



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería
Informática**

**Aplicación web de gestión de
PDI en universidades
Documentación Técnica**



Presentado por Ignacio Dávila García
en Universidad de Burgos — 26 de junio
de 2023

Tutores: Álvaro Arnaiz González y Carlos
Pardo Aguilar

Índice general

Índice general	i
Índice de figuras	iii
Índice de tablas	v
Apéndice A Plan de Proyecto Software	1
A.1. Introducción	1
A.2. Planificación temporal	1
A.3. Estudio de viabilidad	23
Apéndice B Especificación de Requisitos	29
B.1. Introducción	29
B.2. Objetivos generales	29
B.3. Catálogo de requisitos	29
B.4. Especificación de requisitos	31
Apéndice C Especificación de diseño	79
C.1. Introducción	79
C.2. Diseño de datos	79
C.3. Diseño procedimental	89
C.4. Diseño arquitectónico	97
Apéndice D Documentación técnica de programación	101
D.1. Introducción	101
D.2. Estructura de directorios	101
D.3. Manual del programador	104

D.4. Compilación, instalación y ejecución del proyecto	104
D.5. Pruebas del sistema	104
Apéndice E Documentación de usuario	105
E.1. Introducción	105
E.2. Requisitos de usuarios	105
E.3. Instalación	105
E.4. Manual del usuario	105
Bibliografía	107

Índice de figuras

A.1. <i>Burndown Report Sprint 1</i>	2
A.2. <i>Burndown Report Sprint 2</i>	3
A.3. <i>Burndown Report Sprint 3</i>	4
A.4. <i>Burndown Report Sprint 4</i>	6
B.1. Diagrama de casos de uso general	31
B.2. Diagrama de caso de uso - 1. Mantenimiento académico	32
B.3. Vista diagrama entidad relación para el CU-1	39
B.4. Diagrama de caso de uso - 2. Mantenimiento de profesorado	40
B.5. Vista diagrama entidad relación para el CU-2	53
B.6. Diagrama de caso de uso - 3. Asignación docente	54
B.7. Vista diagrama entidad relación para el CU-3	63
B.8. Mantenimiento de centros	64
B.9. Añadir/Modificar centros	64
B.10. Mantenimiento de titulaciones	65
B.11. Añadir/Modificar titulaciones	65
B.12. Mantenimiento de asignaturas	66
B.13. Añadir/Modificar asignaturas	66
B.14. Mantenimiento de departamentos	67
B.15. Añadir/Modificar departamentos	67
B.16. Mantenimiento de áreas	68
B.17. Añadir/Modificar áreas	68
B.18. Mantenimiento de docentes	68
B.19. Añadir/Modificar docentes	69
B.20. Mantenimiento de tipos de contrato	69
B.21. Añadir/Modificar tipos de contrato	70
B.22. Mantenimiento de plazas	70
B.23. Añadir/Modificar plazas	71

B.24. Asignar plaza a docente	71
B.25. Mantenimiento cursos académicos	72
B.26. Modificar año curso académico	72
B.27. Añadir curso académico	73
B.28. Añadir curso académico 2	73
B.29. Modificar curso académico	74
B.30. Añadir asignaturas al curso	74
B.31. Mantenimiento de grupos	75
B.32. Añadir/Modificar grupo	75
B.33. Añadir/Modificar grupo 2	76
B.34. Asignación de horas de plazas a grupos	76
B.35. Asignación de horas de plazas a grupos 2	77
B.36. Asignación de horas de plazas a grupos 3	77
C.1. Diagrama general entidad-relación	80
C.2. Diagrama relacional	82
C.3. <i>Diagrama de secuencia referente a la creación de centros.</i>	90
C.4. <i>Diagrama de secuencia referente a la modificación de centros.</i>	91
C.5. <i>Diagrama de secuencia de la eliminación de centros.</i>	92
C.6. <i>Diagrama de secuencia de la asignación de una plaza a un docente.</i>	94
C.7. <i>Diagrama de secuencia de la creación de un curso académico.</i>	95
C.8. <i>Diagrama de secuencia de creación de grupos.</i>	96
C.9. <i>Diagrama de secuencia de la asignación de horas de una plaza a un grupo.</i>	98
C.10. <i>Modelo-Vista-Controlador</i>	99
D.1. <i>Directorios del proyecto</i>	102

