen** llevan a **editarExpediente**.
 - mostrarExpediente** y **mostrarExamen** llevan a **crearExpediente**.
 - mostrarExamen** y **crearExpediente** llevan a **crearExamen**.
 - crearExamen** lleva a **mostrarExamen**.
 - mostrarExamen** y **crearExamen** llevan a **mostrarOPMante...**.
 - mostrarOPMante...** lleva a **crearOPMante...**.

Figura 39: Grafo Navegación – Iteración 3, Cliente Móvil

3.3.5 Resumen de progreso

Al final de la iteración cumplimos con todas las tareas previstas y aquellas atrasadas. Se percibe el incremento de la experiencia de los desarrolladores en el proyecto. A pesar de eso, en el cliente móvil aún no se han realizado las mejoras visuales que se propusieron en la primera iteración.

En la Figura 40 se muestra un resumen del progreso durante la iteración. Y en la Figura 41 y Figura 42 se ilustra el *Burdown Chart* de la iteración.

ID	Tareas	Proyecto	Tareas Inicio	1/07/2019 Dia 1	2/07/2019 Dia 2	3/07/2019 Dia 3	4/07/2019 Dia 4	5/07/2019 Dia 5	6/07/2019 Dia 6	7/07/2019 Dia 7	8/07/2019 Dia 8	9/07/2019 Dia 9	10/07/2019 Dia 10	11/07/2019 Dia 11	12/07/2019 Dia 12	Progreso
H011 Listar Exámenes																
T0217	Implementación Listar exámenes	Cliente de escritorio	1	1												100.0%
T0218	Implementación Listar exámenes	Cliente Móvil	1	1												100.0%
H012 Añadir Vehículo																
T0301	Diseño Añadir Vehículo	Servidor	1	1												100.0%
T0302	Diseño Añadir Vehículo	Cliente de escritorio	1	1												100.0%
T0303	Diseño Añadir Vehículo	Cliente Móvil	1	1												100.0%
T0304	Implementación Añadir Vehículo	Servidor	1		1											100.0%
T0305	Implementación Añadir Vehículo	Cliente de escritorio	1		1											100.0%
T0306	Implementación Añadir Vehículo	Cliente Móvil	1		1											100.0%
H013 Modificar Vehículo																
T0307	Diseño Modificar Vehículo	Servidor	1			1										100.0%
T0308	Diseño Modificar Vehículo	Cliente de escritorio	1			1										100.0%
T0309	Diseño Modificar Vehículo	Cliente Móvil	1			1										100.0%
T0310	Implementación Modificar Vehículo	Servidor	1				1									100.0%
T0311	Implementación Modificar Vehículo	Cliente de escritorio	1				1									100.0%
T0312	Implementación Modificar Vehículo	Cliente Móvil	1				1									100.0%
H014 Borrar Vehículo																
T0313	Diseño Borrar Vehículo	Servidor	1				1									100.0%
T0314	Diseño Borrar Vehículo	Cliente de escritorio	1				1									100.0%
T0315	Diseño Borrar Vehículo	Cliente Móvil	1				1									100.0%
T0316	Implementación Borrar Vehículo	Servidor	1					1								100.0%
T0317	Implementación Borrar Vehículo	Cliente de escritorio	1						1							100.0%
T0318	Implementación Borrar Vehículo	Cliente Móvil	1							1						100.0%
H015 Añadir Operación de mantenimiento																
T0319	Diseño Añadir Mantenimiento	Servidor	1					1								100.0%
T0320	Diseño Añadir Mantenimiento	Cliente de escritorio	1					1								100.0%
T0321	Diseño Añadir Mantenimiento	Cliente Móvil	1						1							100.0%
T0322	Implementación Añadir Mantenimiento	Servidor	1							1						100.0%
T0323	Implementación Añadir Mantenimiento	Cliente de escritorio	1								1					100.0%
T0324	Implementación Añadir Mantenimiento	Cliente Móvil	1									1				100.0%
H016 Borrar Operaciones de mantenimiento																
T0325	Diseño Borrar Mantenimiento	Servidor	1							1						100.0%
T0326	Diseño Borrar Mantenimiento	Cliente de escritorio	1							1						100.0%
T0327	Diseño Borrar Mantenimiento	Cliente Móvil	1								1					100.0%
T0328	Implementación Borrar Mantenimiento	Servidor	1									1				100.0%
T0329	Implementación Borrar Mantenimiento	Cliente de escritorio	1										1			100.0%
T0330	Implementación Borrar Mantenimiento	Cliente Móvil	1											1		100.0%
H017 Listar Vehículos																
T0331	Diseño Listar Vehículos	Servidor	1								1					100.0%
T0332	Diseño Listar Vehículos	Cliente de escritorio	1									1				100.0%
T0333	Diseño Listar Vehículos	Cliente Móvil	1										1			100.0%
T0334	Implementación Listar Vehículos	Servidor	1											1		100.0%
T0335	Implementación Listar Vehículos	Cliente de escritorio	1												1	100.0%
T0336	Implementación Listar Vehículos	Cliente Móvil	1													100.0%
H018 Listar Operaciones de mantenimiento																
T0337	Diseño Listar OpMantenimiento	Servidor	1								1					100.0%
T0338	Diseño Listar OpMantenimiento	Cliente de escritorio	1									1				100.0%
T0339	Diseño Listar OpMantenimiento	Cliente Móvil	1										1			100.0%
T0340	Implementación Listar OpMantenimiento	Servidor	1											1		100.0%
T0341	Implementación Listar OpMantenimiento	Cliente de escritorio	1												1	100.0%
T0342	Implementación Listar OpMantenimiento	Cliente Móvil	1													100.0%
T0343	Pruebas de Unidad	Servidor	1												0.2	0.8
Tareas Restantes				45	42	39	34	27	21	13	7	4	0.8	0		
Puntos de Historia Restantes				15	13	13	10	7	6	4	4	4	1	0		
Esfuerzo Ideal				15	13.5	12	10.5	9	7.5	6	4.5	3	1.5	0		

Figura 40: Resumen de progreso - Iteración 3

Burndown Chart

Iteración 3

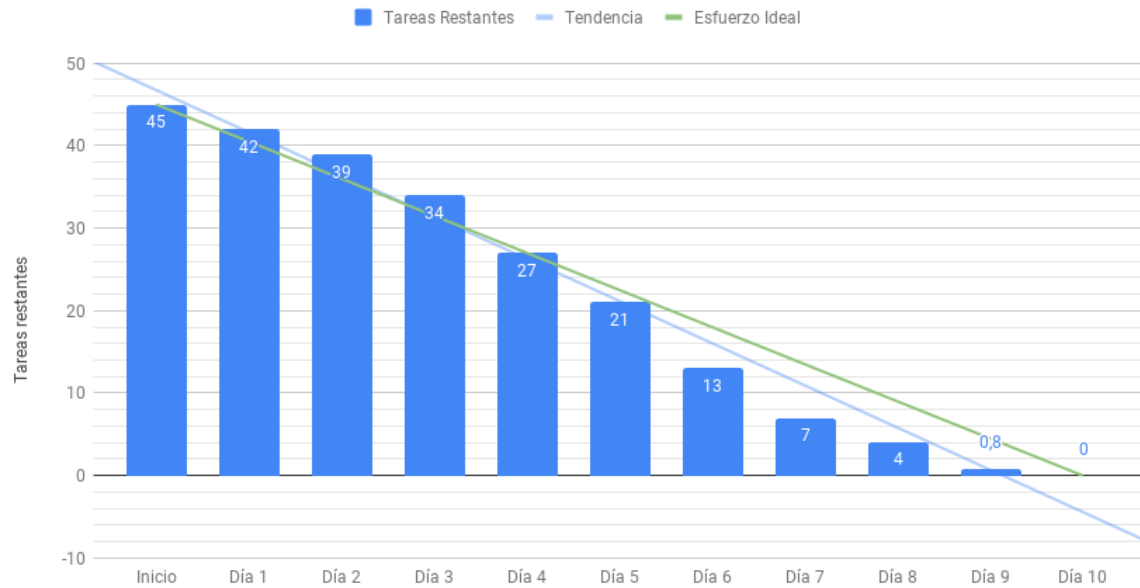


Figura 41: Burndown Chart – Tareas Iteración 3

Burndown Chart

Iteración 3

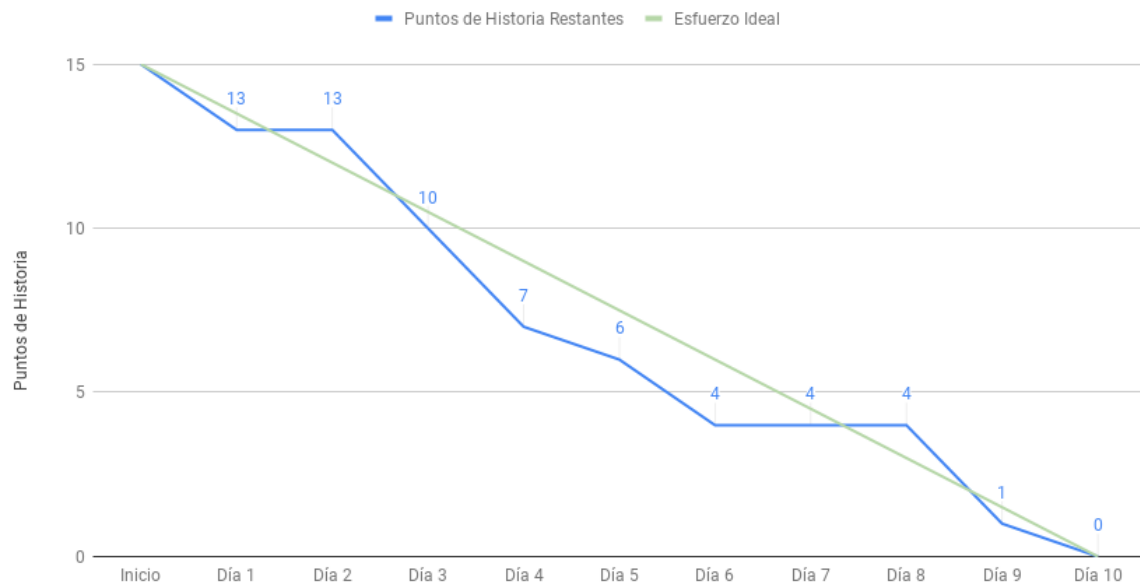


Figura 42: Burndown Chart – Iteración 3

3.3.6 Entrega y reunión final de la iteración

Se muestra la ejecución del módulo de vehículos al cliente. El usuario prueba las distintas funcionalidades disponibles y sugiere la incorporación de las fechas en

las que el vehículo pasó la ITV y la fecha en la que esta caduca. Respecto a las operaciones de mantenimiento se pide que se muestren ordenadas en función de su fecha. Concretamente que las más recientes sean las primeras en verse.

En general es cliente está muy satisfecho con la forma en la que el equipo ha manejado las tareas. El equipo ha tomado nota para desarrollar las futuras tareas de forma similar.

3.4 Cuarta Iteración

En la cuarta iteración nos disponemos a elaborar la gestión de documentos, la idea inicial del cliente es poder generar sus impresos utilizados en el día a día con sus clientes y socios. Para ellos es bastante importante disponer de esta funcionalidad, ya que están acostumbrados a trabajar con documentos impresos y muchos aspectos que no cubre esta aplicación seguirán realizándose mediante los mismos.

3.4.1 Historias de usuario

A continuación se muestran las Historias de Usuario correspondientes a esta iteración resumidas en las tablas 35, 36 y 37.

Tabla 35: H019 Generar Documento Contrato

HISTORIA DE USUARIO		Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas
ID H019	Estimación 3	Prioridad Must Have	
Como Administrador	Quiero Generar un Documento del Contrato	Para Agilizar los procesos administrativos de la autoescuela	
Criterios de satisfacción - Dado que se selecciona un expediente valido, se genera un documento para contrato con los datos de ese expediente			
Redactado por Alberto Carrillo Ortega	Fecha 13/07/2019	Aprobado por Javier Martínez Cavanillas	Fecha 13/07/2019

Tabla 36: H020 Generar Documento Expediente

HISTORIA DE USUARIO	Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas
---------------------	--	---

ID H020	Estimación 3	Prioridad Must Have	
Como Administrador	Quiero Generar un Documento de un Expediente	Para Tener un documento con la información de expediente de un alumno	
Criterios de satisfacción - Dado que se selecciona un expediente valido, se genera un documento con los datos y exámenes de ese expediente			
Redactado por Alberto Carrillo Ortega	Fecha 13/07/2019	Aprobado por Javier Martínez Cavanillas	Fecha 13/07/2019

Tabla 37: H021 Generar Documento Vehículo

HISTORIA DE USUARIO		Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas
ID H021	Estimación 3	Prioridad Must Have	
Como Administrador	Quiero Generar un Documento de un Vehículo	Para Tener un documento con la información de un vehículo de la flota	
Criterios de satisfacción - Dado que se selecciona un vehículo valido, se genera un documento con los datos y operaciones de mantenimiento de ese vehiculo			
Redactado por Alberto Carrillo Ortega	Fecha 13/07/2019	Aprobado por Javier Martínez Cavanillas	Fecha 13/07/2019

3.4.2 Análisis

Las Figuras 43, 44 y 45 mostradas a continuación se desarrollan los diagramas de actividades correspondientes a las historias **H019**, **H020** y **H021**.

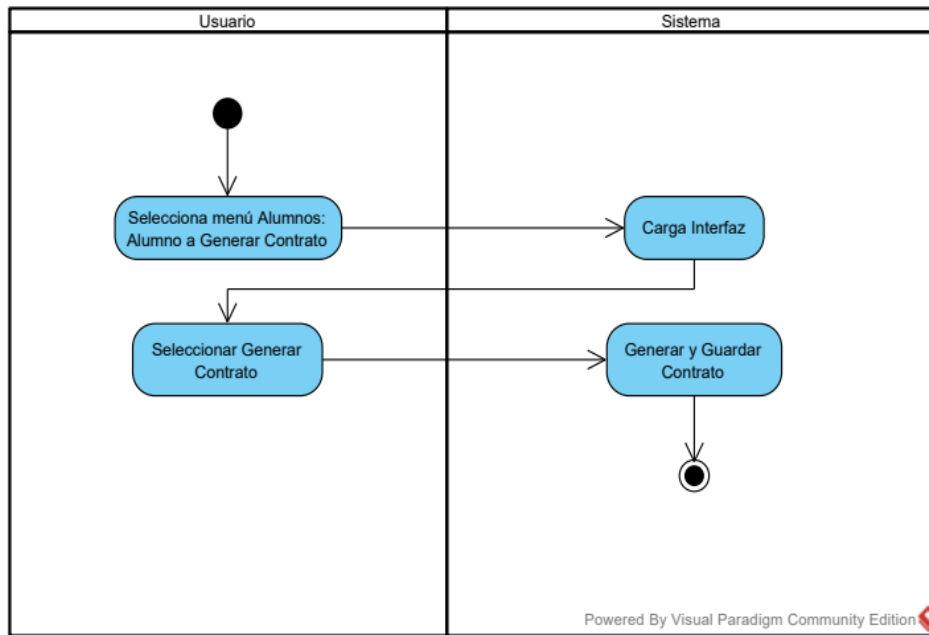


Figura 43: DA019 Generar Documento Contrato (H019)

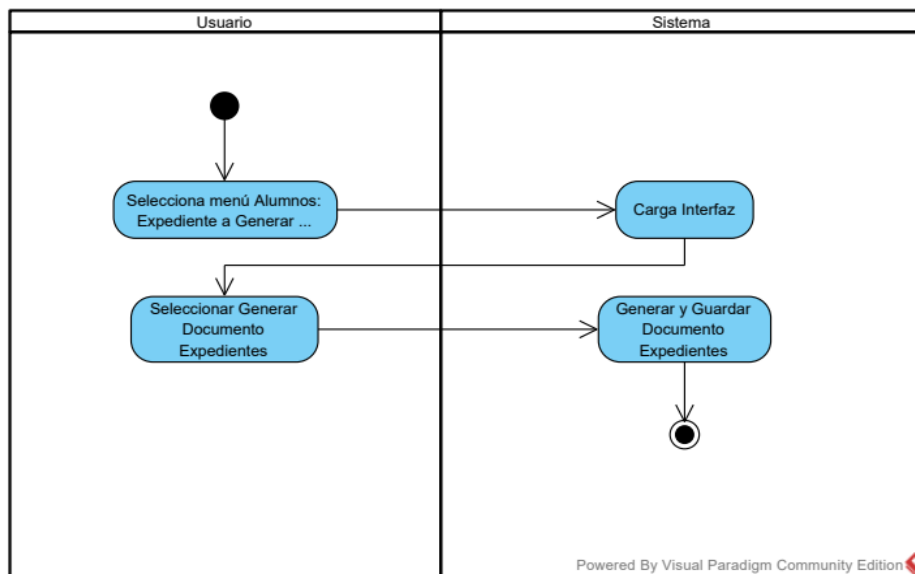


Figura 44: DA020 Generar Documento Expedientes (H020)

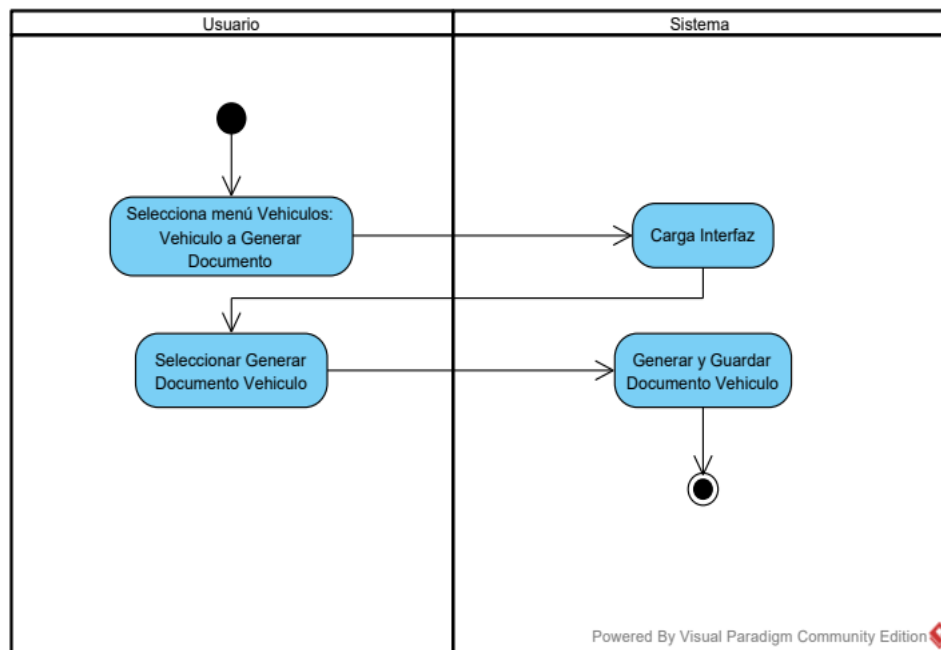


Figura 45: DA021 Generar Documento Vehiculos (H021)

3.4.3 Planificación de la iteración

El equipo de desarrollo desconocía las soluciones para creación de documentos PDF disponibles, por tanto, en esta iteración es importante tomarse su tiempo en la fase de diseño, que incluye un estudio sobre las distintas soluciones posibles. Hasta no saber la tecnología y diseño a utilizar no fue posible comenzar tareas como la maquetación de los documentos o la implementación.

En la Tabla 38 se muestra un esquema del *Backlog de Iteración* de la tercera iteración.

Tabla 38: Backlog de Iteración - Iteración 4

BACKLOG DE ITERACIÓN	Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas
Iteración Iteración 4 – Documentos		Puntos de Historia 15
ID	Tarea	Proyecto
Inicio de la iteración		
T0401	Estudio Soluciones PDF	Servidor
T0402	Elección de Tecnología PDF	Servidor
H019 Generar Documento Contrato		

T0403	Maquetación Contrato	Servidor
T0404	Diseño Contrato de enseñanza	Servidor
T0405	Implementación Contrato de enseñanza	Servidor
T0406	Implementación descarga de Contrato de enseñanza	Cliente de escritorio
T0407	Implementación descarga de Contrato de enseñanza	Cliente Móvil
H020 Generar Documento Expediente		
T0408	Maquetación Ficha de expediente	Servidor
T0409	Diseño Ficha de expediente	Servidor
T0410	Implementación Ficha de expediente	Servidor
T0411	Implementación descarga de Ficha de expediente	Cliente de escritorio
T0412	Implementación descarga de Informe de vehículo	Cliente Móvil
H021 Generar Documento Vehículo		
T0413	Maquetación Informe de vehículo	Servidor
T0414	Diseño Informe de vehículo	Servidor
T0415	Implementación Informe de vehículo	Servidor
T0416	Implementación descarga de Informe de vehículo	Cliente de escritorio
T0417	Implementación descarga de Informe de vehículo	Cliente Móvil
T0418	Pruebas de Unidad	Servidor

3.4.4 Diseño

Desde el momento en que surgieron estos requisitos en la reunión con el cliente, supimos que esta iteración iba a ser el más difícil de estimar puesto que la generación de documentos nos era desconocida. Esta iteración supondría como tareas adicionales un estudio sobre las diferentes alternativas existentes para generar documentos PDF y elegir la más adecuada a la situación del proyecto. Con retraso acumulado sobre todo en el primera iteración, sabíamos que tomar una decisión acertada en este momento determinaría el resultado final del proyecto.

3.4.4.1 **Servidor**

Decidimos que los documentos debían ser generados en el servidor y no en los clientes, ya que varios clientes en varias plataformas deberían tener la misma versión de un documento. Además de esta forma también aseguramos que la información requerida por el documento esté presente en la base de datos. Esto simplificaría la implementación en los clientes, pero complicaría la transmisión de archivos por la red. Una vez decidido dónde generar los diferentes informes o documentos, pasamos a investigar sobre las diferentes tecnologías disponibles para presentar un PDF personalizado a partir de nuestra información en Java.

1. Apache FOP

Apache FOP (*Formating Object Procesor*) es un procesador de documentos impresos basado en el objetos de formato XSL (XSL – FO) pudiendo maquetar un documento en diferentes formatos de salida. Su funcionamiento consiste en leer una jerarquía de objetos de formato y renderizar el contenido de estos de acuerdo a la plantilla XSL en la salida. Los formatos soportados hasta el momento por esta biblioteca son PDF, PS, PCL, AFP, XML, Print, AWT y PNG.

Como ventajas podemos destacar que es una tecnología asentada en el mercado y bastante estable.

Sin embargo, cuenta con la desventaja de utilizar XSL para maquetar el documento. XSL es un lenguaje complicado y desconocido para el equipo de trabajo, por lo que el aprendizaje llevaría una gran cantidad de tiempo. Otra desventaja es que para documentos extensos Apache FOP demanda bastantes recursos, por lo que el servidor podría sufrir en cargas intensas. Este inconveniente podría controlarse mediante un buen sistema de cache, sin embargo, no es una solución completa.

2. Apache PDF Box

Apache PDF es una biblioteca Java que permite trabajar de distintas maneras con archivos PDF. Permite abrir documentos, rellenar formularios, manipular documentos ya existentes y crear nuevos documentos.

Tiene la ventaja de que es muy eficiente, generando documentos de gran longitud en poco tiempo. Sin embargo, presenta la gran desventaja de que el

maquetado se realiza colocando cada elemento sobre una posición concreta de la página; por poner un ejemplo, ni siquiera incluye soporte para crear un salto de línea al llegar al final de la página, el texto se escapa de los márgenes, por lo que habría que implementarlo manualmente, algo simplemente inviable con el tiempo disponible.

3. iText

iText es una plataforma de desarrollo escrita en Java que nos proporciona herramientas para integrar funcionalidades PDF en aplicaciones Java de forma parecida a Apache PDF Box, pero con una API de algo más alto nivel. Sin embargo, la licencia de esta biblioteca no es gratuita para proyectos comerciales, por lo que preferimos evitarlo en la medida de lo posible.

4. ThymeLeaf + FlyingSaucer

Flying Saucer es un procesador de contenido XHTML y CSS 2.1, escrito en Java. Nos permite visualizar el contenido de un documento XHTML en una ventana o volcarlo como contenido PDF para impresión, incluyendo algunas directivas especializadas de CSS3 para la maquetación de documentos impresos. La alternativa consiste en usar *ThymeLeaf* o cualquier otro motor de plantillas, como *Apache FreeMarker*, para generar un archivo XHTML con la información correspondiente, y después utilizar *Flying Saucer* para generar el documento PDF final.

Tras sopesar las diferentes alternativas disponibles, decidimos optar por utilizar *ThymeLeaf* y *Flying Saucer* para crear los documentos por varias razones:

- Utiliza HTML y CSS, lenguajes de maquetación conocidos y familiares para el equipo de desarrollo, más utilizados hoy en día que lenguajes como XSL y más sencillos y legibles.
- Proporciona un PDF completo sin necesidad de implementar mucho código rutinario adicional, a diferencia de otras alternativas de más bajo nivel como *Apache PDF Box* o *iText*.
- Proporciona un control bastante amplio sobre la elaboración del documento, pudiendo generar en un futuro los documentos no solamente como PDF sino como HTML.

- *ThymeLeaf* es sencillo de configurar y se integra bien con *Spring*, por lo que generar las plantillas y aplicarlas es más sencillo que con otras alternativas.

Por tanto, se ha realizado una clase *Documento*; en esta clase se encapsulan las operaciones encargadas de generar los archivos HTML y posteriormente el PDF a partir de plantillas de *ThymeLeaf*. Para cada tipo de documento se dedicará una clase heredada donde se definen la ruta de la plantilla y los diferentes datos de entrada necesarios para cumplimentar la información necesaria en el mismo. Por último, en la capa de servicios se le asignará al documento un nombre y se servirá al cliente a través de la interfaz REST correspondiente.

En la Tabla 39 se recoge la especificación de la API REST correspondiente a la petición de documentos.

Tabla 39: Especificación API REST - Documentos

ESPECIFICACIÓN API REST		Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas
Descargar ficha expediente	URL Recurso	Método	
	<i>/alumnos/{dni_alumno}/expedientes/{num_exp}/documentos/ficha_expediente</i>	GET	
	Parámetros	Tipo	
	dni_alumno : String	Path	
	num_exp : int	Path	
Descargar Informe de vehículo	URL Recurso	Método	
	<i>/vehiculos/{matricula}/documentos/informe_vehiculo</i>	GET	
	Parámetros	Tipo	
	matricula : String	Path	
Descargar Contrato de enseñanza	URL Recurso	Método	
	<i>/alumnos/{dni_alumno}/expedientes/{num_exp}/documentos/contrato_ensenanza</i>	GET	
	Parámetros	Tipo	
	dni_alumno : String	Path	

	num_exp : int	Path
--	---------------	------

El servicio de documentos del servidor (Figura 46) implementa la API REST de forma análoga a los servicios desarrollados en las anteriores iteraciones, la principal diferencia es que devuelven archivos PDF completos en forma de arrays de bytes y que a nivel de implementación, se añaden encabezados HTTP específicos para indicar el formato del documento y el nombre del archivo. El servicio necesita acceder a los alumnos y vehículos, mediante sus respectivos repositorios.

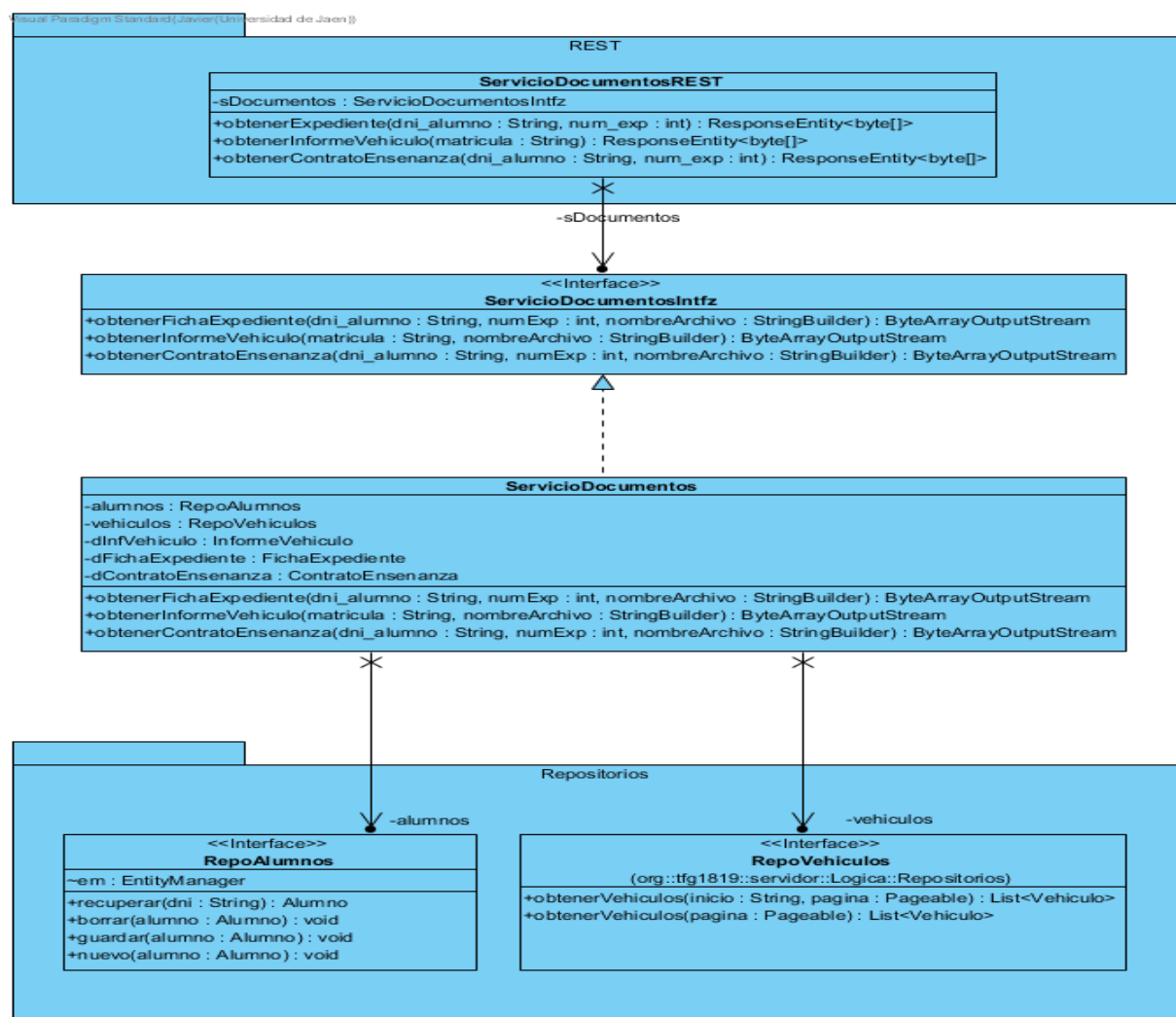


Figura 46: Diagrama de Clases – Documentos: Capa de Servicio, Servidor

En cuanto a la lógica (Figura 47), como se comentó anteriormente se disponen de varias clases que heredan de *Documento*, estas clases no son entidades, por lo que no son persistidas por el sistema ORM, únicamente albergan funcionalidad recibiendo como entrada datos de tipo *Alumno*, *Expediente*, *Vehículo* e *InfoEmpresa*. *InfoEmpresa* es una clase que tampoco es persistente y que es inicializada e inyectada por *Spring Framework* a partir de los archivos de configuración, esto se realiza así puesto que este tipo de configuración es estática y no varía en tiempo de ejecución por lo que almacenarla en el SGBD supone un esfuerzo y complejidad del sistema adicional innecesaria para el alcance actualmente contemplado.

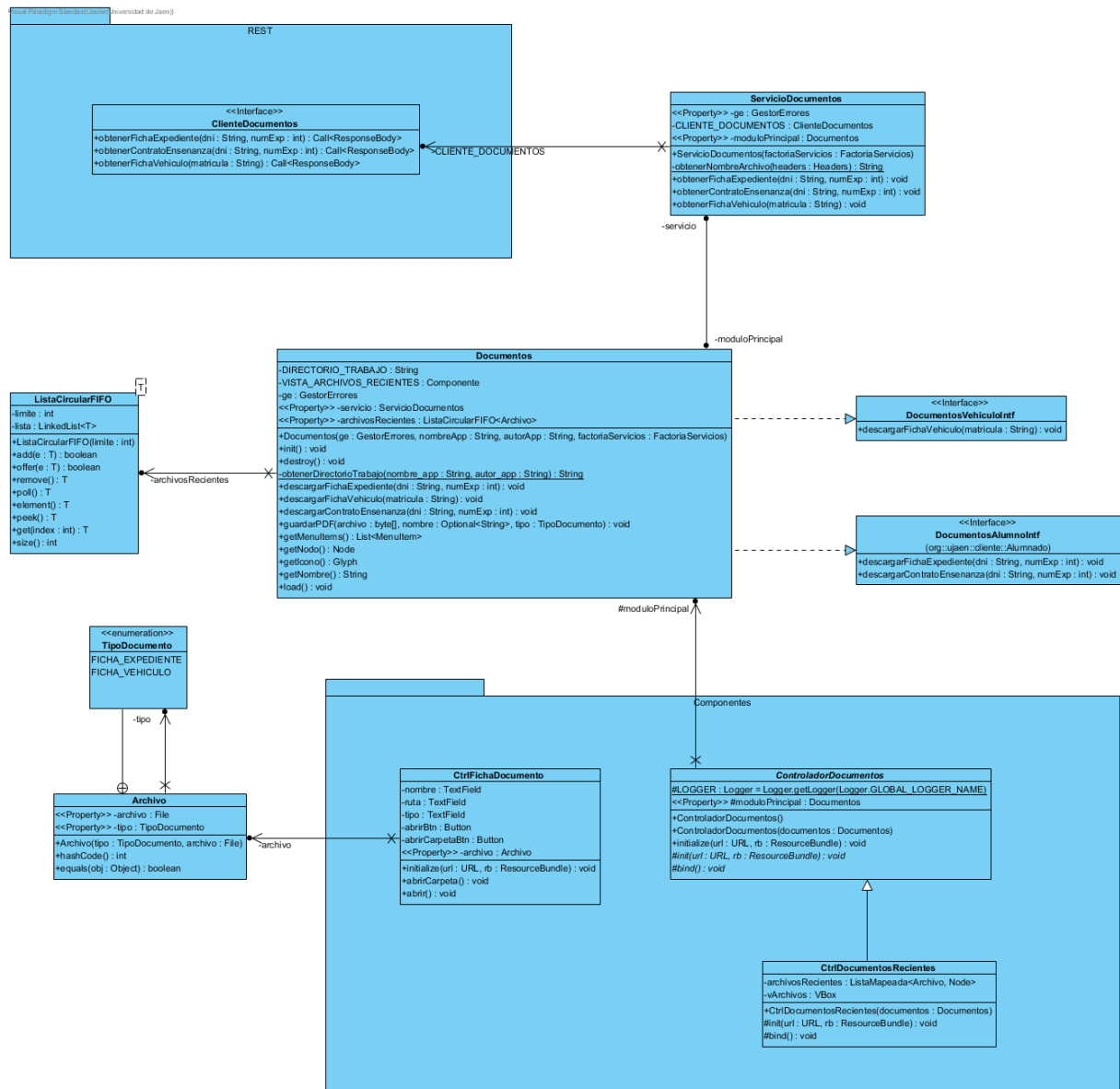


Figura 6: Diagrama de Clases - Módulo Documentos, Cliente Escritorio

3.4.4.2 Cliente Móvil

Tras realizar algunas pruebas con el almacenaje de archivos en Android descubrí que la primera vez que el usuario intentase almacenar un documento la aplicación tendría que pedir permisos. Lo cual una vez más se realiza de forma asíncrona y supone un problema pues la instancia del documento se perdería en la espera a la respuesta del usuario. Por ello opté por diseñar la parte de la aplicación destinada a los documentos como un *Singleton*. De este modo solo habría un documento en toda la aplicación accesible desde cualquier sitio y entonces desde el contexto en el que se procesa la petición de permisos, al recibir la respuesta de del

usuario de permiso concedido podemos finalmente almacenar el documento en la carpeta de descargas.

Las vistas de los fragmentos *MostrarExpedientes* y *MostrarVehiculos* incorporarán nuevas funcionalidades para realizar peticiones de documentos al servidor. La interfaz *RecibirRespuestaDocumentos* se encargará de recibir la respuesta del servidor. Cuando la tenga en caso de ser una respuesta afirmativa actualizará los campos de el *Singleton Documentos* e invocará el método *guardarDocumentos()*. Este método intentará almacenar el documento en la carpeta de descargas del dispositivo en caso de tener permisos, de no tenerlos entonces se realiza una petición por ellos y se espera la respuesta del usuario.

La respuesta se recibirá en el método *OnRequestPermission()* de la actividad principal *BarraNavegacion*. Si el usuario concede los permisos entonces la aplicación almacenará el documento, si no los concede entonces se mostrará una alerta avisando de que son necesarios para guardar documentos.

La Figura 48 muestra el diagrama de clases tras añadir las funcionalidades de esta iteración.



3.4.5 Resumen de progreso

Escuela Politécnica Superior de Jaén

ID	Tarea	Proyecto	Tareas Inicio	15/07/2019 Día 1	16/07/2019 Día 2	17/07/2019 Día 3	18/07/2019 Día 4	19/07/2019 Día 5	22/07/2019 Día 6	23/07/2019 Día 7	24/07/2019 Día 8	25/07/2019 Día 9	26/07/2019 Día 10	Progreso
Inicio de la iteración														
T0401	Estudio Soluciones PDF	Servidor	1	0,4	0,6									100.0%
T0402	Elección de Tecnología PDF	Servidor	1			0,5	0,5							100.0%
H019 Generar Documento Contrato														
T0403	Maquetación Contrato	Servidor	1					1						100.0%
T0404	Diseño Contrato de enseñanza	Servidor	1					1						100.0%
T0405	Implementación Contrato de enseñanza	Servidor	1						1					100.0%
T0406	Implementación descarga de Contrato de enseñanza	Cliente de escritorio	1							1				100.0%
T0407	Implementación descarga de Contrato de enseñanza	Cliente Móvil	1								1			100.0%
H020 Generar Documento Expediente														
T0408	Maquetación Ficha de expediente	Servidor	1						1					100.0%
T0409	Diseño Ficha de expediente	Servidor	1						1					100.0%
T0410	Implementación Ficha de expediente	Servidor	1							1				100.0%
T0411	Implementación descarga de Ficha de expediente	Cliente de escritorio	1								1			100.0%
T0412	Implementación descarga de Informe de vehículo	Cliente Móvil	1									1		100.0%
H021 Generar Documento Vehículo														
T0413	Maquetación Informe de vehículo	Servidor	1							1				100.0%
T0414	Diseño Informe de vehículo	Servidor	1								1			100.0%
T0415	Implementación Informe de vehículo	Servidor	1									1		100.0%
T0416	Implementación descarga de Informe de vehículo	Cliente de escritorio	1										1	100.0%
T0417	Implementación descarga de Informe de vehículo	Cliente Móvil	1										1	100.0%
T0418	Pruebas de Unidad	Servidor	1											100.0%
Tareas Restantes				18	17,6	17	16,5	16	14	11	7	4	2	0
Esfuerzo Ideal				18	16	14	13	11	9	7	5	4	2	0
Puntos de Historia Restantes				15	15	15	15	15	15	10	5	5	0	0
Esfuerzo Ideal				15	13,5	12	10,5	9	7,5	6	4,5	3	1,5	0

Figura 49: Resumen de progreso - Iteración 4

Burndown Chart

Iteración 4



Figura 50: Burndown Chart – Iteración 4

Burndown Chart

Iteración 4



Figura 51: Burndown Chart – Iteración 4

3.4.6 Entrega y reunión final de Iteración

Como entrega final de esta iteración se proporciona al cliente el conjunto de documentos generados con datos de ejemplo, para que revise si son correctos y los prototipos de la aplicaciones. Le gusta especialmente el encabezado de los mismos en los cuales se muestra el logo e información referente a la empresa, entre todos nos damos cuenta de que la aplicación todavía no dispone de un distintivo de marca que la haga única para la *Autoescuela Zona L* y acordamos que en la última iteración debería incluirse en la aplicación de escritorio mejoras visuales que concordasen con la imagen de marca de la empresa.

Para ello elaboramos un apartado *preferencias* en la interfaz de usuario. En este apartado se incluiría la información básica sobre la empresa para ser reconocible por el usuario como aplicación exclusiva y además se irán incluyendo los distintos ajustes de parámetros que pueda modificar el usuario a la hora de su uso diario. Teniendo en cuenta el retraso del proyecto, por cuestiones de tiempo, no se han podido incluir las opciones del programa ajustables por el usuario. No obstante, se ha estado comentando la posibilidad de incluir un cambio en la interfaz a un tema más oscuro, para los momentos en los que la luz sea molesta a los ojos del usuario. El equipo cree que se puede incluir este requisito no funcional de manera sencilla con el trabajo previamente realizado en la interfaz del cliente de escritorio.

Otra funcionalidad interesante fue la propuesta por el usuario final, trabajador de la autoescuela. Ya que se disponía de acceso al alumno mediante el DNI/NIF del alumno, le gustaría poder visualizar un listado de los alumnos registrados según el estado de su expediente en la autoescuela. De esta forma dispondría de una visión general y de rápido acceso de los alumnos activos en la autoescuela. Al equipo nos pareció interesante esta funcionalidad e informamos al cliente para ver si daba el visto bueno para incluirla en la última iteración.

menor prioridad. La autoescuela dispone de un número controlado de profesores en nómina, y por ahora solo planean usar el sistema dentro de la red local, por lo que el alta de partes pierde parte de su atractivo al no poderse hacer directamente desde el móvil en la calle.

3.5 Quinta Iteración

En la quinta iteración se llevan a cabo la últimas historias de usuario, su mayoría son historias no contempladas en el alcance inicial del proyecto, surgidas a partir de la comunicación del cliente durante las anteriores iteraciones. Al ser la última iteración del proyecto, debemos mostrar especial cuidado en asegurar el correcto funcionamiento del sistema y elaborar una documentación que sin ser muy extensa, sirva de referencia para futuros desarrollos que puedan continuar a este proyecto.

3.5.1 Historias de usuario

En las tablas Tabla 40 y Tabla 41 se muestran las historias de usuario de esta iteración.

Tabla 40: H032 Acceso Seguro a la Aplicación

HISTORIA DE USUARIO		Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas
ID H032	Estimación 3	Prioridad Must Have	
Como Administrador	Quiero Acceso Seguro a la Aplicación	Para Que nadie ajeno a mi organización ponga en peligro la integridad de mis datos	
Criterios de satisfacción - Dado que se han introducido credenciales válidas, se obtiene acceso a la aplicación			
Redactado por Alberto Carrillo Ortega	Fecha 20/07/2019	Aprobado por Javier Martínez Cavanillas	Fecha 20/07/2019

Tabla 41: H033 Listar Alumnos por Estado Expediente

HISTORIA DE USUARIO	Nombre de Proyecto	Coordinador de Proyecto
---------------------	--------------------	-------------------------

		TFG Zona L 18/19	Javier Martínez Cavanillas
ID H033	Estimación 3	Prioridad Should Have	
Como Administrador	Quiero Listar Alumnos por Estado Expediente	Para Tener un acceso rápido a los alumnos que están en proceso de formación	
Criterios de satisfacción - Dado que hay alumnos con expedientes validos, se muestran las entradas de esos alumnos agrupadas por en función de los estados de sus expedientes			
Redactado por Alberto Carrillo Ortega	Fecha 20/07/2019	Aprobado por Javier Martínez Cavanillas	Fecha 20/07/2019

3.5.2 Análisis

En las figuras Figura 52 y Figura 53 se muestran los diagramas de actividad referentes a las historias de usuario del apartado anterior.