Índice de tablas

A.1. Total de costes durante el desarrollo del proyecto.	25
A.2. Licencias de las bibliotecas utilizadas.	26
B.1. CU-1 Mantenimiento académico.	32
B.2. CU-1.1 Mantenimiento de centros.	33
B.3. CU-1.1.1 Añadir/Modificar centros.	34
B.4. CU-1.2 Mantenimiento de titulaciones.	35
B.5. CU-1.2.1 Añadir/Modificar centros.	36
B.6. CU-1.3 Mantenimiento de asignaturas.	37
B.7. CU-1.3.1 Añadir/Modificar asignaturas.	38
B.8. CU-2 Mantenimiento de profesorado.	41
B.9. CU-2.1 Mantenimiento de departamentos.	42
B.10. CU-2.1.1 Añadir/Modificar departamentos.	43
B.11. CU-2.2 Mantenimiento de áreas.	44
B.12. CU-2.2.1 Añadir/Modificar áreas.	45
B.13. CU-2.3 Mantenimiento de docentes.	46
B.14. CU-2.3.1 Añadir/Modificar docentes.	47
B.15. CU-2.4 Mantenimiento de tipos de contrato.	48
B.16. CU-2.4.1 Añadir/Modificar tipos de contrato.	49
B.17. CU-2.5 Mantenimiento de plazas.	50
B.18. CU-2.5.1 Añadir/Modificar plazas.	51
B.19. CU-2.6 Asignar plaza a docente.	52
B.20. CU-3 Asignación docente.	55
B.21. CU-3.1 Mantenimiento cursos académicos.	56
B.22. CU-3.1.1 Añadir curso académico.	57
B.23. CU-3.1.2 Modificar curso académico.	58
B.24. CU-3.1.3 Duplicar curso académico.	59
B.25. CU-3.2 Mantenimiento de grupos.	60

B.26.CU-3.2.1 Añadir/Modificar grupo.	61
B.27.CU-3.3 Asignación de horas de plazas a grupos.	62
C.1. Diccionario de datos. Tabla Centro	81
C.2. Diccionario de datos. Tabla Titulación	83
C.3. Diccionario de datos. Tabla Asignatura	84
C.4. Diccionario de datos. Tabla Abreviatura	84
C.5. Diccionario de datos. Tabla Curso	84
C.6. Diccionario de datos. Tabla Curso-Asignatura	85
C.7. Diccionario de datos. Tabla Grupo	86
C.8. Diccionario de datos. Tabla Docente	86
C.9. Diccionario de datos. Tabla Departamento	86
C.10.Diccionario de datos. Tabla Área	87
C.11.Diccionario de datos. Tabla Tipo Contrato	87
C.12.Diccionario de datos. Tabla Plaza	88
C.13.Diccionario de datos. Tabla Plaza-Grupo	88

Apéndice A

Plan de Proyecto Software

A.1. Introducción

En este anexo se va a exponer la planificación temporal y el estudio de viabilidad del proyecto.

Siguiendo la metodología Scrum, la planificación temporal está dividida en los *sprints* seguidos para la realización del proyecto. Cada apartado dentro de la planificación cuenta con información sobre la reunión de planificación del *sprint*, de la que se obtienen conjunto de tareas a realizar, y la reunión de revisión, donde se muestra y valora el trabajo realizado.

A.2. Planificación temporal

Durante este proyecto se ha seguido una metodología Scrum con algunos cambios ya que no se hacían reuniones diarias ni los roles estaban tan marcados, pero la forma de trabajar era la misma con planificaciones y revisiones de *sprints*. Los *sprints* que se han realizado son los siguientes:

Sprint 1: Sprint inicial

Fechas: 28 febrero 2023 – 7 marzo 2023.

■ **Planificación del *sprint***

En la reunión de planificación del sprint se fijaron las siguientes tareas:

1. Configuración inicial del repositorio

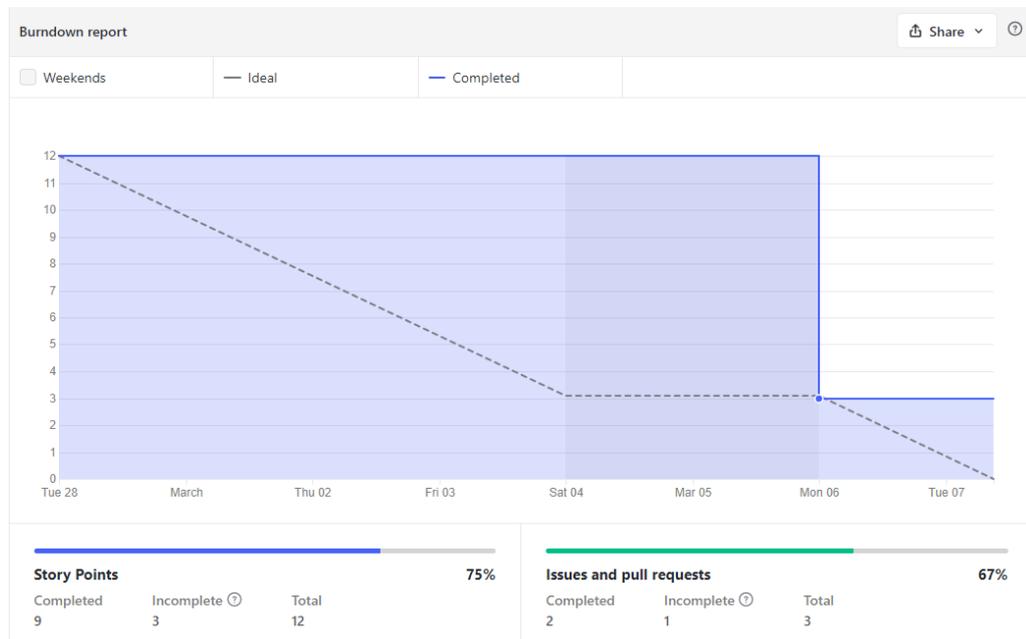


Figura A.1: *Burndown Report Sprint 1*

2. Prototipos de las vistas de la aplicación
3. Creación diagrama entidad-relación

■ ***Burndown Report***

Como se puede apreciar en la figura A.1, no todas las tareas aparecen como completadas. Esto es debido a que el diagrama entidad-relación se dejó abierto ya que faltaba información para completarlo.

■ **Revisión del *sprint***

Durante la revisión se mostró el trabajo realizado y se vieron los cambios que se debían realizar en el diagrama entidad-relación que a su vez implicaban cambios en los prototipos de las vistas de la aplicación. Se llegó a la conclusión de que podía ser buena idea dividir el diagrama E/R haciendo vistas del mismo para que fuere más fácil resolverlo.

Sprint 2: Casos de uso y diagrama E/R de cada uno

Fechas: 7 marzo 2023 – 14 marzo 2023.

Figura A.2: *Burndown Report Sprint 2*

■ Planificación del *sprint*

En la reunión de planificación del sprint se fijaron las siguientes tareas:

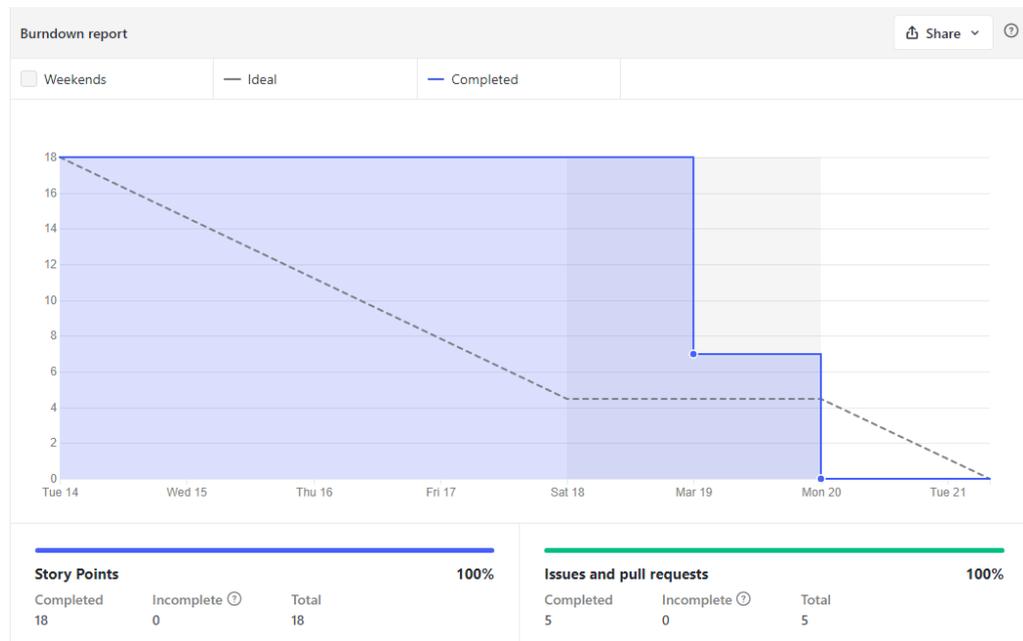
1. Creación de casos de uso junto a su vista del diagrama E/R.
2. Aprendizaje de Flask.

■ *Burndown Report*

En este *sprint* se completaron las tareas marcadas en el tiempo fijado durante la reunión de planificación, pero muchas cosas quedaron pendientes de cambios en futuros *sprints*. En la figura A.2 se puede ver el *Burndown Report* del *sprint*.

■ Revisión del *sprint*

En la reunión de revisión se estudió de nuevo el diagrama E/R y se indicaron nuevos cambios menores en el mismo. También se propuso el comenzar a realizar el diagrama de casos de uso y continuar con el estudio de Flask.

Figura A.3: *Burndown Report Sprint 3*

Sprint 3: Documentación de casos de uso e investigación y aprendizaje de Flask y bibliotecas JavaScript

Fechas: 14 marzo 2023 – 21 marzo 2023.

■ **Planificación del *sprint***

En la reunión de planificación del *sprint* se fijaron las siguientes tareas:

1. Realizar el diagrama de casos de uso
2. Cambios en las vistas adaptándolas a los casos de uso
3. Documentar los casos de uso con sus tablas
4. Investigar bibliotecas de JavaScript que pudiesen ayudar
5. Aprender sobre Flask

■ ***Burndown Report***

En este *sprint* se completaron las tareas marcadas aunque el tiempo marcado para el aprendizaje de Flask fue menor debido a falta de tiempo durante esta semana. Estaba previsto dedicar en total 18 horas

al *sprint*, pero finalmente fueron 15. En la figura A.3 se puede ver el *Burndown Report* del *sprint*.

■ Revisión del *sprint*

Durante la revisión se vio que había casos de uso que no eran necesarios y que se podían añadir como excepciones de otros. Esto produjo que el diagrama de casos de uso se debía cambiar, lo que implica un cambio en la documentación de las tablas y en los prototipos de las vistas de la aplicación.

Sprint 4: Cambios en los casos de uso y vistas, documentación y Flask

Fechas: 21 marzo 2023 – 28 marzo 2023.

■ Planificación del *sprint*

En la reunión de planificación del *sprint* se fijaron las siguientes tareas:

1. Cambios en el diagrama de casos de uso dividiendo en diagrama por niveles y crear vistas del diagrama E/R para los casos de uso.
2. Cambios en algunos prototipos de las vistas de la aplicación.
3. Cambios en la documentación de los casos de uso (tablas).
4. Añadir documentación.
5. Comenzar estructura básica de la aplicación en Flask.

■ *Burndown Report*

A lo largo del *sprint* se realizaron la mayoría de tareas marcadas, pero en las tareas de añadir documentación y comenzar con Flask, no se hizo tanto como estaba esperado debido a la falta de tiempo. Se planeo que se iban a poder dedicar más horas de las que al final se dedicaron y no se avanzó todo lo planeado en estas tareas. Aún así, se decidió cerrarlas, ya que algo si se había avanzado, y crear de nuevo tareas similares en próximos *sprints*.

La tarea de «Cambios en el diagrama de casos de uso dividiendo en diagrama por niveles y crear vistas del diagrama E/R para los casos de uso» tenía una previsión de 2 horas que se realizó en el plazo estimado, la tarea «Cambios en algunos prototipos de las vistas de la aplicación» tenía una estimación de 3 horas aunque finalmente fueron 4, la tarea «Cambios en la documentación de los casos de uso (tablas)» tenía