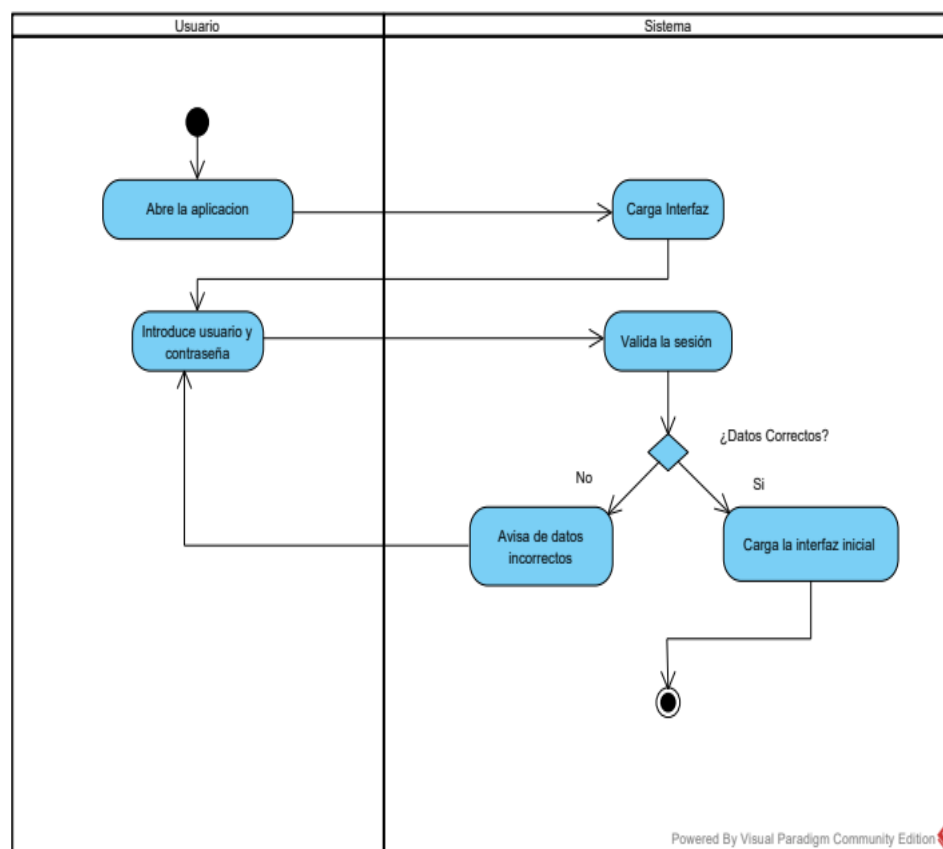


Figura 52: DA032 Acceso Seguro a la Aplicación (H032)

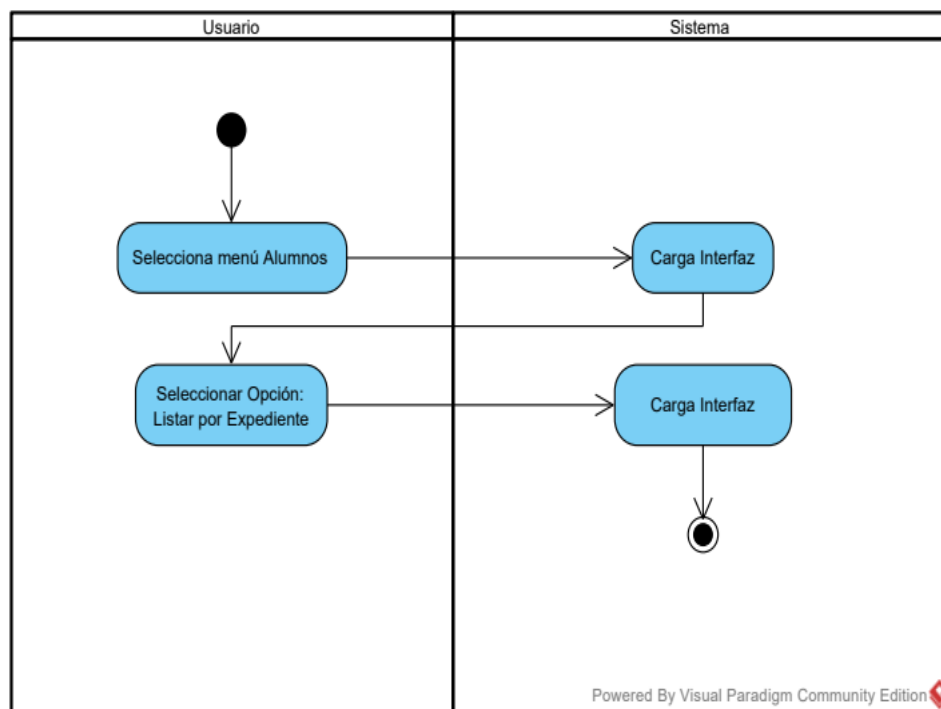


Figura 53: DA033 Listar Alumnos por Estado Expediente (H033)

3.5.3 Planificación de la iteración

Tabla 42: Backlog de Iteración – Iteración 5

BACKLOG DE ITERACIÓN	Nombre de Proyecto TFG Zona L 18/19	Coordinador de Proyecto Javier Martínez Cavanillas
Iteración Iteración 5 – Login, Seguridad y preferencias		Puntos de Historia 13
ID	Tarea	Proyecto
H032 Acceso Seguro a la Aplicación		
T0501	Diseño Login	Servidor
T0502	Diseño Login	Cliente de escritorio
T0503	Diseño Login	Cliente Móvil
T0504	Implementación Login	Servidor
T0505	Implementación Login	Cliente de escritorio
T0506	Implementación Login	Cliente Móvil
T0507	Configuración conexión segura TLS	Servidor
T0508	Configuración conexión segura TLS	Cliente de escritorio

T0509	Configuración conexión segura TLS	Cliente Móvil
H033 Listar Alumnos por Estado Expediente		
T0510	Diseño Listar por Expedientes	Servidor
T0511	Diseño Listar por Expedientes	Cliente de escritorio
T0512	Diseño Listar por Expedientes	Cliente Móvil
T0513	Implementación Listar por Expedientes	Servidor
T0514	Implementación Listar por Expedientes	Cliente de escritorio
T0515	Implementación Listar por Expedientes	Cliente Móvil
Final del proyecto		
T0516	Diseño Preferencias	Cliente de escritorio
T0517	Implementación Preferencias	Cliente de escritorio
T0518	Pruebas de integración	Servidor
T0519	Manual de usuario	Cliente de escritorio
T0520	Manual de usuario	Cliente Móvil
T0521	Repaso Documentación	Servidor
T0522	Repaso Documentación	Cliente de escritorio
T0523	Repaso Documentación	Cliente Móvil

3.5.4 Diseño

En esta iteración nos encargamos de historias de usuario surgidas en su mayor parte durante el proyecto. Los principales esfuerzos de diseño se centrarán en el sistema de inicio de sesión y el listado paginado de alumnos.

3.5.4.1 Gestión de sesiones

Para el sistema de inicio de sesión hemos decidido implementar e integrar un sistema JSON Web Token (JWT) con Spring Security. JWT es un sistema de gestión de sesiones que no necesita mantener el estado en el servidor. Cada vez que un usuario realiza una petición de inicio de sesión, el sistema comprueba las credenciales (usuario y contraseña) y si son correctas, devuelve un token de sesión firmado con la clave privada del servidor. De esta forma, si alguien modificase el

token, el servidor detectaría el cambio y lo descartaría. El token contiene en formato JSON cualquier información necesaria para la gestión de sesiones, como el nombre de usuario, los roles o privilegios y la fecha de expiración. Si alguien consigue hacerse con algún token, solo dispondrá de acceso durante un tiempo, y no dispondrá de las verdaderas credenciales del usuario. Puesto que la gestión de la sesión reside en el propio token, el servidor puede caerse y tras recuperarse, la sesión seguiría siendo válida. El esquema de funcionamiento básico se ilustra en la Figura 54.

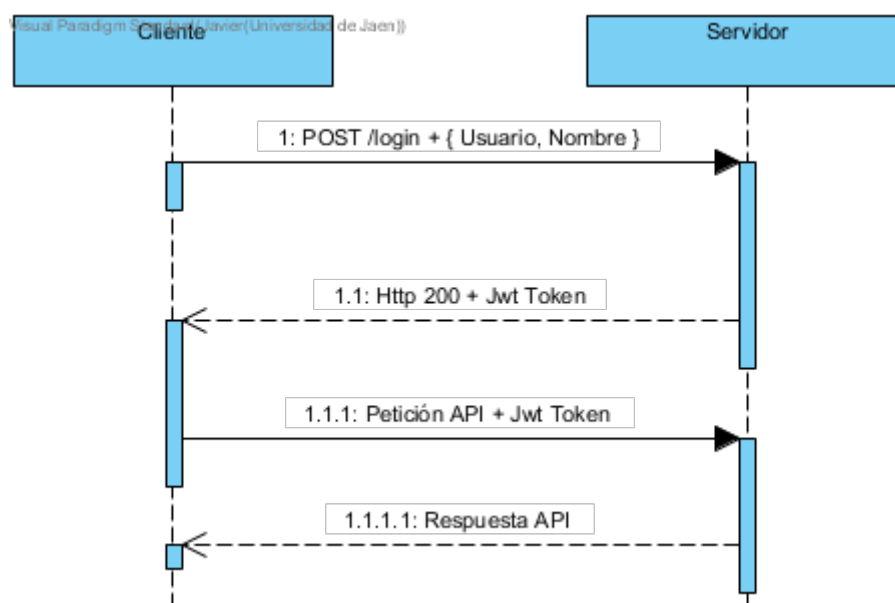


Figura 54: Diagrama de Secuencia - Funcionamiento básico JWT

Un JSON web token se estructura en tres partes, el encabezado o *header*, el contenido o *payload* y la firma o *signature*. En el encabezado se indica el algoritmo hash utilizado para generar la firma, el contenido contiene los llamados *claims* del token, estos son los privilegios, fecha de expiración u otra información personalizada necesaria para la gestión de la sesión. La firma se calcula sobre el resultado de codificar encabezado y contenido en base64url y después concatenar ambas mediante un punto separador. Por último la firma se añade también codificada en base64url al encabezado mediante un punto separador.

La Tabla 43 muestra un ejemplo de la estructura de un token JWT generado por nuestro servidor, *sub* indica el usuario autenticado, *exp* marca la fecha de caducidad del token e *iat* indica la fecha de expedición, son campos estándares.

Además incluimos el campo *scopes*, indicado los roles y privilegios asignados a la sesión.

Tabla 43: JSON Web Token - Ejemplo

JWT Token
<code>eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJhZG1pb3B1cyI6WyJST0x0FETU10I10sImV4cCI6MTU2NjQwMzk2MywiaWF0IjoxNTY2Mzg1OTYzfQ.sUdcmQT3PAn8_fI9dEhzgsMCa5f3Moj0qo0LZQimyyYsU3SqNDuwIs1Gqi6v8PI6InmJPEpHwwg9ZeL30NPh-A</code>
Header
<pre>{ "alg": "HS512" }</pre>
Payload
<pre>{ "sub": "admin", "scopes": ["ROLE_ADMIN"], "exp": 1566403963, "iat": 1566385963 }</pre>

3.5.4.2 Servidor

El diseño del sistema de inicio de sesión está marcado principalmente por las pautas impuestas por Spring Framework, para poder así integrar el servicio con Spring Security. Se han definido dos nuevas entidades en el modelo ORM, usuario y rol. Para personalizar la forma de Spring Security de autorizar las peticiones

implementado dos nuevos proveedores de autorización, (uno para las peticiones AJAX convencionales, utilizadas en el inicio de sesión, y otro para las peticiones JWT, análogamente). Hemos definido dos filtros HTTP para interceptar cada tipo de petición y realizar las operaciones convenientes según el resultado de la autorización. Por último hemos configurado los proveedores y la protección de las diferentes rutas en nuestra API desde una clase de configuración dedicada a Spring Security. El resultado final de este diseño es la integración transparente de la autenticación JWT en los servicios protegidos por Spring Security. La Figura 55 muestra las clases y relaciones resultantes de este diseño.

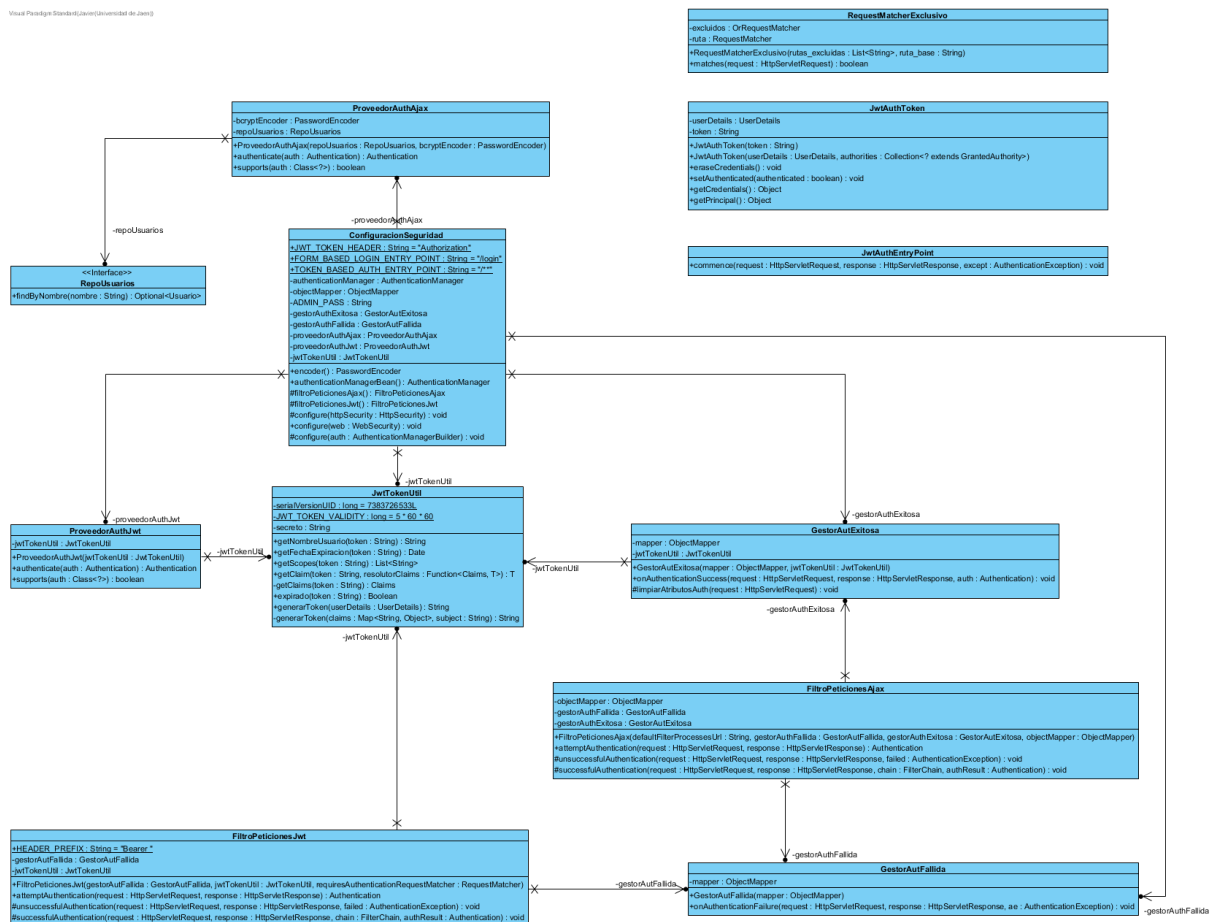


Figura 55: Diagrama de Clases - Autenticación JWT, Servidor

En cuanto al modelo de persistencia, se hace necesario almacenar las credenciales del usuario en el SGBD, por lo que se definen nuevas entidades persistentes dentro de la lógica de negocio para el usuario y los roles de autorización pertinentes (Figura 56), por ahora solo existen dos roles diferenciados, administrador y usuario, el administrador podrá dar de alta nuevos usuarios.

3.5.4.3 *Cliente Móvil*

Por desgracia durante esta iteración enfermé y tuve que reajustar mis planes para el cliente móvil. Descarté la tarea de listar alumnos por estado de expediente y decidí registrar al usuario como administrador por defecto, pues en este punto del desarrollo era imperativo que los clientes tuvieran conexión segura con el servidor, esta conexión solo se obtiene al iniciar sesión como usuario y no tuve tiempo para diseñar las vistas y la lógica de la aplicación necesarias para asegurar un desarrollo completo y satisfactorio del proceso de inicio de sesión.

La lógica es la siguiente: al iniciar la aplicación esta iniciará los parámetros (*ApiLink.CrearContextoSSL()*) necesarios para una conexión segura. Entonces realizará una petición de inicio de sesión (*ApiLink.IniciarSesion()*) al servidor y esperará la respuesta. Si el inicio de sesión se produce con éxito entonces el servidor devolverá un token que la aplicación usará en el resto de peticiones mediante el método (*ApiLink.anadirToken()*) y el interceptor *GestorTokenJwt* (Clase de mi compañero de proyecto) se encargará de añadir este token a las cabeceras de las peticiones.

Si en algún momento algo en este proceso saliese mal, el usuario sería notificado de el error y tras su confirmación la aplicación se cerraría.

En otro orden de cosas se han seguido las recomendaciones del cliente en la primera iteración y se ha mejorado el aspecto visual de las vistas *CrearAlumno*, *EditarAlumno* y *ModificarAlumno*.

En la Figura 57 se muestra el diagrama de clases correspondiente a la aplicación móvil tras el desarrollo de la quinta iteración.

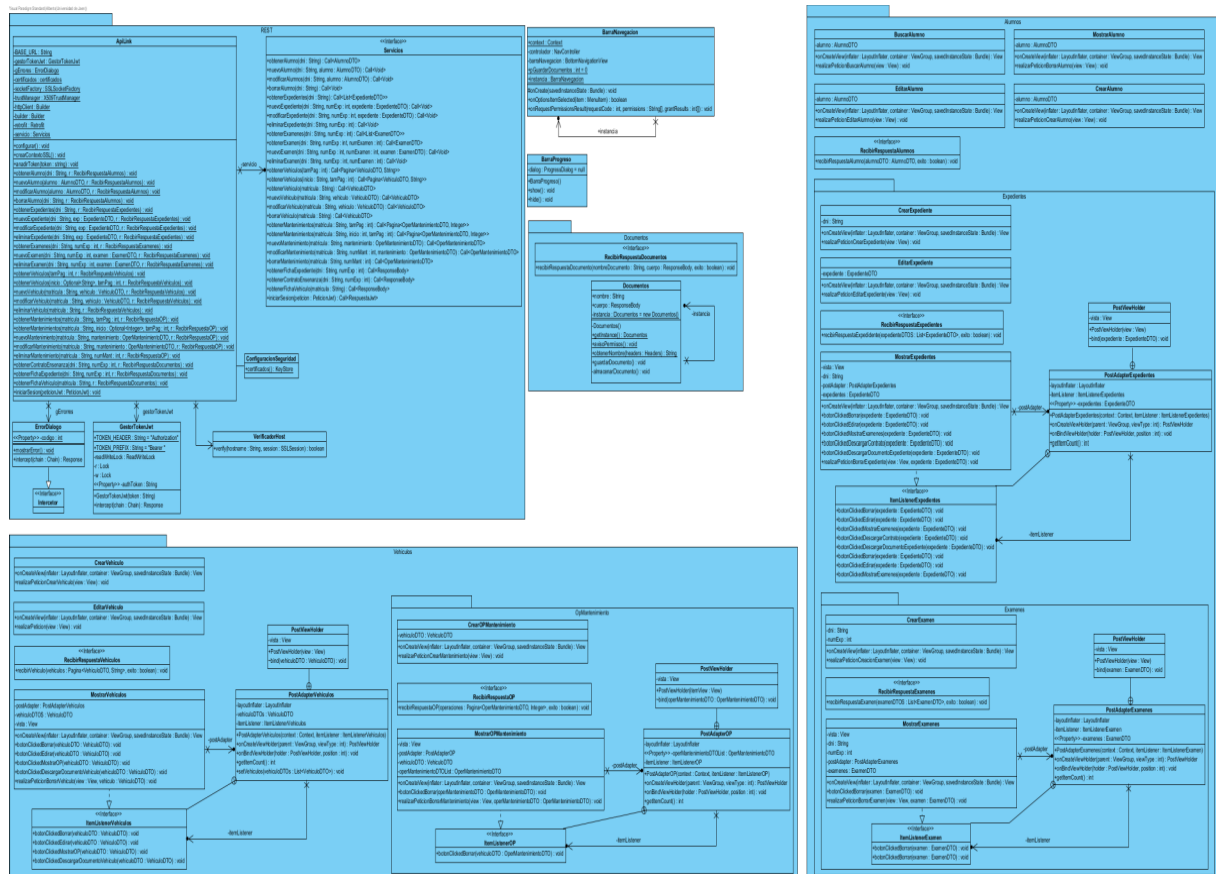


Figura 57: Diagrama de Clases – Iteración 5, Cliente Móvil

Puesto que es la última iteración, adjunto la Figura 58, el grafo de navegación entre las vistas de la aplicación, aunque ha cambiado poco desde la tercera iteración.



Figura 58: Grafo de Navegación – Iteración 5, Cliente Móvil

3.5.5 Resumen de progreso

El transcurso de esta iteración ha sido accidentado. La enfermedad de uno de los miembros del equipo ha supuesto un contratiempo en el desarrollo del proyecto. Se produce el riesgo *Baja temporal de un miembro del equipo*, por lo que se ejecutaría el plan de actuación correspondiente. Sin embargo, el plan de acción correspondiente conlleva el uso de la reserva adicional del presupuesto; al ser este un TFG desarrollado con fines académicos, el trabajo ha sido realizado de forma gratuita para la autoescuela Zona L a cambio de su colaboración con el equipo de desarrollo durante el proyecto. Como en la realidad no se dispone del presupuesto planteado, ha sido imposible suplir el retraso derivado de esta incidencia. El equipo de desarrollo no dispone del tamaño necesario para compensar el esfuerzo

correspondiente, la carga compartida en el desarrollo del servidor recayó en su totalidad en un solo miembro del equipo, que consiguió realizar el trabajo correspondiente al servidor y el cliente de escritorio a tiempo. Por el contrario, el trabajo en la aplicación móvil no pudo ser suplido, ya que el resto del equipo no posee conocimientos necesarios para el desarrollo Android, tras finalizar su enfermedad, el miembro del equipo se reincorpora al proyecto en la segunda semana de trabajo, consigue implementar las historia de usuario de inicio de sesión y conexión segura con el cliente, pero queda sin desarrollar al completo la historia de usuario **H033** faltando por realizar el diseño e implementación correspondientes en el cliente móvil. En las figuras Figura 59, Figura 60 y Figura 61 se muestra el resumen del progreso.

ID	Tarea	Proyecto	Tareas Inicio	29/07/2019 Día 1	30/07/2019 Día 2	31/07/2019 Día 3	1/08/2019 Día 4	2/08/2019 Día 5	5/08/2019 Día 6	6/08/2019 Día 7	7/08/2019 Día 8	8/08/2019 Día 9	9/08/2019 Día 10	Progreso
H022 Acceso Seguro a la Aplicación														
T0501	Diseño Login	Servidor	1	0,5	0,5									100.0%
T0502	Diseño Login	Cliente de escritorio	1			1								100.0%
T0503	Diseño Login	Cliente Móvil	1				1							100.0%
T0504	Implementación Login	Servidor	1				1							100.0%
T0505	Implementación Login	Cliente de escritorio	1					1						100.0%
T0506	Implementación Login	Cliente Móvil	1					1						100.0%
T0507	Configuración conexión segura TLS	Servidor	1											100.0%
T0508	Configuración conexión segura TLS	Cliente de escritorio	1					1						100.0%
T0509	Configuración conexión segura TLS	Cliente Móvil	1						1					100.0%
H023 Listar Alumnos por Estado Expediente														
T0510	Diseño Listar por Expedientes	Servidor	1	0,5				0,5						100.0%
T0511	Diseño Listar por Expedientes	Cliente de escritorio	1			0,5		0,5						100.0%
T0512	Diseño Listar por Expedientes	Cliente Móvil	1			0,5		0,5						100.0%
T0513	Implementación Listar por Expedientes	Servidor	1							1				100.0%
T0514	Implementación Listar por Expedientes	Cliente de escritorio	1								1			100.0%
T0515	Implementación Listar por Expedientes	Cliente Móvil	1									1		100.0%
Final del proyecto														
T0516	Diseño Preferencias	Cliente de escritorio	1		1									100.0%
T0517	Implementación Preferencias	Cliente de escritorio	1				1							100.0%
T0518	Pruebas de integración	Servidor	1									1		100.0%
T0519	Manual de usuario	Cliente de escritorio	1	0,5									0,4	100.0%
T0520	Manual de usuario	Cliente Móvil	1	0,5									0,4	100.0%
T0521	Repaso Documentación	Servidor	1											100.0%
T0522	Repaso Documentación	Cliente de escritorio	1											100.0%
T0523	Repaso Documentación	Cliente Móvil	1											100.0%
Tareas Restantes				23	21,5	19,3	16,3	14,3	11,3	7,8	6,8	5,8	3,8	0
Esfuerzo Ideal				23	21	18	16	14	12	9	7	5	2	0
Puntos de Historia Restantes				13	13	13	13	13	13	5	5	5	0	0
Esfuerzo Real				13	11,7	10,4	9,1	7,8	6,5	5,2	3,9	2,6	1,3	0