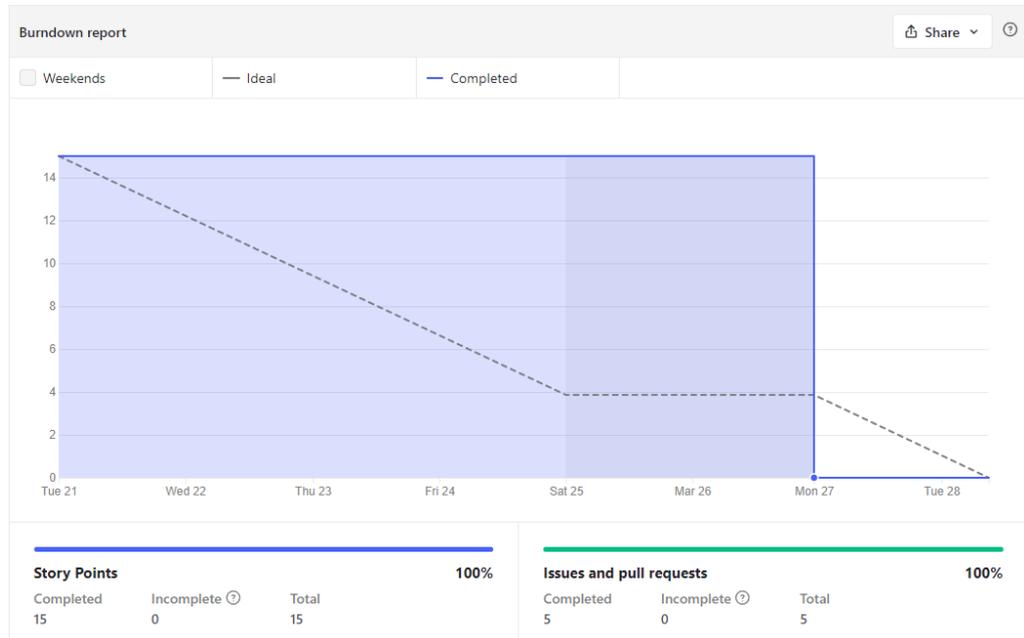


Figura A.4: *Burndown Report Sprint 4*

marcada 4 horas de duración que finalmente fueron 5, la tarea de «Añadir documentación» tenía pensada una dedicación de 3 horas que al final se quedó en 1 hora y media aproximadamente y, por último, la tarea «Comenzar estructura básica de la aplicación en Flask.» tenía una previsión de 3 horas que quedó reducida a una media hora por falta de tiempo durante el *sprint*. En la figura A.4 se puede ver el *Burndown Report* del *sprint*.

■ Revisión del *sprint*

En la reunión de revisión del *sprint* se decidió hacer algunos cambios en los casos de uso, lo que implica un cambio en la documentación de las tablas y los prototipos de las vistas ya que el funcionamiento esperado de la aplicación cambia. También se decidió realizar algún cambio en el diagrama E/R añadiendo nuevos atributos y sacando un atributo a una nueva entidad.

Sprint 5: Documentación y comienzo de la aplicación

Fechas: 28 marzo 2023 – 11 abril 2023.

■ Planificación del *sprint*

En la reunión de planificación del sprint se fijaron las siguientes tareas:

1. Cambios en el diagrama E/R y en algún caso de uso.
2. Cambios en algunas tablas de los CU y en sus respectivas vistas.
3. Evaluar plugins/frameworks para tablas.
4. Añadir documentación.
5. Desarrollo de la aplicación.

■ *Burndown Report*

En este *sprint* surgió un problema con herramienta ZenHub que utilizaba para llevar un seguimiento de los *sprints* desde GitHub. Es una herramienta de pago que en teoría cuenta con versiones gratuitas para proyectos *open source* y para profesores. Contacté con ellos para poder obtener una licencia gratuita de la herramienta, pero no fue posible conseguirla. Por lo tanto, a partir de este *sprint* no pude utilizar la herramienta y no pude sacar la imagen del *Burndown Report*.

Para la tarea de «Cambios en el diagrama E/R y en algún caso de uso» se estimaba un tiempo de realización de 1 hora que se cumplió.

Para la tarea «Cambios en algunas tablas de los CU y en sus respectivas vistas» se puso una estimación de 5 horas que finalmente fueron 7 horas.

Para la tarea «Evaluar plugins/frameworks para tablas» se había marcado una planificación de 2 horas que se cumplió, excediendo un poco de la planificación, ya que aparte de buscar y valorar distintas librerías de JavaScript para la visualización de tablas estuve haciendo pruebas con la librería Grid.js que decidí utilizar para el proyecto.

La tarea de «Añadir documentación» tenía marcada una planificación de 3 horas que se cumplió dentro del plazo.

Por último, la tarea de «Desarrollo de la aplicación», donde se buscaba empezar con la aplicación web, tenía una estimación de 8 horas que se convirtió en unas 10 horas de trabajo real debido a algunos problemas surgidos durante el desarrollo.

■ Revisión del *sprint*

En la reunión de revisión se estuvo viendo el trabajo realizado durante el *sprint*. En la parte de documentación se decidió que se debían hacer algunos pequeños cambios en las tablas de los casos de uso, la

visualización de imágenes, cambios de orden en un diagrama y cambios en algún título.

En la parte de desarrollo de la aplicación web se decidió realizar algunos cambios en la visualización de tablas para que no ocupasen tanta pantalla así como en el menú de la web para hacerlo más pequeño introduciendo submenús. También se dijo que era mejor poner un selector de centro en la pantalla de la asignación de horas para que de esta forma ya saliesen los datos de la tabla filtrados por centro.

Sprint 6: Base de datos y comenzar a dar funcionalidad a la web

Fechas: 11 abril 2023 – 24 abril 2023.

■ **Planificación del *sprint***

En la reunión de planificación del *sprint* se fijaron las siguientes tareas:

1. Creación de los modelos y la base de datos en base a los modelos.
2. Crear la primera carga de la base de datos con CSV y script SQL.
3. Creación de más vistas junto a su funcionalidad.
4. Cambios en la documentación.
5. Añadir más documentación.

■ ***Burndown Report***

Este *sprint* tenía una duración inicial de una semana, pero debido a falta de tiempo por exámenes y entregas de trabajos, decidí aplazar una semana más la reunión de revisión para poder trabajar en el proyecto. Además, se tuvo que cambiar la fecha habitual de las reuniones debido a que empecé prácticas en una empresa y ya no era compatible la hora/día que se tenía.

La primera tarea completada fue la de «Creación de los modelos y la base de datos en base a los modelos». Se puso una estimación de 4 horas, que de trabajo real fue el doble, unas 8 horas. Este exceso de tiempo fue debido en gran medida a errores surgidos a la hora de crear la base de datos utilizando el *ORM SQLAlchemy*. Los problemas eran principalmente por como hacía la carga de la base de datos que hacía que en algunos contextos estuviese disponible y en otros no, provocando errores al ejecutar la aplicación y generar las tablas de la base de datos.

La tarea «Crear la primera carga de la base de datos con CSV y script SQL» tenía una planificación de 2 horas y se completó en el tiempo. Sólo quedo pendiente el hecho de que desde la interfaz phpMyAdmin que estoy utilizando para manejar la base de datos, no fui capaz de cargar los archivos csv desde el script de SQL y tuve que cargarlos por separado desde la interfaz.

Para la tarea «Creación de más vistas junto a su funcionalidad» se tenía una estimación de 6 horas. En este caso fui bastante optimista y el trabajo real termino siendo de 9 horas. Esto se debido en gran parte a la aparición de problemas que hicieron que tuviera que trabajar más horas de las planeadas en esta tarea.

La tarea «Cambios en la documentación» tenía planificadas 2 horas de trabajo. En horas reales termino siendo 2 horas y media aproximadamente.

Finalmente, la tarea de «Añadir más documentación» tenía una estimación de 1 hora y el trabajo realizado fue también de 1 hora. Se añadió sobre todo documentación sobre técnicas y herramientas, apartado de diseño y apartado de plan de proyecto.

■ Revisión del *sprint*

En la reunión de revisión del *sprint* 6 se estuvieron valorando los cambios realizados y las nuevas funcionalidades añadidas. También se resolvieron algunas dudas sobre la documentación y algunas partes de la funcionalidad de la aplicación web. Se discutieron algunos aspectos como el añadir algún nuevo campo en la base de datos para códigos internos o cómo tratar la eliminación de los centros, que hasta ahora se trataba como una eliminación en cascada, y se decidió cambiar al existir demasiado riesgo de pérdida de información por un borrado accidental o poco planificado.

***Sprint* 7: Creación de CRUDs y documentar**

Fechas: 24 abril 2023 – 2 mayo 2023.

■ Planificación del *sprint*

En la reunión de planificación del *sprint* se fijaron las siguientes tareas:

1. Comentarios sobre memoria y anexos.

2. Crear el CRUD¹ de Docentes, Plazas, Contratos, Áreas y Departamentos.
3. Solucionar bug con las abreviaturas al modificar una asignatura.
4. Algunos cambios en campos de la base de datos.
5. Buscar cómo cargar un CSV desde un script de SQL en phpMyAdmin.
6. Añadir documentación.

■ ***Burndown Report***

La tarea de «Comentarios sobre memoria y anexos» fue una tarea creada por Álar Arnaiz González para dejar las correcciones hechas en la memoria y los anexos. Esta tarea fue reutilizada para indicar que se iban a subir esos cambios en la documentación. Tenía una planificación de 30 minutos y se realizó dentro del tiempo estimado.

La tarea «Crear el CRUD de Docentes, Plazas, Contratos, Áreas y Departamentos» era la que tenía más carga de trabajo en este *sprint* con una estimación de 8 horas que se convirtió en 11 horas de trabajo real. Este incremento de horas se debió principalmente a problemas para configurar correctamente la biblioteca Select2 utilizando Ajax y para mantener las opciones marcadas en el campo en las modificaciones.

La tarea «Solucionar bug con las abreviaturas al modificar una asignatura» era debida a errores en el campo `Select2` al hacer una modificación de una asignatura. Las abreviaturas de la asignatura aparecían bien, pero no se recuperaban de forma correcta en el servidor para saber cuales habían sido añadidas o eliminadas. Esta tarea tenía una estimación de 1 hora y no se pretendía superar ese tiempo de trabajo, ya que en la reunión de planificación no se consideró que fuese un problema crucial debido a que en este momento no era necesario que una asignatura tuviera varias abreviaturas y podía ser cambiado por un campo de texto normal que no diese ese problema y permitiese a una asignatura tener una única abreviatura. Al realizar la tarea anterior se tuvo un problema parecido que dio la pista para resolver este problema dejándolo totalmente subsanado.

La tarea «Algunos cambios en campos de la base de datos» tenía una estimación de 1 hora y se realizó dentro del plazo. La tarea consistía en añadir nuevos campos en la base de datos, lo que suponía también

¹CRUD son las siglas de *Create* (crear), *Read* (leer), *Update* (actualizar) y *Delete* (modificar).

hacer cambios en el código para introducir los nuevos atributos en los modelos y en los formularios. También se cambió el borrado en cascada que tenían los centros para evitar que se eliminen centros que tengan titulaciones vinculadas.