Figura 59: Resumen de progreso - Iteración 5

Burndown Chart

Iteración 5

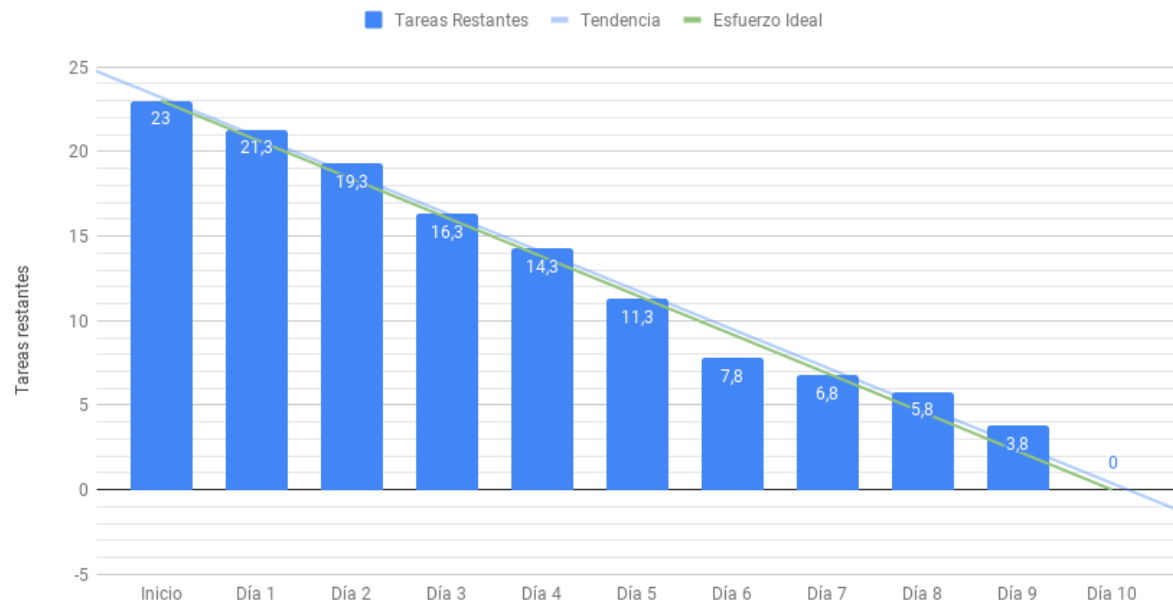


Figura 60: Burndown Chart – Tareas Iteración 5

Burndown Chart

Iteración 5

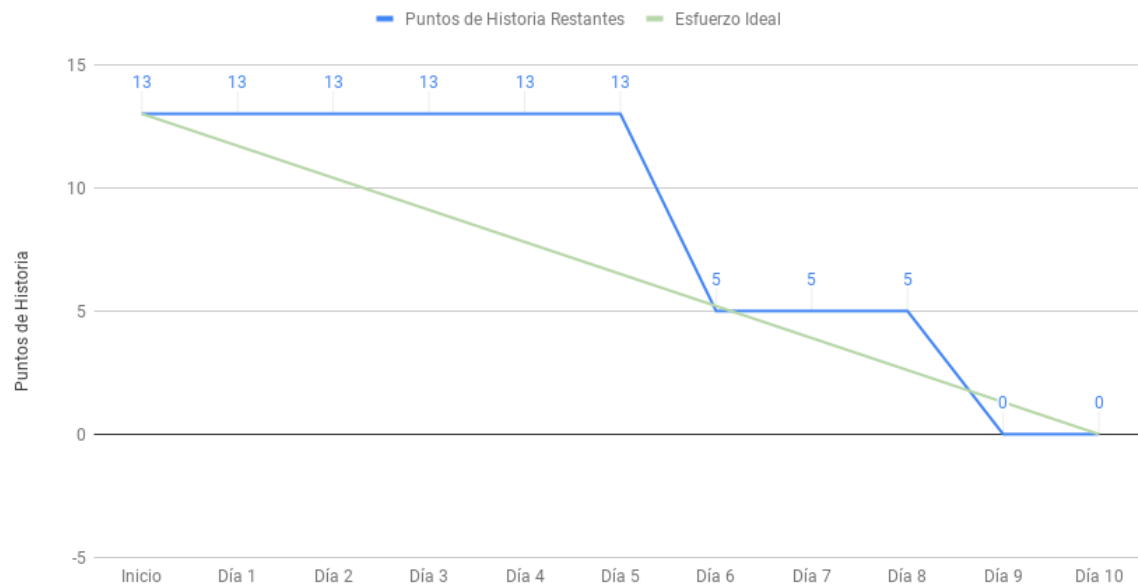


Figura 61: Burndown Chart – Iteración 5

3.5.6 Entrega y reunión final de Iteración

Esta vez las pruebas en la autoescuela se realizan sobre el conjunto total de funcionalidades implementadas en la aplicación para detectar fallos en la integración

de las distintas características. Se descubre un fallo al cambiar entre pantallas en la aplicación de escritorio: en determinadas circunstancias, saltan dos cambios de vista a la vez, uno dentro para cambiar de sección o módulo en el programa y otro dentro de esta misma sección. Esto provoca efectos visuales no deseados a la hora de cambiar de vista. Para solucionarlo, se decide delegar el cambio de vista solamente en la pantalla principal, haciendo que los módulos publiquen un evento de cambio de vista cuando quieran mostrar una nueva pantalla. El cambio se aplicó como resolución de errores antes de la entrega final del proyecto.

El cliente queda contento con la impresión final de la aplicación, la interfaz resulta sencilla para los usuarios en comparación con otras aplicaciones en el mercado probadas con anterioridad.

El proyecto queda finalizado en este punto, una vez visto el resultado obtenido tras 5 iteraciones, el equipo discute con la autoescuela las diferentes posibilidades de cara al futuro. El cliente está interesado en implantar el sistema en la red local de sus oficinas. El sistema es completamente funcional, pero debido a los retrasos ocurridos y analizando el resultado final, acordamos con el cliente que antes de su implantación deberían desarrollarse una serie de modificaciones adicionales necesarias para asegurar una experiencia de uso óptima.

- Elaborar una funcionalidad exclusiva para el administrador que permita registrar usuarios nuevos en la aplicación de forma sencilla. Ya que por ahora los usuarios son almacenados en la base de datos manualmente por el administrador del sistema con la contraseña ya cifrada. El sistema de inicio de sesión se elaboró pensando en ello y ya dispone de un servicio específico preparado para ello, solo se necesita elaborar la correspondiente interfaz en los clientes.
- Mejorar el sistema de notificaciones para mostrar mensajes más claros y sencillos de cara al usuario.
- Implementar la funcionalidad de pagos en los alumnos atrasada. A ser posible a la autoescuela le gustaría llevar un recuento de los pagos de cada alumno. Pudiendo generar un documento con el historial de los pagos realizados.
- En el cliente de escritorio: incluir cambio de tamaño de letra para toda la aplicación en la sección preferencias. De sencilla implementación a priori,

pero necesita un estudio y ajuste en la interfaz para que la relación de aspecto y la disposición de los elementos en pantalla mantengan su forma.

- En el cliente móvil: añadir las funcionalidades que no han podido desarrollarse en esta iteración y personalizar la aplicación con logos de la autoescuela y detalles que mejoren su estética.

4 RESULTADOS Y DISCUSIÓN

4.1 Resumen final del progreso

Tras el desarrollo de las cinco iteraciones, el equipo a logrado desarrollar un total de 24 historias de usuario, que representan un total de 63 puntos de historia. El módulo de gestión de profesores, puesto que no era prioritario para el cliente, se descartó en favor de un buen sistema de seguridad y un mejor control sobre los errores existentes en la aplicación. Además del sistema de sesiones implementado en el última iteración, se implementó el listado de expedientes según el estado del expediente en el cliente de sobremesa, funcionalidad útil para acceder a los alumnos actualmente activos en la autoescuela con mayor facilidad. Finalmente, el proyecto acaba con 5 - 8 puntos de historia de retraso con respecto al plan inicial, en el cual se estimaba una consecución final de 71 puntos de historia.

En la reunión final del equipo realizamos un análisis sobre los motivos del retraso para futuros proyectos o incluso la continuación futura de esta misma aplicación. Consideramos las principales causas del retraso los siguientes factores:

- **Aprendizaje de nuevas tecnologías:** Aunque el equipo poseía experiencia con la mayoría de herramientas utilizadas, otras como JavaFX para la elaboración de interfaces gráficas en Java o el desarrollo de aplicaciones en Android han sido completamente nuevas para el equipo de desarrollo. A esto se le suman otros requisitos como la gestión de documentos, que nos ha llevado a familiarizarnos con herramientas PDF desconocidas también por el equipo hasta el momento. Todo ello genera un esfuerzo extra no solo al principio de la iteración, donde se debe elaborar un diseño compatible y consciente de las capacidades y metodología de tecnologías que utilizan, sino en la refactorización posterior, puesto que a medida que el proyecto avanza se van descubriendo características y posibilidades que proporcionan una manera más sencilla de desarrollar, o malas prácticas presentes en el código anteriormente escrito y que por lo tanto, debe ser corregido y refactorizado para mantener el código legible y ordenado.

- **Inicio del proyecto:** El inicio del proyecto supuso un retraso mayor del esperado. Se cometió un error al desestimarlos en la planificación inicial del proyecto, pensando que podía ser incluido en la primera iteración sin producir un retraso significativo. Esto sería así contando solo con la instalación y configuración del entorno de desarrollo. Sin embargo, el diseño de la estructura general del sistema y la consiguiente creación de proyectos para JavaFX y Android trajo problemas de configuración adicionales específicos que no se encuentran presentes en proyectos Java sin interfaz gráfica como el servidor.

En la Figura 62 y Figura 63 se muestra la evolución del *Product Backlog* y del *Burndown Chart* a lo largo del proyecto.

ID	Tarea	Esfuerzo estimado	Inicio	Iteración 1	Iteración 2	Iteración 3	Iteración 4	Iteración 5	Progreso
H001	Añadir Alumno	3	3	3					100,0%
H002	Modificar Alumno	3	3	2	1				100,0%
H003	Borrar Alumno	1	1	1					100,0%
H004	Busqueda por DNI	1	1	1					100,0%
H005	Añadir expediente	3	3		3				100,0%
H006	Borrar expediente	1	1		1				100,0%
H007	Modificar expediente	3	3		3				100,0%
H008	Añadir examen	2	2		2				100,0%
H009	Borrar examen	1	1		1				100,0%
H010	Listar Expedientes de un alumno	2	2		2				100,0%
H011	Listar Exámenes de un expediente	2	2			2			100,0%
H012	Añadir vehiculo	3	3			3			100,0%
H013	Modificar vehiculo	3	3			3			100,0%
H014	Borrar vehiculo	1	1			1			100,0%
H015	Añadir Operacion de mantenimiento	1	1			1			100,0%
H016	Borrar Operacion de mantenimiento	1	1			1			100,0%
H017	Listar Vehiculos	2	2			2			100,0%
H018	Listar Operaciones de Mantenimiento	2	2			2			100,0%
H019	PDF contrato	5	5				5		100,0%
H020	PDF expediente	5	5				5		100,0%
H021	PDF vehiculo	5	5				5		100,0%
H022	Añadir Profesor	3	3				3		100,0%
H023	Borrar Profesor	1	1				1		100,0%
H024	Modificar Profesor	3	3				3		100,0%
H025	Añadir parte	3	3				3		100,0%
H026	Borrar parte	1	1				1		100,0%
H027	Modificar parte	3	3				3		100,0%
H028	Buscar partes	2	2				2		100,0%
H029	Añadir pago	3	3						0,0%
H030	Borrar pago	1	1						0,0%
H031	Mostrar pagos	1	1						0,0%
H032	Login	8			-8			8	100,0%
H033	Listar Expedientes según su estado	5					-5	5	100,0%
Esfuerzo restante			71	64	59	44	18	5	
Esfuerzo Ideal			71	57	43	28	14	0	

Leyenda

- Historia no inicial
- Valor añadido
- Historia descartada

Figura 62: Resumen de progreso – Product backlog

Burndown Chart

Autoescuela Zona L TFG 18/19

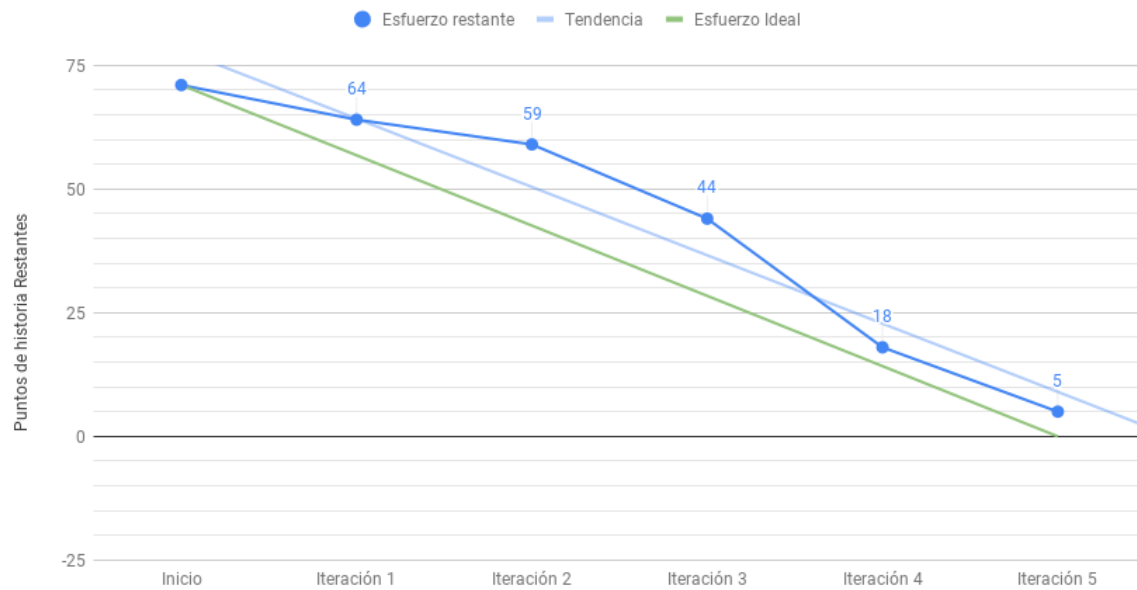


Figura 63: Burndown Chart – Final de proyecto

4.2 Entrega

Los distintos documentos y recursos resultantes del proyecto se presentarán junto a la memoria en un archivo ZIP con la siguiente estructura:

- Subdirectorio *Ejecutables*: Contiene los ejecutables finales del servidor y del cliente sobremesa. En caso del cliente móvil se proporcionará el archivo APK para su instalación en dispositivos Android.
- Subdirectorio *Documentos*: Contiene los documentos proporcionados por la autoescuela para ilustrar el desarrollo de sus actividades diarias y que se emplearon durante el análisis.
- Subdirectorio *Código Fuente*: Contiene el código fuente de los respectivos proyectos de las aplicaciones desarrolladas como parte del sistema.

5 CONCLUSIONES

En retrospectiva, el desarrollo del proyecto ha sido duro pero hemos adquirido muchas nuevas habilidades, competencias y experiencias que nos serán útiles en nuestro futuro laboral. Lo que al principio supuso un retraso en el desarrollo y fuente de estrés, ahora es experiencia en tecnologías punteras y de uso masivo por todo el mundo. Como resultado, consideramos el producto final del proyecto una aplicación que cumple con las expectativas que nos presentaron al principio en la *Autoescuela Zona L*.

Actualmente seguimos en contacto con la autoescuela y existe la posibilidad tras entregar este Trabajo de Fin de Grado de continuar con el proyecto, ahora sin las restricciones impuestas por la Universidad. La gestión de profesor y partes sería fácilmente integrable con el contenido actual del proyecto y daría mayor sentido a el desarrollo de la aplicación móvil. En estos momentos la autoescuela está en proceso de presentar la documentación de los alumnos a la DGT de manera telemática. Este sistema todavía está en fase de pruebas pero una vez que se implante se podría presentar una petición de convenio para acceder directamente a la API del sistema de la DGT mediante nuestra aplicación. Si la autoescuela estuviera dispuesta a contratarnos el proyecto tiene margen para crecer hasta poder rivalizar en el mercado.

En conclusión creemos que ha sido una buena experiencia pero si hubiésemos podido trabajar con tecnologías web y no dividirnos en dos clientes, la aplicación resultante hubiese sido más completa. En otras palabras, nos hemos dado cuenta de la gran importancia de la elección de tecnologías a la hora de plantear un proyecto. Las aplicaciones nativas poseen ciertas ventajas frente a las aplicaciones web; son más ligeras, consumen menos recursos del sistema y se integran mejor con el entorno proporcionado por el sistema operativo. Pero las tecnologías web resuelven un sinfín de problemas a la hora del desarrollo: Conexión con la red, diseño de vistas, separación Modelo-Vista-Controlador, gestión de hilos y concurrencia, etc. Es labor del equipo sopesar las ventajas que proporciona una u otra opción frente a los inconvenientes que acarrearán.