Para la tarea «Añadir documentación» se estimaron 3 horas de trabajo que se cumplieron en cuanto a trabajo real.

Finalmente, para la tarea «Buscar cómo cargar un CSV desde un script de SQL en phpMyAdmin» se estimó 1 hora de trabajo que finalmente fue un trabajo real de 2 horas debido a que tuve que investigar por los problemas que me daba con la carga de los ficheros csv. El problema finalmente fue donde estaban colocados los csv.

■ Revisión del *sprint*

En la reunión de revisión se mostró el trabajo realizado y se resolvieron algunas dudas acerca de la memoria y algunas partes del funcionamiento de la aplicación web.

Sprint 8: Gestión de cursos

Fechas: 2 mayo 2023 – 15 mayo 2023.

■ Planificación del *sprint*

En la reunión de planificación del sprint se fijaron las siguientes tareas:

1. Subir la web a Heroku.
2. Programar el apartado de gestión de cursos.
3. Hacer pequeños cambios en la web.
4. Añadir documentación de trabajos relacionados.

■ *Burndown Report*

La primera tarea que se realizó fue «Subir la web a Heroku». La tarea tenía una carga de trabajo estimada en 2 horas. Para esta tarea la estimación no fue nada correcta, ya que el trabajo real fue aproximadamente de 8 horas. Esto fue debido principalmente a problemas a la hora de configurar el proyecto para ser reconocido por la plataforma Heroku y para configurar correctamente la base de datos. Surgieron un gran listado de problemas que fueron resueltos uno por uno hasta que la instalación fue satisfactoria.

Para la tarea «Programar el apartado de gestión de cursos» se habían marcado 8 horas de trabajo que finalmente se convirtieron en mucho más, 20 horas de trabajo. Esta gran discrepancia entre la planificación y el trabajo real fue debida en gran medida a la investigación y planteamiento de cómo mostrar el listado de asignaturas que se deben seleccionar a la hora de crear un curso. Se intentó crear de la forma más cómoda para utilizar y con el menor impacto posible en el rendimiento. También surgieron algunos problemas durante la programación que fueron subsanados pero que hicieron que el tiempo de trabajo se fuese desplazando aun más de la estimación realizada.

La siguiente tarea fue «Hacer pequeños cambios en la web». Esta tarea tenía una estimación de 3 horas que de trabajo real en realidad fueron unas 2 horas. Estos pequeños cambios eran principalmente sobre la visualización del contenido.

Finalmente, la tarea «Añadir documentación de trabajos relacionados» tenía una estimación de trabajo de 1 hora y el trabajo real tuvo también una duración aproximada de una hora.

■ **Revisión del *sprint***

Durante la reunión de revisión se vieron principalmente los cambios realizados en la web. Se estuvieron discutiendo diferentes formas de crear los cursos y se llegó a la conclusión de hacer pequeños cambios para que fuese más cómoda la creación de cursos. También se estuvieron viendo las diferentes secciones de la memoria para dar ideas sobre la documentación que se podía ir introduciendo.

Sprint 9: Cambios, gestión de grupos y documentación

Fechas: 15 mayo 2023 – 23 mayo 2023.

■ **Planificación del *sprint***

En la reunión de planificación del *sprint* se fijaron las siguientes tareas:

1. Algunos cambios en la gestión de cursos.
2. Creación de la gestión de grupos.
3. Pequeños cambios/mejoras en el código.
4. Avanzar en la memoria.

■ ***Burndown Report***

Para la tarea «Algunos cambios en la gestión de cursos» se fijaron 4 horas. Esta tarea tenía como propósito realizar los cambios vistos en la reunión de revisión del *sprint* anterior. Estos cambios llevaron unas 3 horas de trabajo, pero debido a algunos errores al realizar los cambios se consumieron las 4 horas planificadas.

La segunda tarea realizada fue «Creación de la gestión de grupos». En esta tarea se quería crear todo lo relacionado con la gestión de grupos. Desde su visualización en una tabla, a la creación y eliminación de los mismos para cada asignatura de un curso académico. Para esta tarea se dio una estimación de 8 horas que en trabajo real fueron algo menos, pero cercano a las 8 horas.

La tercera tarea de «Pequeños cambios/mejoras en el código» tenía una estimación de tiempo de 1 hora y se cumplió en este tiempo. En esta tarea se realizaron algunos cambios menores de diseño, cambios en vistas, algunos pequeños cambios de lógica, etc.