6 ANEXOS

6.1 A1 – Diagramas y Grafos de navegación

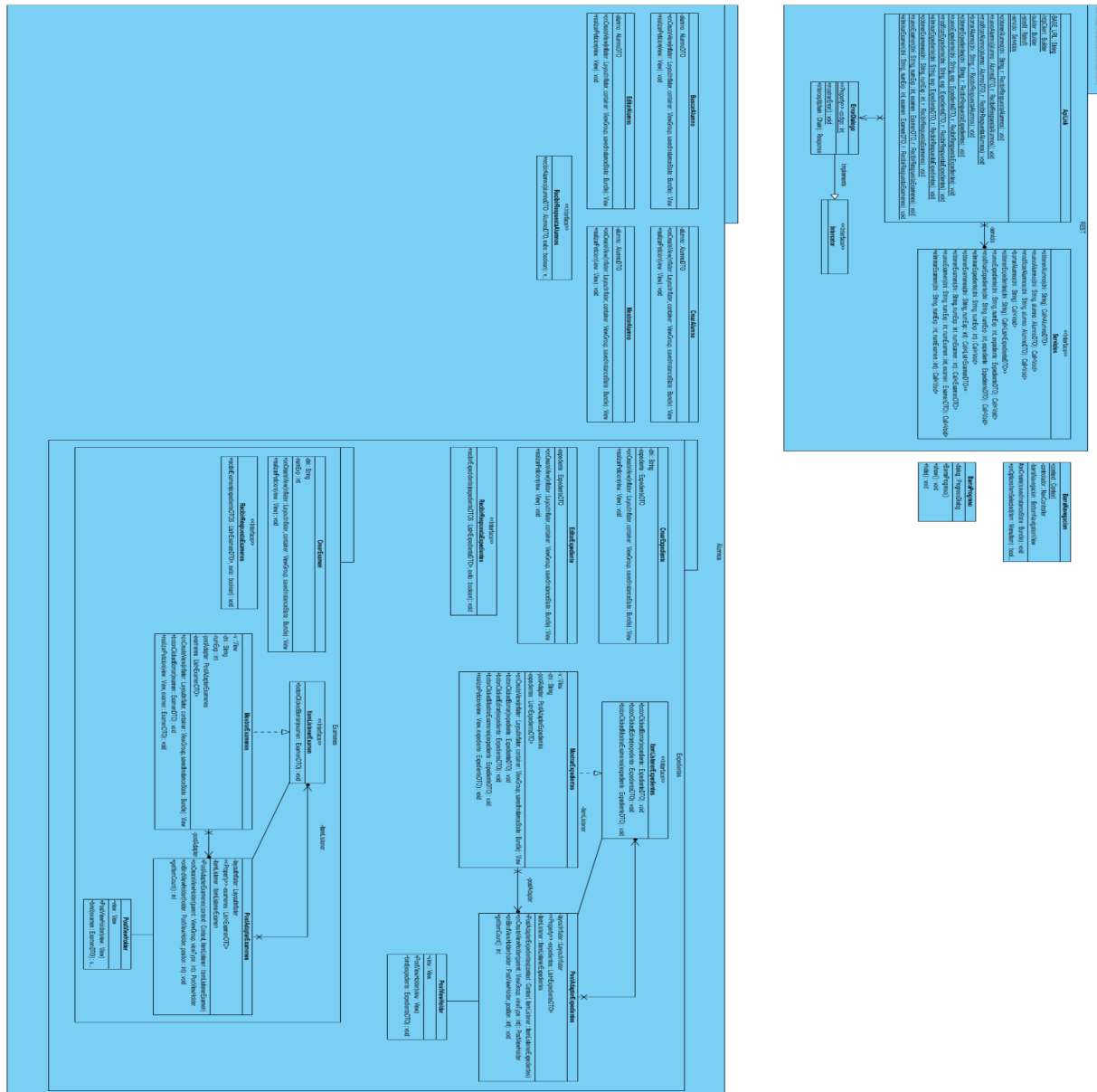


Figura 64: Diagrama de Clases – Iteración 1, Cliente Móvil

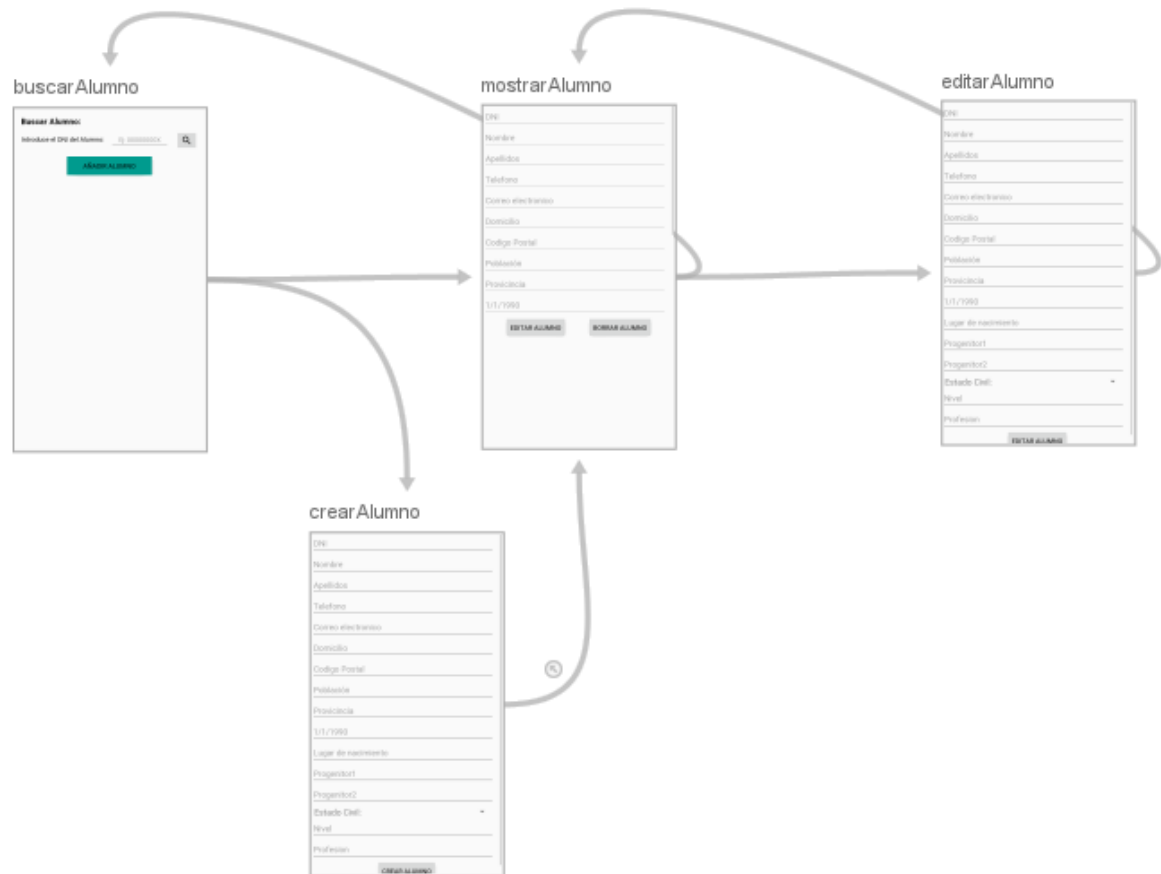


Figura 65: Grafo Navegación – Iteración 1, Cliente Móvil

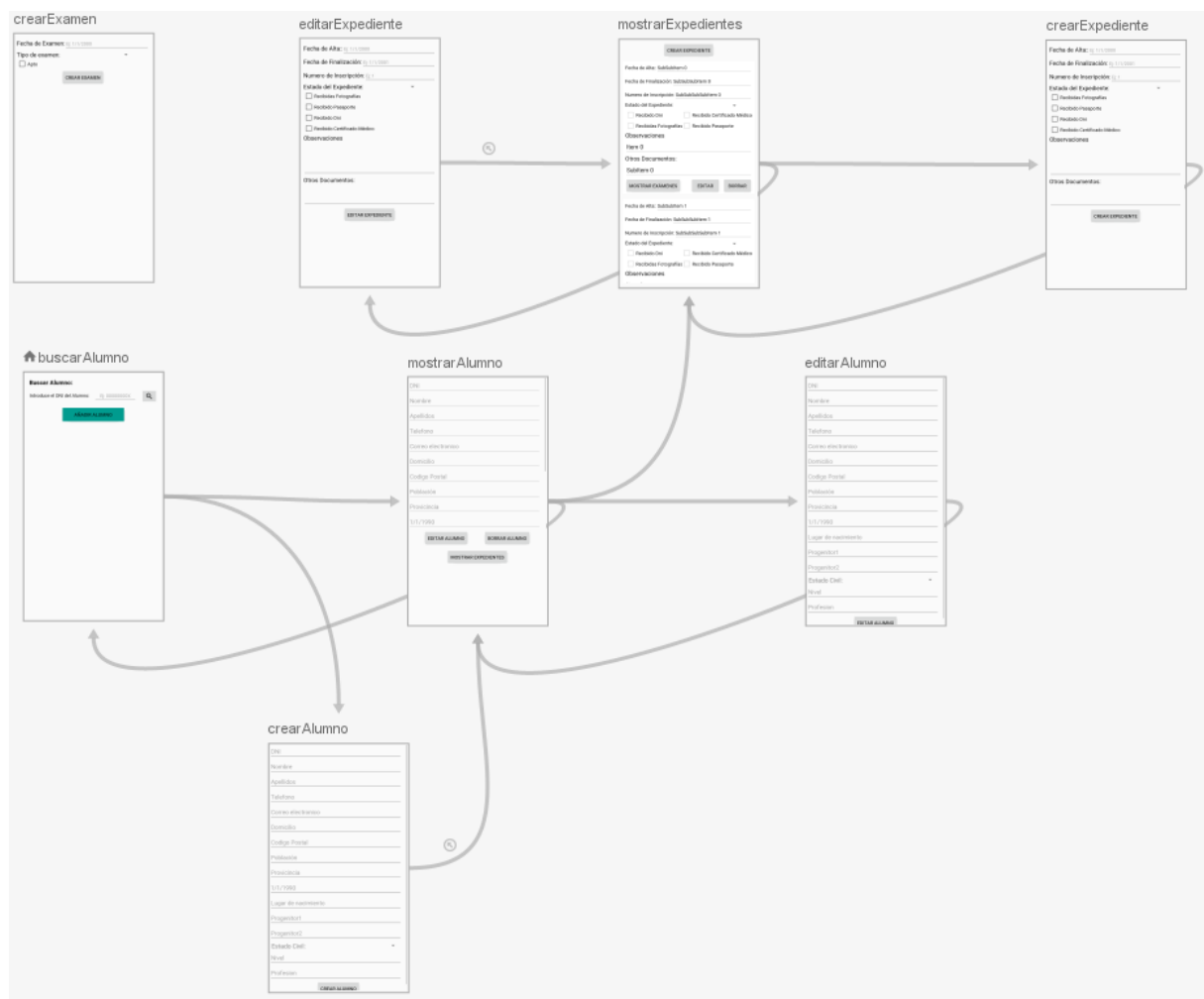


Figura 66: Cliente Móvil: Grafo Navegación 2ª Iteración

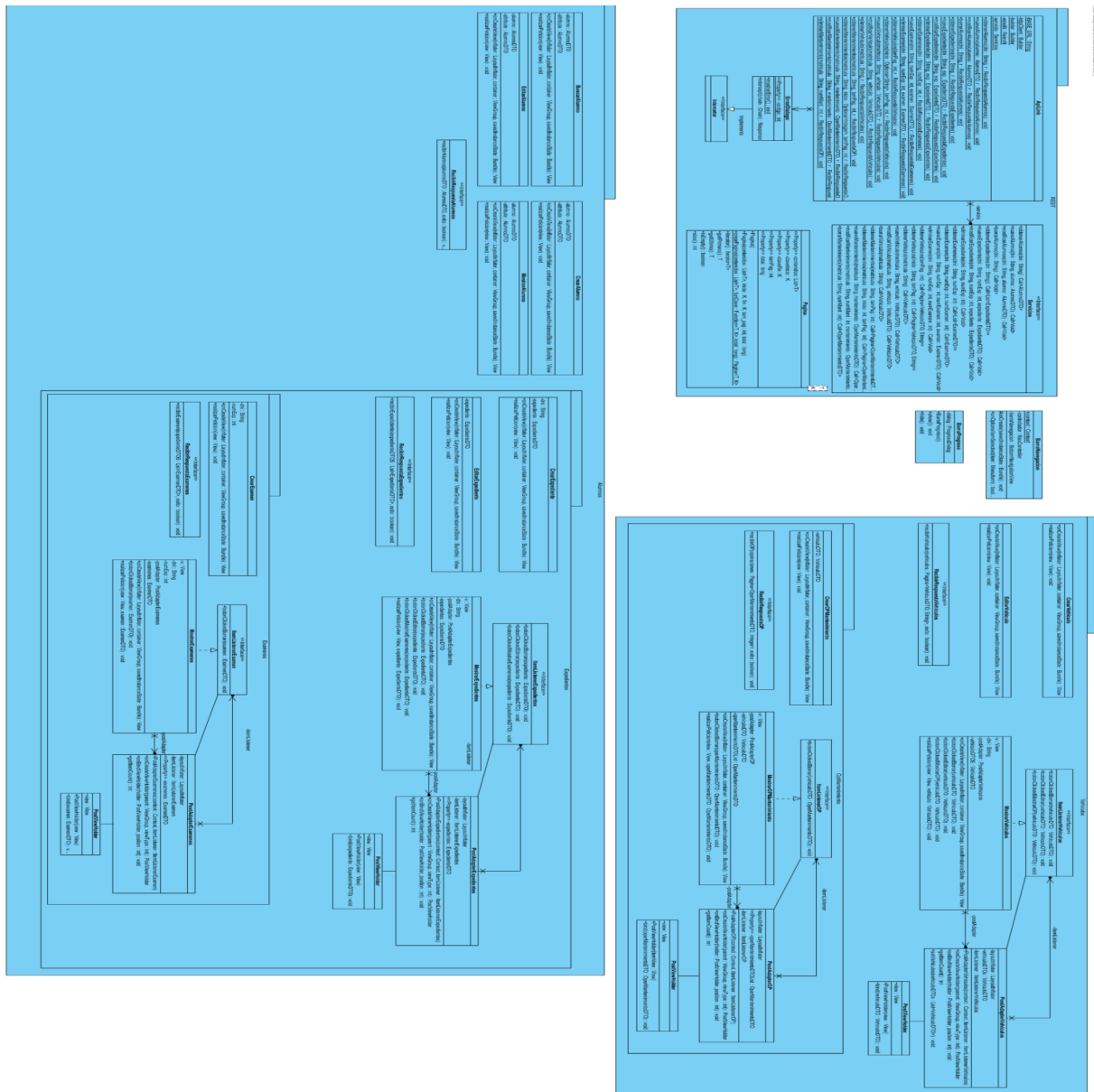


Figura 67: Diagrama de Clases – Iteración 3, Cliente Móvil

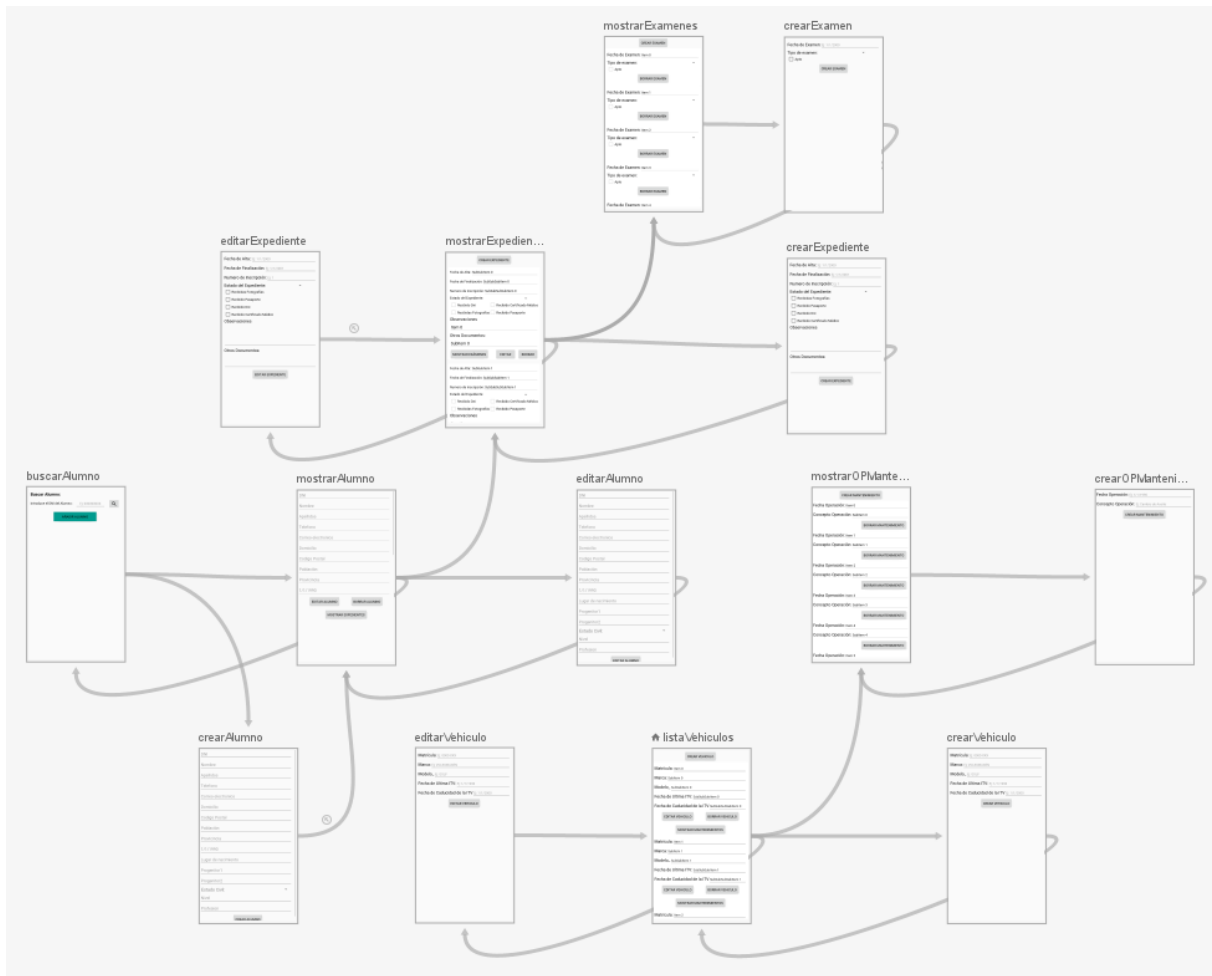


Figura 68: Grafo Navegación - 3ª Iteración, Cliente Móvil

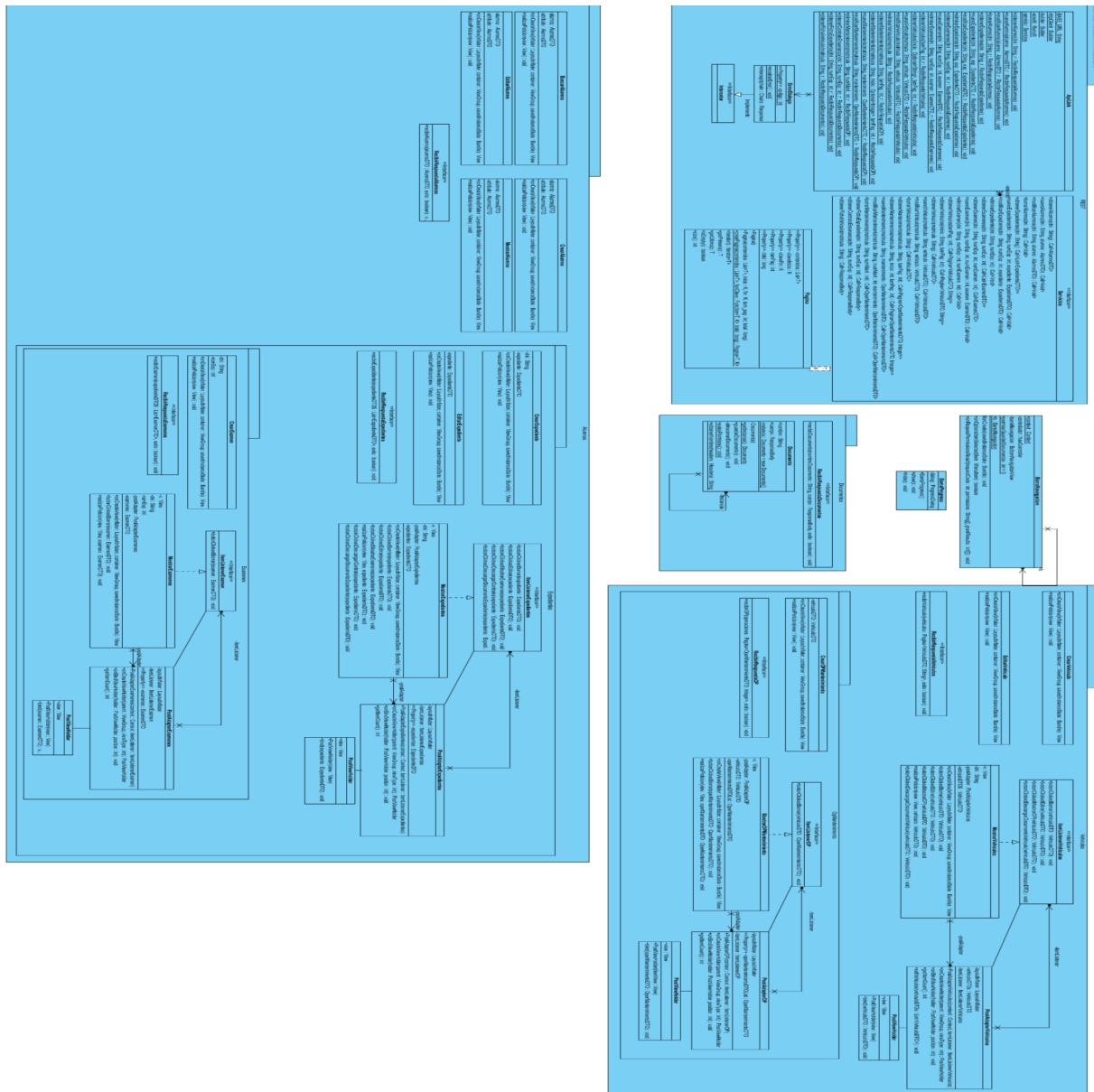


Figura 69: Diagrama de Clases – Iteración 4, Cliente Móvil

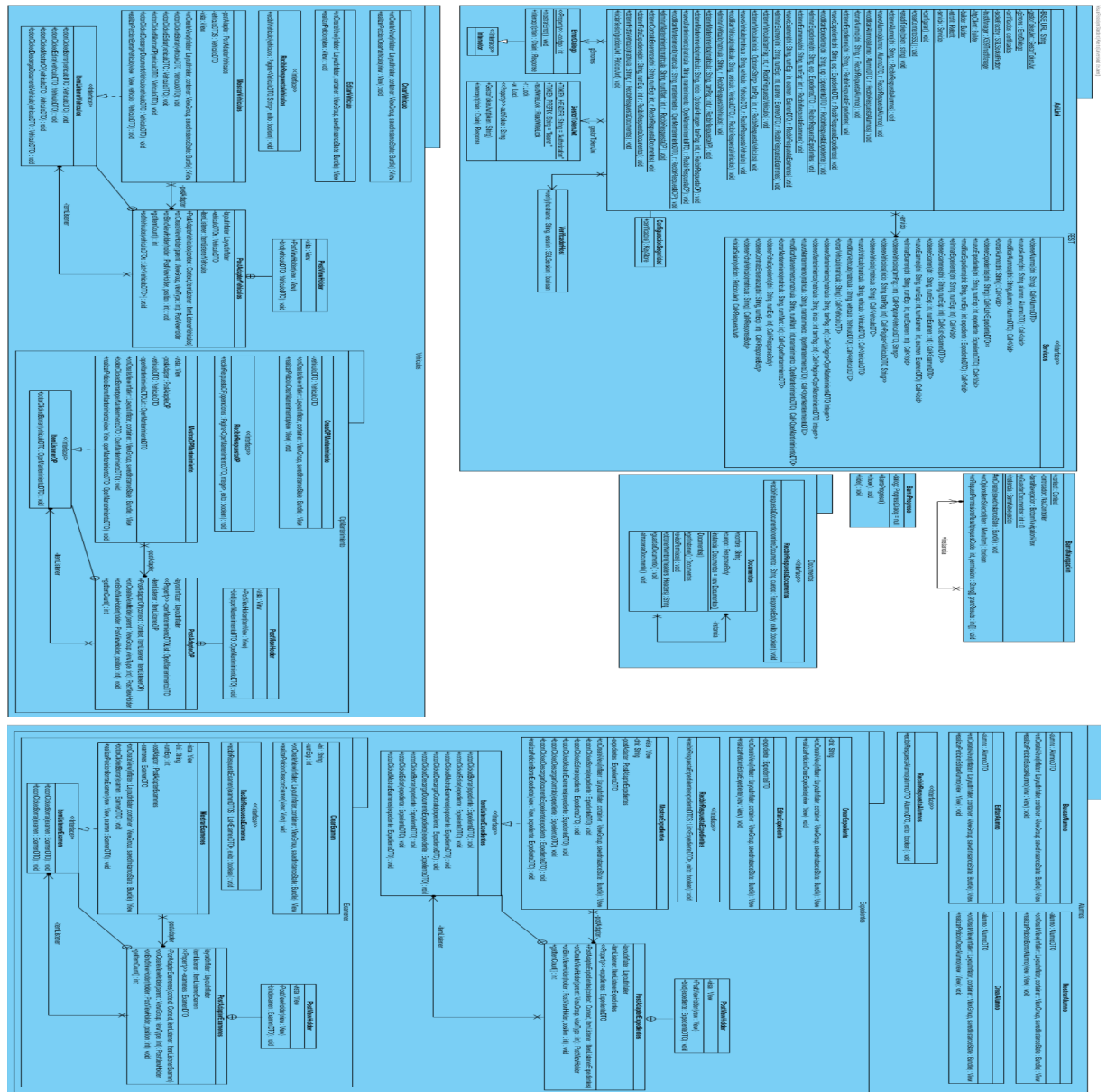


Figura 70: Diagrama de Clases – Iteración 5, Cliente Móvil



Figura 71: Grafo de Navegación – Iteración 5, Cliente Móvil

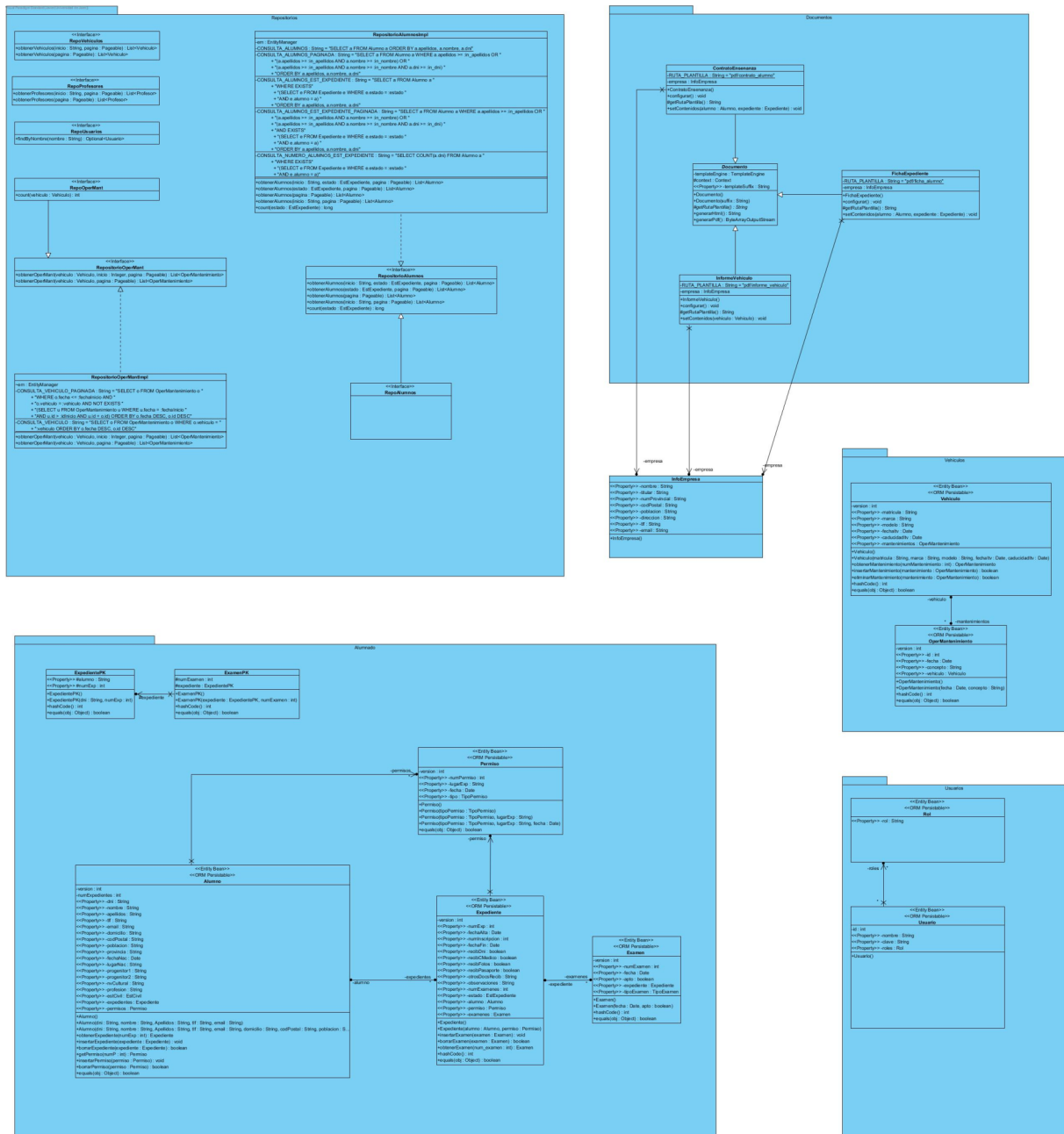


Figura 72: Diagrama de Clases – Lógica de Neocio, Servidor

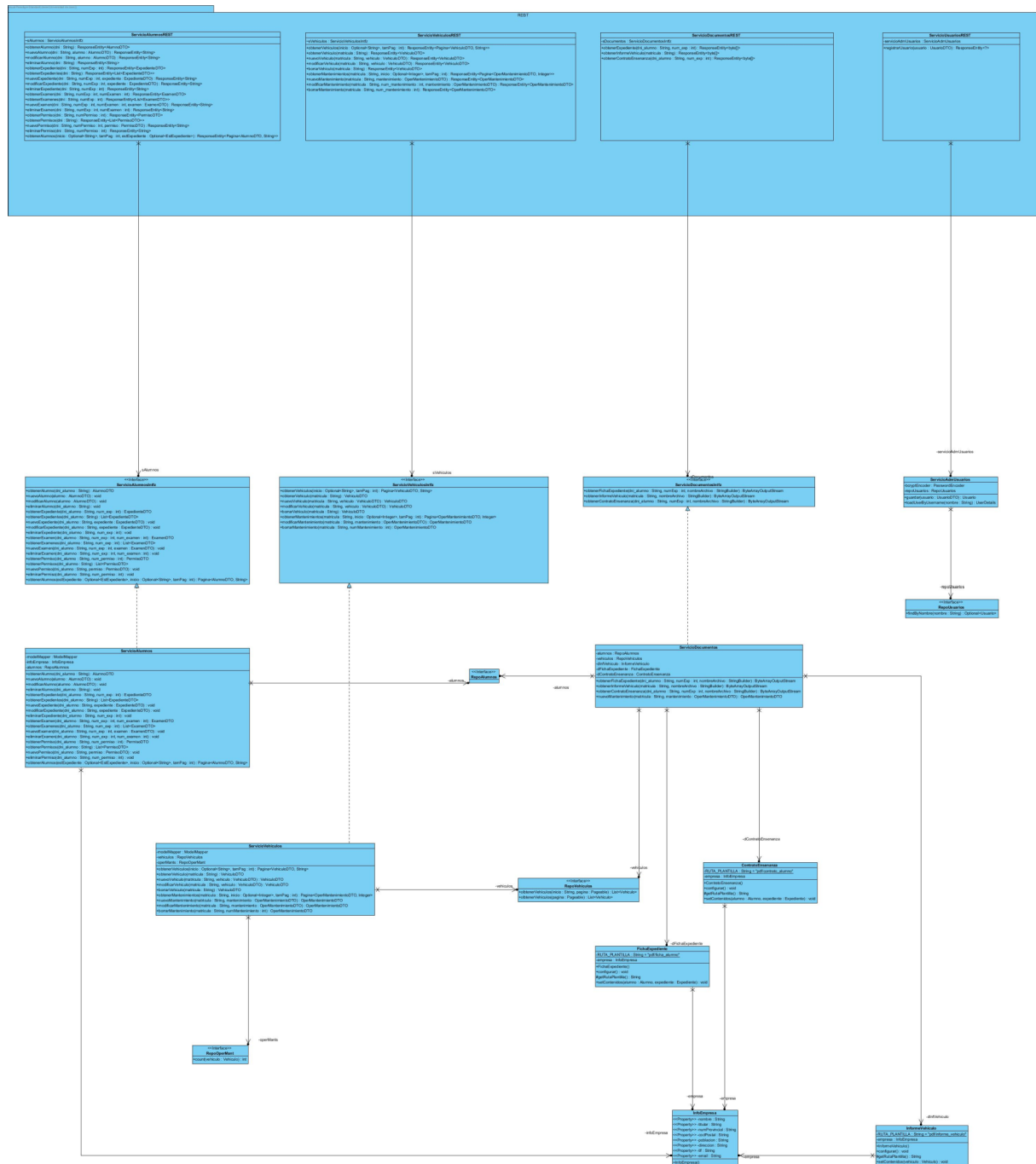


Figura 73: Diagrama de Clases – Capa de Servicios ,Servidor

6.2 A2 – Documentación relevante del proyecto

CONTRATO DE ENSEÑANZA

En Jaén, a de de se reúnen, de una parte, D. Juan Robles Gutierrez, Titular/Director de la Escuela Particular de Conductores denominada **Zona L**, con número provincial J-0285-01-DG8, con domicilio en Jaén, Avda. de Madrid nº 37, y de otra D/Dª con D.N.I. y domicilio en calle C.P., ambos con capacidad legal para contratar y obligarse, convienen el presente Contrato de Aprendizaje para la obtención del Permiso de Conducción de la clase y de acuerdo con el régimen de enseñanza establecido en el capítulo 6º del Reglamento de Escuelas Particulares de Conductores de Vehículos a Motor, aprobado por el Real Decreto 1295/2003, de fecha 17 de Octubre.

CONDICIONES GENERALES

1.- La escuela impartirá las enseñanzas necesarias para la obtención del Permiso de conducir reseñado, con la máxima dedicación y eficacia, y con el profesorado que cumpla la totalidad de los requisitos previstos en el Reglamento de Escuelas Particulares de Conductores.

2.- La duración de cada clase teórica será de 45 minutos, la de cada clase práctica de 45 minutos. En el caso de ofertar clases prácticas gratuitas, la duración de cada una de ellas será de 30 minutos y se realizarán en la última fase de la enseñanza práctica.

3.- Para poder ser presentado a examen será necesaria una certificación conjunta del Director y del Profesor que haya impartido la enseñanza de la conducción en la que se exprese que a su juicio, el alumno ha recibido las enseñanzas necesarias para obtener el Permiso de Conducción. Si el alumno insistiera a ser presentado a examen sin este requisito, el Director de la Escuela lo hará constar expresamente, resolviendo en cada caso la Jefatura Provincial de Tráfico, según proceda.

4.- Salvo en casos debidamente justificados y autorizados por la Jefatura Provincial de Tráfico, el alumno será acompañado durante la prueba de Circulación por el Profesor que le haya impartido la enseñanza práctica, que se responsabilizará del doble mando.

5.- La Escuela velará por el cumplimiento de todo lo dispuesto en la legislación vigente, así como del estricto cumplimiento de las obligaciones de profesorado, atendiendo cualquier reclamación o queja del alumno en relación con la enseñanza recibida, a cuyo efecto, en el caso de quedar disconforme el alumno dispone de un Libro de Reclamaciones en el que podrá manifestar sus quejas para conocimiento de la Jefatura Provincial de Tráfico.

CONDICIONES PARTICULARES

MATRÍCULA.....	€	CLASES PRÁCTICAS B.....	€
INSCRIPCIÓN	€	CLASES PRÁCTICAS A. CC.....	€
CURSOS INTENSIVOS	€	CLASES PRÁCTICAS A. CA	€
TASAS TRÁFICO	€	CLASES PRÁCTICAS CICLOMOTOR	€
RENOVACIÓN EXPEDIENTE	€	CLASES DE EXAMEN	€

Todos los precios con IVA incluido.

6.- El alumno debe avisar, al menos con VEINTICUATRO HORAS su falta de asistencia a la clase práctica. En caso de no hacerlo, la clase se contará como no realizada.

7.- Si el alumno interrumpe las clases prácticas durante dos días consecutivos, pierde su derecho a la hora reservada, pudiendo disponer la Dirección de esta hora para otro alumno.

8.- Las fechas de convocatorias a examen del alumno serán fijadas por la Autoescuela en función a la capacidad de examen de la Jefatura Provincial de Tráfico y de la Autoescuela.

9.- La Autoescuela se reserva el derecho de nuevas convocatorias a alumnos repetidores en función del artículo anterior.

10.- Si el alumno desea cambio de Profesor o Vehículo, deberá notificarlo al Director, quien resolverá al respecto.

11.- La Autoescuela se reserva el derecho de cambiar de Profesor o Vehículo por razones de fuerza mayor.

12.- El Alumno deberá estar al corriente de Pago antes de la presentación de solicitud de examen ante la Jefatura Provincial de Tráfico.

13.- Este Contrato tendrá una vigencia de 3 meses.

Y de prueba de conformidad, ambas partes firman por duplicado, el presente Contrato, cuyo original queda depositado en la Escuela, entregándose el duplicado al alumno, en la fecha y lugar al principio reseñados.

En Jaén, a de de

El alumno (o su representante) Por la Escuela

Figura 74: Formularios Autoescuela – Contrato de enseñanza

 Tel: 957 372 634

EXPEDIENTE DEL ALUMNO

EXPEDIENTE N.º: -
 N.º INSCRIPCIÓN: -
 FECHA DE FINALIZACIÓN: -
 PERMISO QUE SOLICITA: -
 PERMISO QUE POSEE: -
 EXPEDIDO EN: -
 FECHA INICIO: -
 FECHA: -
 TELEF. - FAX -
 E-mail: -
 FOTO

APELLIDOS: - DOMICILIO: - NOMBRE: -
 D.N.I. N.º: - C.P.: -
 POBLACIÓN: -
 PROVINCIA: -
 NACIDO/A EL: - EN: -
 HIJO/A DE: - Y - ESTADO CIVIL: -
 NIVEL CULTURAL: - PROFESIÓN: -
 PROFESOR TEORICA - PROFESOR PRACTICA - VEHICULO MATRICULA

DOCUMENTACION RECIBIDA
☐ D.N.I. (Fotocopia)
☐ Cert. médico/psicotécnico
☐ Fotografías
☐ Pasaporte
☐ R. Conducir (Fotocopia)
 OTROS DOCUMENTOS

E		X		A		M		E		N		E		S		OBSERVACIONES	
TEORICO						MANIOBRAS						CIRCULACION					
FECHA	APTO	NO APTO	FECHA	APTO	NO APTO	FECHA	APTO	NO APTO	FECHA	APTO	NO APTO	FECHA	APTO	NO APTO	FECHA	APTO	NO APTO

Figura 75: Formulario Autoescuela – Expediente del alumno

GESTION DE CONTABILIDAD

FECHA	N.º	CONCEPTO		INGRESOS	I.V.A.	SALDO
ANTICIPOS						
1-1-17			TOTAL			
			VºBº CAJA			
			COLABORADOR			
						
						
						
						
						
			TOTAL Gastos			
			Beneficio			
			A cobrar			
			Anticipos			
			SALDO a favor			

Handwritten notes in the form:

- 1-1-17
- PERSONAS
- tasas B.
- D -
- 146 -

Figura 76: Formulario Autoescuela – Contabilidad alumno



EDICIONES
MAITER

PARTE DIARIO DE CLASES

Autoescuela: Sección:

Profesor:

Vehículo: Matrícula: Clase de Permiso:

Prueba de destreza ☐ Prueba de circulación ☐

[illegible]

LECTURA KMS.		
Anterior	Recorrido	Total

de de 20.

Firma del Profesor,

Figura 77: Formulario Autoescuela – Parte diario de clases

Zona L

Dirección: Av. de Madrid, n.º 37 23008 Jaén
Teléfono de contacto: +34 953 22 47 52 / +34 662 09 87 62
Correo electrónico: zona.l.autoescuela@gmail.com



Contrato de enseñanza

En Jaén, a 02 - 09 - 2019 se reúnen, de una parte, D. Juan Robles Gutierrez, Titular/Director de la Escuela Particular de Conductores denominada Zona L, con número provincial J-0285-01-DG8, con domicilio en Jaén, Av. de Madrid, n.º 37, y de otra D/Dª GEOFFREY BALL, con D.N.I. Z1320392J y domicilio en JAÉN, calle AV DE MADRID, 17, 23003 JAÉN, ESPAÑA C. P. 23003, ambos con capacidad legal para contratar y obligarse, convienen el presente Contrato de Aprendizaje para la obtención del Permiso de Conducción de la clase C1 y de acuerdo con el régimen de enseñanza establecido en el capítulo 6º del Reglamento de Escuelas Particulares de Conductores de Vehículos a Motor, aprobado por el Real Decreto 1295/2003, de fecha 17 de Octubre.

Condiciones Generales

- 1.- La escuela impartirá las enseñanzas necesarias para la obtención del Permiso de conducir reseñado, con la máxima dedicación y eficacia, y con el profesorado que cumpla la totalidad de los requisitos previstos en el Reglamento de Escuelas Particulares de Conductores.
- 2.- La duración de cada clase teórica será de 45 minutos, la de cada clase práctica de 45 minutos. En el caso de ofertar clases prácticas gratuitas, la duración de cada una de ellas será de 30 minutos y se realizarán en la última fase de la enseñanza práctica.
- 3.- Para poder ser presentado a examen será necesaria una certificación conjunta del Director y del Profesor que haya impartido la enseñanza de la conducción en la que se exprese que a su juicio, el alumno ha recibido las enseñanzas necesarias para obtener el Permiso de Conducción. Si el alumno insistiera a ser presentado a examen sin este requisito, el Director de la Escuela lo hará constar expresamente, resolviendo en cada caso la Jefatura Provincial de Tráfico, según proceda.
- 4.- Salvo en casos debidamente justificados y autorizados por la Jefatura Provincial de Tráfico, el alumno será acompañado durante la prueba de Circulación por el Profesor que le haya impartido la enseñanza práctica, que se responsabilizará del doble mando.
- 5.- La Escuela velará por el cumplimiento de todo lo dispuesto en la legislación vigente, así como del estricto cumplimiento de las obligaciones del profesorado, atendiendo cualquier reclamación o queja del alumno en relación con la enseñanza recibida, a cuyo efecto, en el caso de quedar disconforme el alumno dispone de un Libro de Reclamaciones en el que podrá manifestar sus quejas para conocimiento de la Jefatura Provincial de Tráfico.

Condiciones Particulares

MATRÍCULA	€ CLASES PRÁCTICAS B	€
INSCRIPCIÓN	€ CLASES PRÁCTICAS A. CC	€
CURSOS INTENSIVOS	€ CLASES PRÁCTICAS A. CA	€
TASAS TRÁFICO	€ CLASES PRÁCTICAS CICLOMOTOR	€
RENOVACIÓN EXPEDIENTE	€ CLASES DE EXAMEN	€

- 6.- El alumno debe avisar, al menos con VEINTICUATRO HORAS su falta de asistencia a la clase práctica. En caso de no hacerlo, la clase se contará como no realizada.
- 7.- Si el alumno interrumpe las clases prácticas durante dos días consecutivos, pierde su derecho a la hora reservada, pudiendo disponer la Dirección de esta hora para otro alumno.
- 8.- Las fechas de convocatorias a examen del alumno serán fijadas por la Autoescuela en función a la capacidad de examen de la Jefatura Provincial de Tráfico y de la Autoescuela.
- 9.- La Autoescuela se reserva el derecho de nuevas convocatorias a alumnos repetidores en función del artículo anterior.
- 10.- Si el alumno desea cambio de Profesor o Vehículo, deberá notificarlo al Director, quien resolverá al respecto.
- 11.- La Autoescuela se reserva el derecho de cambiar de Profesor o Vehículo por razones de fuerza mayor.
- 12.- El Alumno deberá estar al corriente de Pago antes de la presentación de solicitud de examen ante la Jefatura Provincial de Tráfico.
- 13.- Este Contrato tendrá una vigencia de 3 meses.

Y de prueba de conformidad, ambas partes firman por duplicado, el presente Contrato, cuyo original queda depositado en la Escuela, entregándose el duplicado al alumno, en la fecha y lugar al principio reseñados.
 En Jaén, a 02 - 09 - 2019.

El alumno, (o su representante)

La escuela

Zona L**Dirección:** Av. de Madrid, n.º 37 23008 Jaén**Teléfono de contacto:** +34 953 22 47 52 / +34 662 09 87 62**Correo electrónico:** zona.l.autoescuela@gmail.com**Información del Alumno: GEOFFREY BALL****DNI:** Z1320392J**Nombre:** GEOFFREY**Apellidos:** BALL**Teléfono:** +34 752 13 91 44**Domicilio:** AV DE MADRID, 17, 23003 JAÉN, ESPAÑA**Código Postal:** 23003**Población:** JAÉN**Provincia:** JAÉN**Fecha de Nacimiento:** 31 / 08 / 2019**Lugar de Nacimiento:** LACONIA (US)**Padre:** JOHN BALL**Madre:** SUSSAN BALL**Estado Civil:** COMPROMETIDO**Nivel Cultural:** -**Profesión:** CHEF**Expediente:****Fecha de inscripción:** 31 / 08 / 2019**Número de inscripción:** 7854**Fecha de finalización:** 31 / 08 / 2019**Estado del expediente:** TEORICO**Observaciones****Recibida fotocopia del DNI:** SI**Recibido certificado médico:** SI**Recibidas fotos del carné:** SI**Recibido pasaporte:** NO**Otros documentos recibidos****Exámenes realizados**

Fecha	Tipo de Examen	Apto
31 / 08 / 2019	TEORICO	SI
31 / 08 / 2019	MANIOBRAS	NO
31 / 08 / 2019	CIRCULACION	NO

Figura 79: Documento – Ficha del alumno

Zona L**Dirección:** Av. de Madrid, n.º 37 23008 Jaén**Teléfono de contacto:** +34 953 22 47 52 / +34 662 09 87 62**Correo electrónico:** zona.l.autoescuela@gmail.com**Información del Vehículo :****Matrícula:** E 0404 VPO**Marca:** FIAT**Modelo:** PUNTO**Última revisión de la ITV:** 13 / 08 / 2019**Fecha de caducidad de la ITV:** 15 / 04 / 2020**Operaciones de mantenimiento:**

Fecha	Concepto
07 / 08 / 2019	Cambio de ruedas
30 / 08 / 2019	Limpieza tapicería
06 / 08 / 2019	Taller: solución avería en cambio de marchas
31 / 08 / 2019	Limpieza
26 / 08 / 2019	Revisión inyectores

Figura 80: Documento – Informe del vehículo

6.3 A3 – Instalación y configuración del sistemas

El anexo dos recopila las instrucciones de instalación y configuración de los diferentes componentes del sistema en un entorno de producción real.

6.3.1 *Instalación del servidor*

En esta sección los pasos necesarios para realizar una correcta instalación del servidor en el equipo de trabajo del sistema.

6.3.1.1 *Requisitos previos*

La máquina deberá tener instalada una máquina virtual de Java, OpenJDK 11 o superior, la aplicación ha sido desarrollada y probada usando dichas versiones la JVM, por lo que no se recomienda utilizar la versión privativa de Oracle u otras versiones diferentes a esta ya que no se dispone de las garantías necesarias para asegurar su correcto funcionamiento.

De igual forma, la máquina debe tener instalada el sistema de bases de datos PostgreSQL, cuya versión recomendada es la 10.2. Dentro del SGBD se debe de dar de alta al menos una base de datos, de la cuál necesitaremos su dirección y credenciales de acceso.

Si el servidor no se va a instalar en un servidor público, como resulta en el caso de la Autoescuela Zona L, se recomienda configurar la red local para impedir llamadas del exterior a los puertos específicos de la base de datos y el servidor. El programa viene con una configuración inicial por defecto, en la cual se incluyen unos certificados propios, en caso de querer utilizar certificados propios, estos deberán ser previamente generados y firmados por una Autoridad Certificadora válida.

6.3.1.2 *Instalación*

Dentro del directorio de entrega contenedor del proyecto se encontrará dentro dentro del subdirectorio *release* un archivo titulado *servidor_zona_L_tfg1819.jar*, este archivo es un ejecutable de FAT JAR de Java, puede ejecutarse con el siguiente comando:

```
java -jar servidor_zona_L_tfg1819.jar
```

Suponiendo el directorio donde se encuentra el ejecutable como el directorio de trabajo o contexto de la consola.

El ejecutable lanzará una instancia del servidor Apache Tomcat de forma automática escuchando en el puerto establecido en la configuración por defecto. Para sobrescribir la configuración por defecto, se debe crear un archivo llamado *application.yml* en un directorio concreto, el orden de prioridad de búsqueda del archivo es el siguiente:

1. Un subdirectorio */config* del directorio actual.
2. El directorio actual.

Los valores especificados en el archivo de configuración sobrescribirán los valores especificados en rutas de menor prioridad.

Alternativamente, se puede especificar la ruta del archivo de configuración desde la línea de comandos a la hora de ejecutar el programa:

```
java -jar servidor_zona_L_tfg1819.jar -spring.config.location =  
file://ruta_al_archivo_de_configuración
```

A continuación, mostramos los diferentes parámetros de configuración disponibles y sus valores por defecto.

Configuración conexión a base de datos

```
spring.datasource.driver-class-name: org.postgresql.Driver
```

```
spring.datasource.url: jdbc:postgresql://localhost:5432/tfg1819
```

```
spring.datasource.username: tfg1819
```

```
spring.datasource.password: tfg1819
```

Configuración de dialecto SQL para mejorar las consultas en postgresQL

```
spring.jpa.properties.hibernate.dialect:org.hibernate.dialect.PostgreSQLDialect
```

Configuración de Hibernate (create, create, validate, update)

```
spring.jpa.hibernate.ddl-auto : update
```

```
spring.jpa.properties.hibernate.enable_lazy_load_no_trans: true
```

```
spring.datasource.tomcat.max-active: 50
```

```
# Configuración de seguridad
```

```
jwt.secret: julioplumasiete23
```

```
spring.admin.pass: tfg1819
```

```
# Configuración conexión TLS
```

```
server.port: 8443
```

```
server.ssl.key-store-type: PKCS12
```

```
server.ssl.key-store: classpath:config/tfg1819_cert.p12
```

```
server.ssl.key-store-password: tfg1819
```

```
server.ssl.key-alias: tfg1819
```

```
security.require-ssl: true
```

```
# Información de la empresa
```

```
# Configurado por defecto con la información de la empresa  
contratante
```

```
info-empresa.nombre: Zona L
```

```
info-empresa.titular: D. Juan Robles Gutiérrez
```

```
info-empresa.num-provincial: J-0285-01-DG8
```

```
info-empresa.direccion: Av. de Madrid, n37
```

```
info-empresa.poblacion: Jaén
```

info-empresa.cod-postal: 23008

info-empresa.tlf: +34 953 22 47 52 / +34 662 09 87 62

info-empresa.email: zona.1.autoescuela@gmail.com

Explicación de cada uno de los parámetros disponibles:

spring.datasource.driver-class-name : Ruta al controlador necesario para la conexión con la base de datos, valor por defecto, no se recomienda modificarlo.

spring.datasource.url : Ruta de conexión con la base de datos, a configurar con la ruta de la base de datos utilizada.

spring.datasource.username : Nombre de usuario de acceso a la base de datos, modificable.

spring.datasource.password : Contraseña de acceso a la base de datos, modificable.

spring.jpa.properties.hibernate.dialect : Dialecto específico de la base de datos para mejorar el rendimiento de las consultas generadas, por defecto, no se recomienda modificarlo.

spring.jpa.hibernate.ddl-auto : Modo de creación de base de datos, se recomienda update por defecto para que el programa genere el esquema definido por la aplicación en caso de que no lo haya, la base de datos registrada en el SGBD debe estar vacía o tener un esquema válido. La opción *create* genera el esquema cada vez que el servidor es ejecutado borrando los datos previos, la opción *validate* valida que el esquema es correcto antes de continuar.

spring.jpa.properties.hibernate.enable_lazy_load_no_trans: Configuración de carga perezosa en Hibernate, por defecto, no se recomienda modificarlo.

spring.datasource.tomcat.max-active : máximo de hilos de consulta disponibles, por defecto, no se recomienda modificarlo.

jwt.secret : Clave secreta para la firma de los tokens JWT. Se recomienda modificarla.

spring.admin.pass : Clave del usuario administrador por defecto, se recomienda modificarla.

server.port : Puerto de escucha.

server.ssl.key-store-type : Tipo de archivo para el almacén de certificados, por defecto, no se recomienda modificarlo.

server.ssl.key-store : Almacén de certificados TLS.

server.ssl.key-store-password : Contraseña del almacén de certificados.

server.ssl.key-alias : Nombre del almacén de certificados.

security.require-ssl : Fuerza la conexión TLS, por defecto, no se recomienda modificarlo.

info-empresa.nombre : Nombre de la empresa.

info-empresa.titular : Titular de la empresa.

info-empresa.num-provincial : Código provincial de la empresa.

info-empresa.direccion : Dirección de la empresa.

info-empresa.poblacion : Localidad de la empresa.

info-empresa.cod-postal : Código postal de la empresa.

info-empresa.tlf : Número de teléfono de la empresa.

info-empresa.email : Correo electrónico de la empresa.

6.3.2 Instalación del Cliente Móvil

6.3.2.1 Requisitos previos

Los requisitos para la instalación del cliente móvil son escasos pero de vital importancia. Estos requisitos son:

- **Una versión de la aplicación del cliente móvil configurada para la IP donde se aloje el servidor.** Esto es muy importante en nuestro caso pues la autoescuela carece de nombre de dominio.
 - La IP se puede cambiar desde el entorno de desarrollo y se halla dentro de la clase ApiLink como el atributo BASE_URL. Este atributo está configurado como "<https://10.0.2.2:8443>" que es la IP por defecto para conectar con la maquina host de el emulador Android que nos proporciona Android Studio.
- **Un dispositivo Android con versión API 19 4.4 (KitKat).** Esta versión funciona en el 95% de los dispositivos con lo cual es improbable que suponga ningún problema.

6.3.2.2 Instalación

Los pasos para la instalación de la aplicación son los siguientes:

- 1) Una vez configurada la aplicación y obtenido el instalador Apk proporcionar el instalador al dispositivo.
- 2) En el dispositivo al ejecutar el instalador. El sistema operativo avisará de que se ha bloqueado por motivos de seguridad. Android por defecto no permite instalar aplicaciones de origen desconocido.
 - Nos propondrá ir a ajustes. Aceptar y activar 'Orígenes Desconocidos'.
 - Volver a ejecutar el instalador. Tal vez la nos salga un aviso de Play Protect advirtiéndonos de no instalarla. Continuar de todas formas.
 - Ejecutar la aplicación

6.4 A4 – Manuales de Usuario

En el anexo tres se presentan los manuales de usuario necesarios para manejar las aplicaciones del sistema.

6.4.1 *Cliente Móvil*

(Se adjunta versión independiente de la memoria como archivo PDF)

Manual de Usuario Cliente Android



6.4.1.1 *Introducción*

Este documento explica de forma general el funcionamiento y las características del cliente móvil del trabajo de fin de grado con código (Cód.: 18/19-2318) desarrollado por Alberto Carrillo Ortega en la Universidad de Jaén durante el curso 2018-2019.

Resumen del sistema

El cliente móvil es una aplicación para dispositivos Android con versión API 19 4.4 (KitKat) o superiores. La aplicación permite conectar con la realización de las diferentes actividades realizadas en la autoescuela *Zona L* de forma electrónica mediante la conexión con el servidor de la misma. Para el funcionamiento de la misma es necesario conexión permanente con dicho servidor.

Configuración del sistema

El cliente móvil requiere de conexión permanente al servidor de la autoescuela. Además, para las funcionalidades de almacenamiento de documentos se requiere de permisos de almacenamiento por parte del usuario del dispositivo. Tras una correcta instalación, el cliente móvil puede ejecutarse sin ninguna configuración previa.

Contingencias

En caso de pérdida de energía en el dispositivo, la aplicación no almacena ningún dato excepto los documentos ya almacenados en la carpeta de descargas. En caso de desconexión con la red, ninguna de las funciones de la aplicación estarán disponibles. Sin embargo, la aplicación no se cerrará y las funciones se restablecerán tan pronto como se encuentre disponible la conexión con el servidor.

6.4.1.2 Vistas y navegación

En esta sección se describirán brevemente las vistas de la aplicación, su uso y la navegación entre estas.

Barra de Navegación y parámetros de navegación.

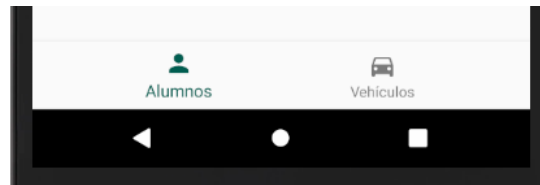


Figura 81: Barra de navegación

La barra de navegación (Figura 81) está presente en todas la vistas de la aplicación y sus elementos nos permiten navegar entre los diferentes conjuntos de vistas.

- El botón “Alumnos” nos redirige a el módulo de vistas de Alumnado. Concretamente a la vista inicial *BuscarAlumno* (Figura 83).
- El botón “Vehículos” nos redirige a el módulo de vistas de Vehículos. Concretamente a la vista *MostrarVehículos* (Figura 97).

En todo momento durante la aplicación podemos pulsar en el botón “atrás”  para retroceder a la vista anterior. Pulsar los botones de la barra de navegación reinicia las pila de vistas por lo que si pulsamos en “Alumnos” y pulsamos en el botón “atrás” la aplicación se minimizará. Con el botón “Vehículos” sucede una excepción y nos devolvería a la vista inicial *BuscarAlumno* (Figura 83).

Barra de progreso

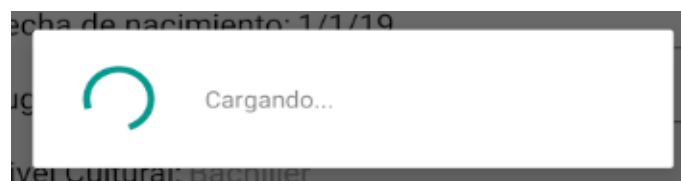


Figura 82: Barra de progreso

La barra de progreso (Figura 83) se muestra en diferentes momentos durante el uso de la aplicación. Frecuentemente se muestra durante la espera de respuesta por parte del servidor. Mientras esté en pantalla la interacción estará desactivada.

Módulo de Alumnado

Vista *Buscar Alumno*



Figura 83: Vista *BuscarAlumno*

La vista *BuscarAlumno* (Figura 83) consiste en un campo para introducir el DNI del alumno que se desea buscar; así como un botón para realizar la búsqueda y visualizar los datos del alumno, navegando a *MostrarAlumno* (Figura 86); y el botón “Añadir Alumno” para crear un alumno, navegando a la vista *CrearAlumno* (Figura 84 y Figura 85).

Vista Crear Alumno

4:09

← Crear Alumno

DNI: 00000000X

Nombre: Alumno

Apellidos: Apellido 1, Apellido 2

Teléfono: +00 000 00 00 00

Correo electrónico: alumno@gmail.com

Provincia: Jaén

Población: Jaén

Código Postal: 00000 Postal

Domicilio: Av. de Madrid, n00

Fecha de nacimiento: 1/1/1990

Lugar de nacimiento: Jaén

Nivel Cultural: Bachiller

Alumnos Vehículos

Figura 84: Vista CrearAlumno

4:09

← Crear Alumno

Correo electrónico: aco00032@red.ujaen.es

Provincia: Jaén

Población: Jaén

Código Postal: 23000

Domicilio: Campus Las Lagunillas s/n 230071 Jaén

Fecha de nacimiento: 8/10/1997

Lugar de nacimiento: Jaén

Nivel Cultural: Universitario

Profesión: Desarrollador Software

Nombre de 1º Progenitor Progenitor1

Nombre de 2º Progenitor Progenitor2

Estado Civil: SOLTERO

CREAR ALUMNO

Alumnos Vehículos

Figura 85: Vista CrearAlumno

La vista *CrearAlumno* (Figura 84 y Figura 85) consiste en una serie de campos para introducir los datos del alumno que se desea crear; así como un botón para realizar la creación del mismo una vez rellenados los datos y mostrarlos, navegando a *MostrarAlumno* (Figura 86).

Vista Mostrar Alumno



Figura 86: Vista MostrarAlumno

La vista *MostrarAlumno* (Figura 86) consiste en una serie de campos no editables que contienen los datos del alumno que se desea mostrar; así como el botón “Editar Alumno” para editar los datos del mismo, navegando a la vista *EditarAlumno* (Figura 87 y Figura 88); el botón “Borrar Alumno” para borrar el alumno; y el botón “Mostrar Expedientes” para mostrar los expedientes de alumno, navegando a la vista *MostrarExpedientes* (Figura 89 y Figura 90).

Vista Editar Alumno

4:11

← Editar Alumno

DNI: 77432305E

Nombre: Alberto

Apellidos: Carrillo, Ortega

Teléfono: +64 000 00 00 00

Correo electrónico: aco00032@red.ujaen.es

Provincia: Jaén

Población: Jaén

Código Postal: 23000

Domicilio: Campus Las Lagunillas s/n 230071 Jaén

Fecha de nacimiento: Sun Aug 10 00:00:00 GMT 19

Lugar de nacimiento: Jaén

Nivel Cultural: Universitario

Alumnos Vehículos

Figura 87: Vista EditarAlumno

4:11

← Editar Alumno

Correo electrónico: aco00032@red.ujaen.es

Provincia: Jaén

Población: Jaén

Código Postal: 23000

Domicilio: Campus Las Lagunillas s/n 230071 Jaén

Fecha de nacimiento: Sun Aug 10 00:00:00 GMT 19

Lugar de nacimiento: Jaén

Nivel Cultural: Universitario

Profesión: Desarrollador Software

Nombre de 1º Progenitor Progenitor1

Nombre de 2º Progenitor Progenitor2

Estado Civil: SOLTERO

EDITAR ALUMNO

Alumnos Vehículos

Figura 88: Vista EditarAlumno

La vista *EditarAlumno* (Figura 87 y Figura 88) consiste en una serie de campos para modificar los datos del alumno que se desea editar; así como un botón para realizar la edición del mismo una vez modificados los datos y mostrarlos, navegando a *MostrarAlumno* (Figura 86).

Vista Mostrar Expedientes

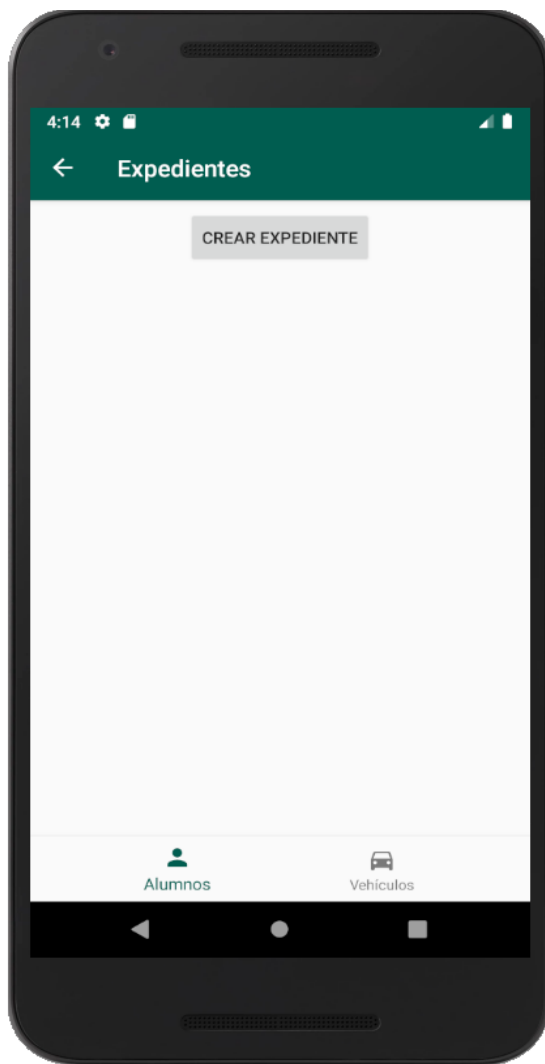


Figura 89: Vista MostrarExpedientes

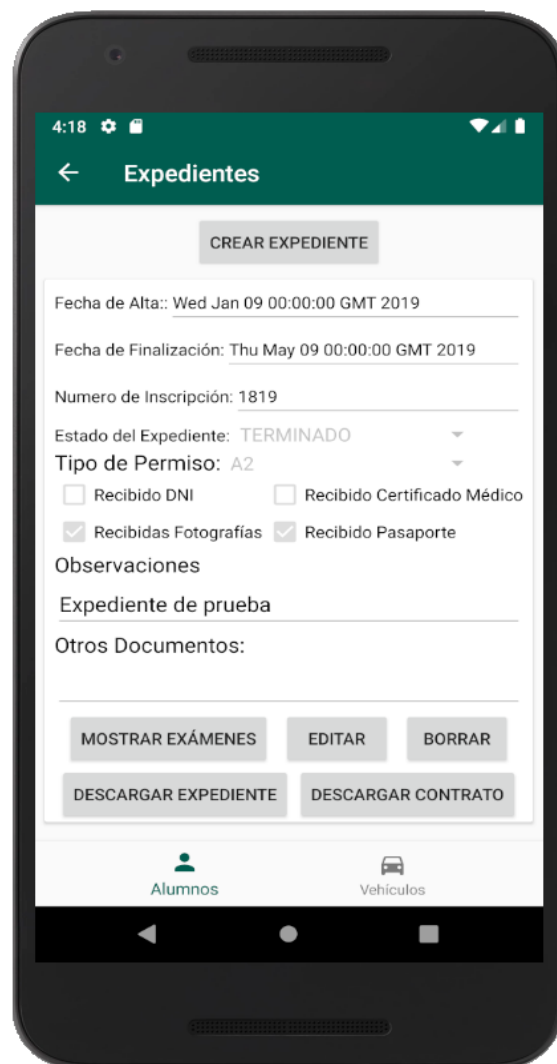


Figura 90: Vista MostrarExpedientes

La vista *MostrarExpedientes* (Figura 89 y Figura 90) consiste en un botón “Crear Expediente” para crear un expediente, navegando a la vista *CrearExpediente* (Figura 91 y Figura 92); y una lista de expedientes.

Cada expediente de la lista contiene los datos del mismo; así como el botón “Mostrar Exámenes” para mostrar los exámenes del expediente, navegando a la vista *MostrarExámenes* (Figura 94 y Figura 95); el botón “Editar Expediente”, para editar los datos del expediente, navegando a la vista *EditarExpediente* (Figura 93), el botón “Borrar” para borrar el expediente; el botón “Descargar Expediente”, para almacenar en la carpeta de descargas el documento del expediente; y el botón “Descargar Contrato” para almacenar en la carpeta de descargas el contrato del expediente.

Vista Crear Expediente

4:14

← Crear Expediente

Fecha de Alta:: Ej: 1/1/2000

Fecha de Finalización: Ej: 1/1/2001

Numero de Inscripción: Ej: 1

Estado del Expediente: TEORICO

Tipo de Permiso: AM

☐ Recibidas Fotografías

☐ Recibido Pasaporte

☐ Recibido DNI

☐ Recibido Certificado Médico

Observaciones

Otros Documentos:

CREAR EXPEDIENTE

Alumnos Vehículos

Figura 91: Vista CrearExpediente

4:18

← Crear Expediente

Fecha de Alta:: 1/9/2019

Fecha de Finalización: 5/9/2019

Numero de Inscripción: 1819

Estado del Expediente: TERMINADO

Tipo de Permiso: AM

☒ Recibidas Fotografías

☒ Recibido Pasaporte

☐ Recibido DNI

☐ Recibido Certificado Médico

Observaciones

Expediente de prueba

Otros Documentos:

CREAR EXPEDIENTE

Alumnos Vehículos

Figura 92: Vista CrearExpediente

La vista *CrearExpediente* (Figura 91 y Figura 92) consiste en una serie de campos para introducir los datos del expediente que se desea crear; así como un botón para realizar la creación del mismo una vez rellenados los datos y mostrarlos, navegando a *MostrarExpedientes* (Figura 89 y Figura 90).

Vista Editar Expediente

Figura 93: Vista EditarExpediente

La vista *EditarExpediente* (Figura 93) consiste en una serie de campos para modificar los datos del expediente que se desea editar; así como un botón para realizar la edición del mismo una vez modificados los datos y mostrarlos, navegando a *MostrarExpedientes* (Figura 89 y Figura 90).

Vista Mostrar Exámenes

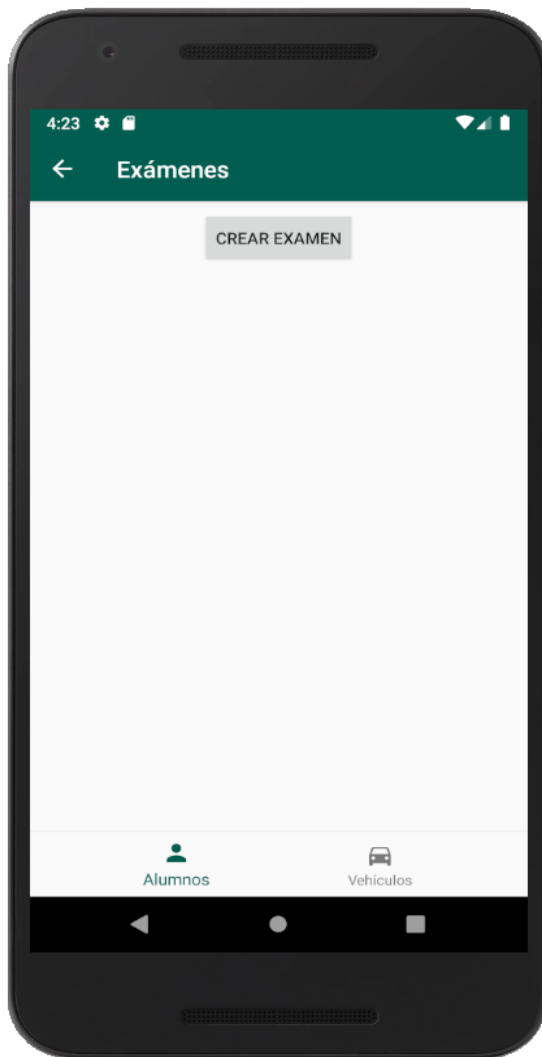


Figura 94: Vista MostrarExámenes

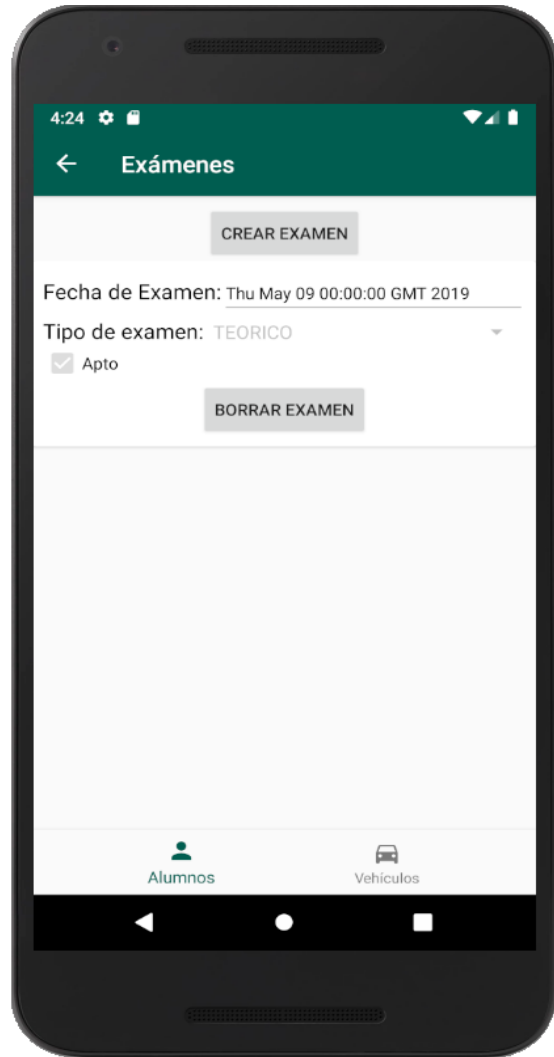


Figura 95: Vista MostrarExámenes

La vista *MostrarExámenes* (Figura 94 y Figura 95) consiste en un botón “Crear Examen” para crear un examen, navegando a la vista *CrearExamen* (Figura 96); y una lista de exámenes.

Cada exámenes de la lista contiene los datos del mismo; así como el botón “Borrar” para borrar el examen.

Vista Crear Examen

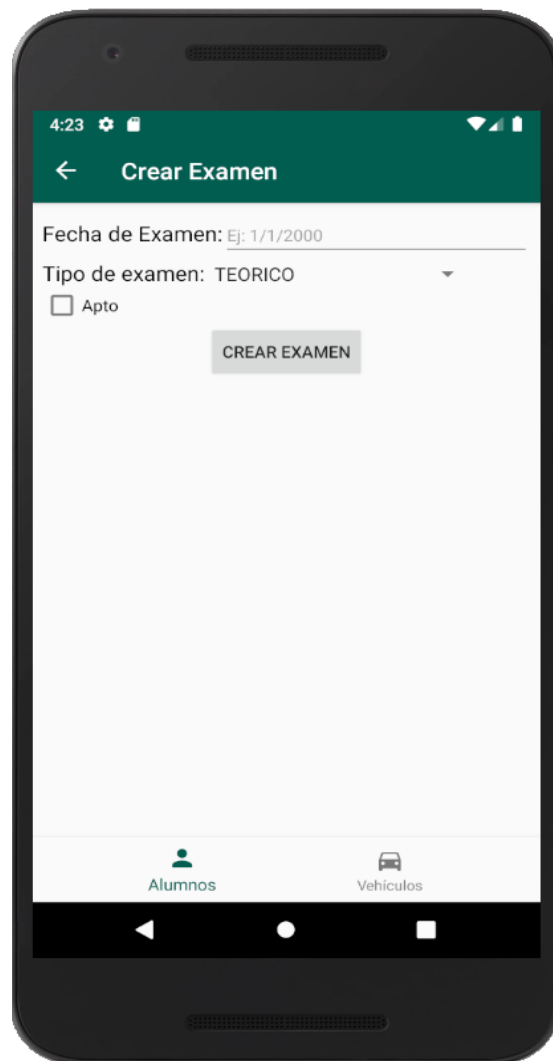


Figura 96: Vista CrearExamen

La vista *CrearExamen* (Figura 96) consiste en una serie de campos para introducir los datos del examen que se desea crear; así como un botón para realizar la creación del mismo una vez rellenados los datos y mostrarlos, navegando a *MostrarExámenes* (Figura 94 y Figura 95).

Modulo de Vehículos

Vista Mostrar Vehículos

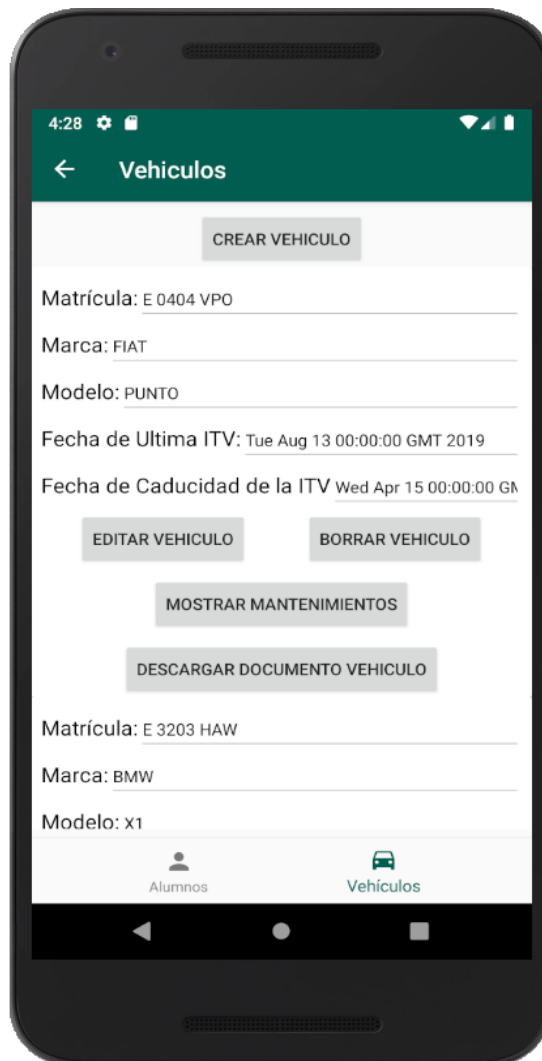


Figura 97: Vista *MostrarVehiculos*

La vista *MostrarVehiculos* (Figura 97) consiste en un botón “Crear Vehículo” para crear un vehículo, navegando a la vista *CrearVehículo* (Figura 98 y Figura 99); y una lista de vehículos.

Cada vehículo de la lista contiene los datos del mismo; así como el botón “Mostrar Mantenimientos” para mostrar las operaciones de mantenimiento del vehículo, navegando a la vista *MostrarMantenimientos* (Figura 101); el botón “Editar Vehículo”, para editar los datos del vehículo, navegando a la vista *EditarVehículo* (Figura 100), el botón “Borrar” para borrar el vehículo; y el botón “Descargar Documento Vehículo” para almacenar en la carpeta de descargas el documento con los datos del vehículo.

Vista Crear Vehículo

4:32

← Añadir Vehículo

Matrícula: Ej: E 0000 XXX

Marca: Ej: VOLKSWAGEN

Modelo: Ej: GOLF

Fecha de Ultima ITV: Ej: 1/1/1990

Fecha de Caducidad de la ITV Ej: 1/1/2000

CREAR VEHICULO

Alumnos Vehículos

Figura 98: Vista CrearVehículo

4:33

← Añadir Vehículo

Matrícula: E 0000 AAA

Marca: Marca1

Modelo: Modelo1

Fecha de Ultima ITV: 1/1/2019

Fecha de Caducidad de la ITV 1/9/2019

CREAR VEHICULO

Alumnos Vehículos

Figura 99: Vista CrearVehículo

La vista *CrearVehículo* (Figura 98 y Figura 99) consiste en una serie de campos para introducir los datos del vehículo que se desea crear; así como un botón para realizar la creación del mismo una vez rellenados los datos y mostrarlos, navegando a *MostrarVehículos* (Figura 97).

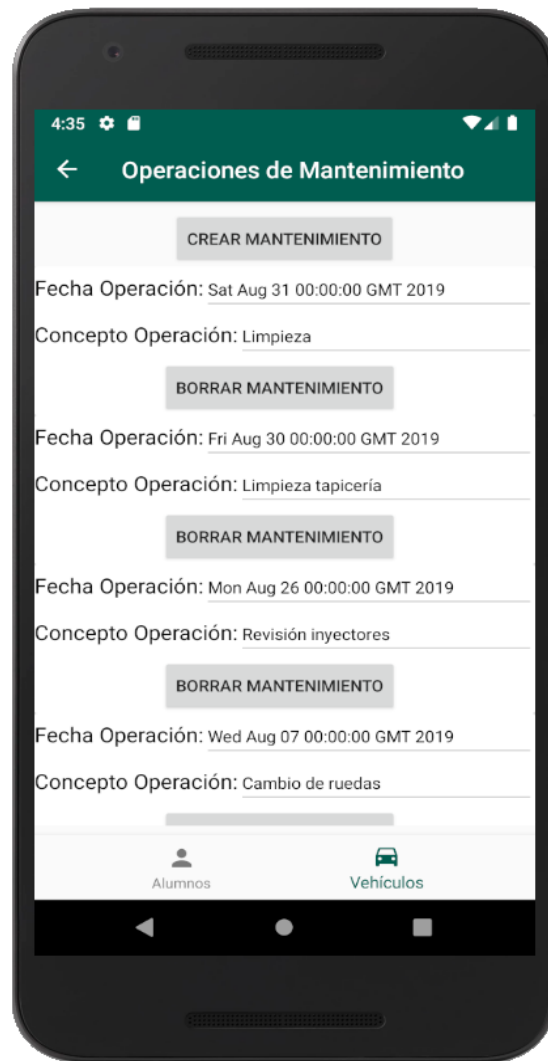
Vista Editar Vehículo



Figura 100: Vista EditarVehículo

La vista *EditarVehículo* (Figura 100) consiste en una serie de campos para modificar los datos del vehículo que se desea editar; así como un botón para realizar la edición del mismo una vez modificados los datos y mostrarlos, navegando a *MostrarVehículos* (Figura 97).

Vista Mostrar Mantenimientos



*Figura 101: Vista
MostrarMantenimientos*

La vista *MostrarMantenimientos* (Figura 101) consiste en un botón “Crear Mantenimiento” para crear un mantenimiento, navegando a la vista *CrearMantenimiento* (Figura 102); y una lista de mantenimientos.

Cada mantenimiento de la lista contiene los datos del mismo; así como el botón “Borrar Mantenimiento” para borrar el mantenimiento.

Vista Crear Mantenimiento

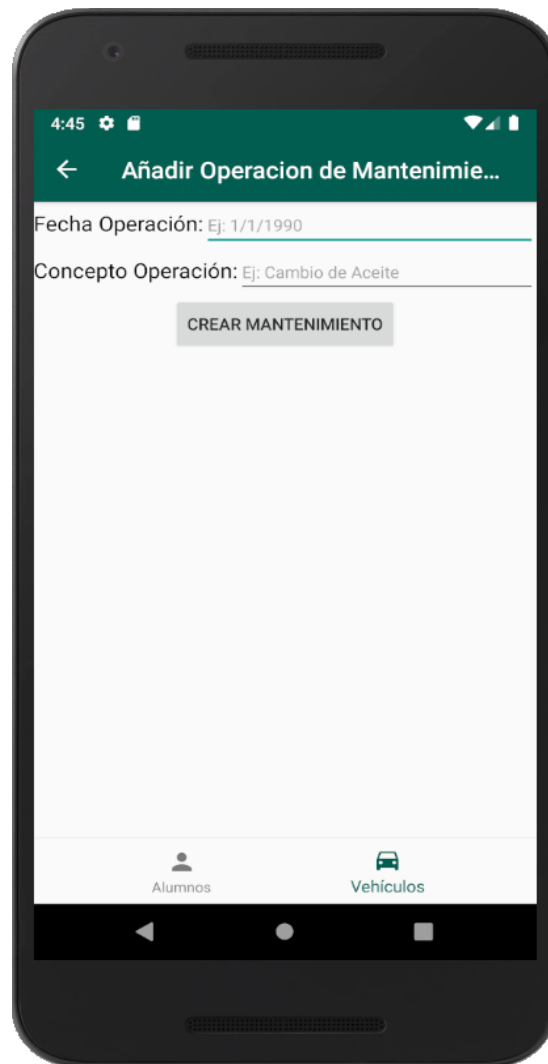


Figura 102: Vista CrearMantenimiento

La vista *CrearMantenimiento* (Figura 102) consiste en una serie de campos para introducir los datos del mantenimiento que se desea crear; así como un botón para realizar la creación del mismo una vez rellenados los datos y mostrarlos, navegando a *MostrarMantenimiento* (Figura 101).

6.4.1.3 Gestión de Errores

Durante el uso de la aplicación es posible que se produzcan errores. En ese caso la aplicación nos mostrará una ventana emergente describiendo en que consiste el error y de ser posible como solventarlo.

Error de elemento no encontrado

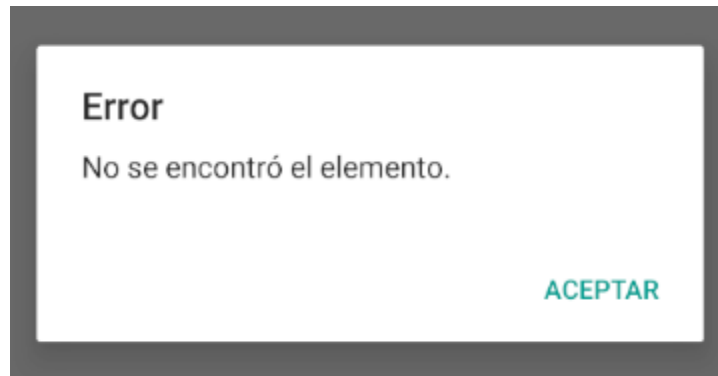


Figura 103: Error de elemento no encontrado

Este error ocurre cuando al buscar alumno no se encuentra ningún alumno con el DNI especificado, o en raras ocasiones cuando otro operario del servidor borra(alumno, expediente, etc) un elemento sobre el que estamos trabajando.

Error de conexión

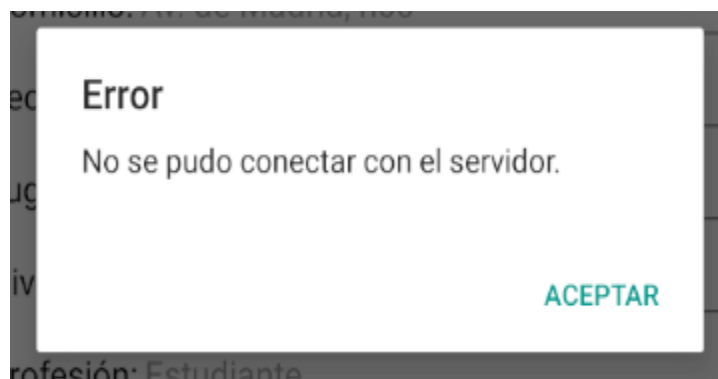


Figura 104: Error de conexión

Este error ocurre cuando al realizar una petición al servidor este no nos ha respondido. Normalmente se debe a que el servidor no se encuentra disponible. En caso de que se produzcan por un periodo largo de tiempo continuado consultar con un administrador.

Error Interno del Servidor

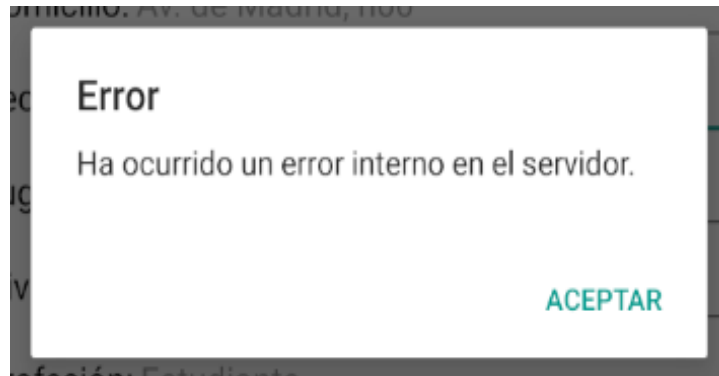


Figura 105: Error Interno del Servidor

Este error puede ocurrir cuando la acción que estamos realizando supone algún problema dentro de la lógica del servidor. Se recomienda en el caso de estar introduciendo datos, revisar los campos escritos. Si el problema persiste consultar con un administrador.

Error Desconocido

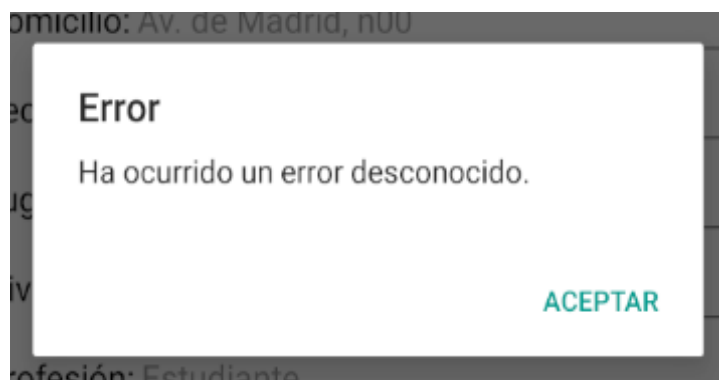


Figura 106: Error Desconocido

Este error puede ocurrir por razones desconocidas. Se recomienda en el caso de estar introduciendo datos, revisar los campos escritos. Si el problema persiste consultar con un administrador.

6.5 A4 – Glosario y Abreviaturas

AFP : *Advanced Function Presentation*. Lenguaje de descripción de páginas utilizado por algunas impresoras.

AJAX : *Asynchronous JavaScript And XML*. Comunicación asíncrona del cliente con el servidor mediante JSON y XML.

AOP : *Aspect Oriented Programming*. Programación orientada a aspectos, paradigma de programación que persigue incrementar la modularidad separando aspectos comunes a un conjunto de funciones.

API : *Application Programming Interface*. Conjunto de funciones y procedimientos que proporciona un sistema para ser utilizada por otro software como capa de abstracción.

APK : *Android Application Package*. Paquete para el sistema operativo Android. Este formato es una variante del formato JAR de Java y se usa para distribuir e instalar componentes empaquetados para la plataforma Android.

ASCII : *American Standard Code for Information Interchange*. Código de caracteres basado en el alfabeto latino, tal como se usa en inglés moderno.

AWT (1) : *Abstract Window Toolkit*. Herramientas de interfaz de usuario diseñadas para ser independientes de la plataforma Java utilizada.

AWT (2) : *AbiWord Template*. Formato de plantilla de Abiword, un procesador de textos.

Base64url : Codificación basada en el sistema de numeración posicional con base 64. Es la mayor potencia de dos representable con caracteres ASCII.

BOE : *Boletín Oficial del Estado*. Diario oficial español dedicado a la publicación de determinadas leyes, disposiciones y actos de inserción obligatoria.

Burndown Chart : Diagrama que representa el esfuerzo restante para finalizar el proyecto a lo largo del tiempo.

CASE : *Computer Aided Software Engineering* Diversas aplicaciones informáticas o programas informáticos destinadas a aumentar el balance en el desarrollo de software .

CSS : *Cascading Style Sheets*. Lenguaje de diseño gráfico que define la disposición y aspecto de un conjunto de elementos.

DB : *Data Base*. Base de datos.

Delphi : Técnica de estimación basada en un proceso iterativo para la obtención de un consenso entre expertos.

DNI : *Documento Nacional de Identidad*.

Historia de Usuario : Forma de representación de un requisito en un desarrollo Scrum.

HTML : *Hypertext Markup Language*. Lenguaje de marcas utilizado para la construcción de documentos web.

HTTP : Hypertext Transfer Protocol. Protocolo de transferencia de hipertexto usado para la transmisión de contenidos por la red.

IoC : *Inversion of Control*. Principio de programación que invierte el flujo de control gracias a un marco de trabajo genérico.

IVA : *Impuesto al Valor Agregado*. Carga fiscal sobre el consumo.

ITV : *Inspección Técnica de Vehículos*. Mantenimiento legal preventivo en el que un vehículo es revisado periódicamente por un ente certificador que verifica el cumplimiento de las normas de seguridad y emisiones contaminantes que le sean aplicables.

JAR : *Java Archive*. Tipo de archivo que permite ejecutar aplicaciones y herramientas escritas en el lenguaje Java.

JAX RS : *Java API for Restful Web Services*. API Java que proporciona soporte en la creación de servicios web siguiendo la arquitectura REST.

JPA : *Java Persistence API*. Api de persistencia desarrollada para la plataforma Java EE.

JSON : *JavaScript Object Notation*. Formato de texto sencillo para el intercambio de datos.

JVM . *Java Virtual Machine*. Máquina virtual encargada de ejecutar código java en el equipo independientemente de la plataforma.

JWT : JSON Web Token. Tecnología de gestión de sesiones centrada en mantener el estado de la sesión en las peticiones mediante tokens en formato JSON.

MoSCoW : Técnica usada en la gestión de proyectos para priorizar e indicar la importancia de un conjunto de tareas. El acrónimo se basa en la primera letra de las cuatro categorías de priorización existentes (*Must have*, *Should have*, *Could have* y *Would have*).

MVC : *Modelo Vista Controlador*. Arquitectura utilizada para mostrar la información de un modelo independientemente de la vista, desacoplando estos dos.

OOP : *Object Oriented Programming*. Programación Orientada a Objetos, paradigma de programación basado en encapsular datos y operaciones asociadas en objetos para evitar acoplamiento y favorecer la cohesión.

ORM : *Object Relational Mapping*. Técnica de programación utilizada para persistir datos usados en un lenguaje de programación en una base de datos relacional.

PC : *Personal Computer*. Ordenador personal.

PCL : *Printer Comand Language*. Lenguaje de descripción de páginas utilizado por algunas impresoras.

PDF : *Portable Document Format*. Formato de almacenamiento para documentos digitales independiente de plataformas de software o hardware.

PMI : *Project Management Institute*. Organización estadounidense sin ánimo de lucro que asocia a profesionales relacionados con la Gestión de Proyectos.

PMBOK : *Project Management Base of Knowledge*. Conjunto de guías y buenas prácticas a la hora de gestionar un proyecto recopilado y mantenido por el PMI.

PNG : *Portable Network Graphics*. Formato gráfico basado en un algoritmo de compresión sin pérdidas no sujeto a patentes.

Product Backlog : Conjunto de historias de usuario pertenecientes a un proyecto Scrum.

Product Owner : Persona dentro de un equipo Scrum encargada de manejar los requisitos y opinión del cliente y transmitirlos al resto del equipo.

PS : *PostScript*. Lenguaje de descripción de páginas utilizado por muchas impresoras.

Punto de historia : Unidad relativa utilizada para estimar el esfuerzo necesario para realizar una tarea.

REST : *Representational State Transfer*. Estilo de arquitectura orientado a sistemas hipermedia en internet basado en un conjunto de operaciones bien definidas mediante los estados HTTP.

RBS : *Requirements Breakdown Structure*. Diagrama de descomposición de requisitos, muestra los requisitos de forma jerárquica llegando a tener hasta tres niveles en la jerarquía, funciones, subfunciones y procesos.

Scrum : Marco de trabajo para el desarrollo ágil de software, basado en el desarrollo iterativo y el feedback del usuario.

Scrum Master : Persona dentro de un equipo Scrum encarga de supervisar y controlar el proceso de trabajo.

SGBD : *Sistema de Gestión de Bases de Datos*. Software encargado de mantener un conjunto de datos en almacenamiento persistente asegurando integridad, persistencia, disponibilidad y confidencialidad.

Sprint : Iteración dentro de un proceso de desarrollo Scrum.

SQL : *Structured Query Language*. Lenguaje de programación utilizado para administrar y recuperar información de SGBDs relacionales.

TFG : *Trabajo de Fin de Grado*.

UML : *Unified Modeling Language* Lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

WBS : *Work Breakdown Structure*. Diagrama de descomposición del trabajo, muestra y desglosa las tareas a realizar en el proyecto en una jerarquía.

XHTML : *eXtensible HyperText Markup Language*. HTML expresado como XML válido.

XML : *eXtensible Markup Language*. Lenguaje de marcas extensible utilizado en múltiples tecnologías.

XSL : *eXtensible Stylesheet Language*. Lenguaje de diseño gráfico basado en XML que define la disposición y aspecto de un conjunto de elementos.

XSLT : *XSL Transformations*. Documento que especifica las transformaciones necesarias para convertir un documento XML en otro.

XSL-FO : *eXtensible Stylesheet Language Formating Objects*. Documento XML que especifica el formato de los elementos para mostrarlos en pantalla.

ZIP : Formato de compresión sin pérdida, muy utilizado para la compresión de datos como documentos, imágenes o programas.

6.6 A6 – Guía Original del Trabajo de Fin de Grado

Ficha del Trabajo Fin de Grado

(Cód.: 18/19-2318) Análisis, diseño e implementación de una aplicación web de gestión de autoescuelas: backend+ frontend android

Tutor del TFG: **ANDRES MOLINA AGUILAR**

Modalidad: Proyecto de Ingeniería | Tipo: TFG Específico

Número máximo de estudiantes: 1 (1 asignados)

Idioma: Castellano

Asignado al alumno con DNI: 77432305

Segundo tutor del TFG: **ÁNGEL LUIS GARCÍA FERNÁNDEZ**

Se pretende implementar una aplicación para gestionar diversos aspectos relacionado con la actividad profesional y académica desarrollada en una autoescuela.

Consistirá en un sistema con arquitectura cliente - servidor en la que se desarrollará una aplicación del lado del servidor y una aplicación del lado del cliente para smartphones

1. Obtención de requisitos, organización del proyecto y correcta documentación del mismo.
2. Análisis y diseño de la todos los componentes del sistema (aplicación del lado del servidor y cliente).
3. Implementación, control y corrección de errores de la aplicación del lado del servidor.
4. Implementación, control y corrección de errores de la aplicación cliente android

Conocimientos Previos

Gestión de Proyectos Software

Desarrollo de Aplicaciones Web

Desarrollo de Aplicaciones para dispositivos móviles

Objetivos del TFG

- Realizar un estudio de necesidades y soluciones para la empresa colaboradora
- Implementar un sistema informático adecuado a los requisitos del negocio siguiendo una metodología de trabajo consolidada en la ingeniería del software

Metodología a Desarrollar

- Estudio de los requisitos del sistema a desarrollar.
- Estudio y elección de la metodología de desarrollo acorde al proyecto
- Estimación temporal y coste del desarrollo.
- Diseño de los componentes de la aplicación, estructura del sistema e interfaces de comunicación.
- Implementación de los componentes diseñados conforme a la especificación establecida.
- Control, revisión y corrección de errores en el código.

Documentos y Formatos de Entrega

- Se entregará documentación sobre el estado y evolución del proyecto, entrevistas, requisitos, el diseño en sus diferentes fases, así como la justificación de las decisiones tomadas durante el desarrollo en la memoria.
- La memoria se entregará únicamente en formato PDF junto con un fichero .zip que contenga el código fuente desarrollado, últimas versiones del/os ejecutables, documentación del código y demás documentos y archivos adicionales a los que se haga mención expresa en la memoria
- El material anteriormente indicado se entregará almacenado en una memoria flash USB

7 BIBLIOGRAFÍA Y FUENTES

[1] *Documentación JavaFX*

- Recuperado de <https://openjfx.io/javadoc/11/> el 1/6/2019
- Recuperado de <https://openjfx.io/openjfx-docs/> el 1/6/2019

[2] *Baeldung | Java, Spring and Web Development tutorials*

- Recuperado de <https://www.baeldung.com/> el 1/6/2019

[3] *Documentación Java*

- Recuperado de <https://docs.oracle.com/javase/8/docs/> el 1/6/2019

[4] *Documentación Android*

- Recuperado de <https://developer.android.com/docs> el 3/6/2019

[5] *Wikipedia, La enciclopedia Libre*

- Recuperado de <https://en.wikipedia.org/> el 15/7/2019

[6] *Stack Overflow. For developers, by developers*

- Recuperado de <https://stackoverflow.com/> el 6/6/2019

[7] *Project Management Institute*

- Recuperado de <https://www.pmi.org/> el 1/6/2019

[8] *Manifesto for Agile Software Development*

- Recuperado de <https://agilemanifesto.org> el 1/6/2019

[9] *Documentación Visual Paradigm*

- Recuperado de <https://www.visual-paradigm.com/support/documents/> el 12/6/2019

[10] *Documentación LibreOffice*

- Recuperado de <https://documentation.libreoffice.org/es/documentacion-en-espanol/> el 10/6/2019

[11] *Documentación y tutoriales de Retrofit*

- Recuperado de <https://square.github.io/retrofit/> el 8/6/2019

[12] *Documentación y tutoriales de OkHttp*

- Recuperado de <https://square.github.io/okhttp/> el 8/6/2019

[13] *Documentación y tutoriales de PostgreSQL*

- Recuperado de <https://www.postgresql.org/docs/> el 2/6/2019

[14] *REST API Design: Filtering, sorting and pagination*

- Recuperado de <https://www.moesif.com/blog/technical/api-design/REST-API-Design-Filtering-Sorting-and-Pagination/#> el 13/7/2019

[15] *JWT Token Authentication with Spring Boot*

- Recuperado de <https://svlada.com/jwt-token-authentication-with-spring-boot/#jwt-authentication> el 29/7/2019

[16] *XVII Convenio colectivo estatal de empresas de consultoría, y estudios de mercados y de la opinión pública*

- Recuperado de <https://www.boe.es/boe/dias/2018/03/06/pdfs/BOE-A-2018-3156.pdf> el 29/7/2019