Finalmente, la tarea de «Avanzar en la memoria» tenía una estimación de 3 horas. En esta tarea se pretendían realizar los cambios vistos en la documentación y añadir nueva información a la memoria. La tarea llevó un trabajo real de aproximadamente las 3 horas. Se hicieron avances en la memoria y se añadió en los anexos el diccionario de datos, además de realizar los cambios vistos por toda la documentación.

■ Revisión del *sprint*

En la reunión del *sprint* 9 se estuvieron viendo los avances realizados. Se comentó que podría ser buena idea realizar algunos cambios en la creación de grupos y añadir una vista donde visualizar las titulaciones de un centro y las asignaturas de una titulación. Además, se descubrió un error en la eliminación de grupos y se estuvo hablando sobre como realizar el inicio de sesión de la aplicación a través de un *token* devuelto por Moodle.

***Sprint* 10: Asignación de horas**

Fechas: 23 mayo 2023 – 29 mayo 2023.

■ Planificación del *sprint*

En la reunión de planificación del *sprint* se fijaron las siguientes tareas:

1. Crear la asignación de horas de plazas a grupos.
2. Añadir forma avanzada de creación de grupos.

3. Arreglar la eliminación de grupos.
4. Cambios de campos obligatorios en «Añadir Plaza».
5. Arreglo *bug* de números negativos.
6. Mejoras de navegación en la web.
7. Añadir documentación en los anexos.

■ **Burndown Report**

Para la tarea «Crear la asignación de horas de plazas a grupos» se fijaron 8 horas de trabajo. La realidad fue un trabajo de aproximadamente 12 horas. Este exceso de horas es debido a algunas complicaciones para realizar lo deseado que no se habían tenido en cuenta, pero que se fueron solucionando durante el desarrollo.

Para la tarea «Añadir forma avanzada de creación de grupos» se realizó una estimación de 4 horas. Sin embargo, se dejó la tarea pausada hasta la próxima reunión debido a algunas incompatibilidades con el funcionamiento actual del sistema. En la próxima reunión habrá que decidir si añadir esa funcionalidad cambiando las actuales o descartar la idea.

En la tarea «Arreglar la eliminación de grupos» se pretendía arreglar la reasignación de nombres de grupos al eliminar un grupo teórico. En la creación de esta funcionalidad no se había probado bien y si se eliminaba un grupo teórico al que le pertenecían grupos prácticos, estos no se renombraban y daban lugar a futuros problemas. Para la realización de la tarea se hizo una estimación de 2 horas y se realizó dentro de ese tiempo.

La tarea «Cambios de campos obligatorios en «Añadir Plaza»» fue añadida por los tutores del proyecto al probar la aplicación y decidir realizar algunos cambios en los campos que debían ser obligatorios al crear una nueva plaza. Tenía una previsión de 15 minutos, pero estos pequeños cambios llevaron entre 20 y 30 minutos debido a que hubo que realizar cambios en la base de datos y, para ello, instalar y configurar una biblioteca para migraciones de la base de datos.

La tarea «Arreglo *bug* de números negativos» también fue añadida por los tutores del proyecto al realizar sus pruebas en la web. Algunos campos no se validaban bien y permitían ingresar números negativos en sitios donde esto no debía ser posible. Estos pequeños cambios llevaron aproximadamente 15 minutos.

La tarea «Mejoras de navegación en la web» fue otra de las añadidas por los tutores del proyecto. Se pretendía mejorar la navegación por la web añadiendo botones de volver atrás sin realizar cambios en los formularios. Además, se deseaba añadir una ventana desde donde ver las titulaciones de un centro y las asignaturas de una titulación. Estos cambios tuvieron una estimación de poco más de una hora y, al final, el trabajo real también fue acorde.

Finalmente, para la tarea de «Añadir documentación en los anexos» se realizó una estimación de 3 horas. Debido a falta de tiempo apenas se pudo avanzar y el trabajo real se quedó en aproximadamente media hora.

■ Revisión del *sprint*

En la reunión del *sprint* 10 se estuvieron viendo todos los cambios realizados. En cuanto a la asignación de horas a plazas se dio el visto bueno, pero se propusieron algunos cambios en la visualización de los datos y se estudió la posibilidad de actualizar alguna información de la tabla mediante *Ajax* al realizar una actualización sobre la misma.

Otro de los aspectos que fueron claves en la reunión fue la creación de grupos de forma avanzada indicando el nombre del grupo. Se llegó a la conclusión de que es una característica que podría ser interesante en un futuro aunque no primordial. Además, de la forma que se había hecho el diseño era difícil adaptarlo y se decidió dejarlo como una característica interesante para próximas versiones de la aplicación.

***Sprint* 11: Pequeños cambios y arreglos**

Fechas: 29 mayo 2023 – 5 junio 2023.

■ Planificación del *sprint*

En la reunión de planificación del *sprint* se fijaron las siguientes tareas:

1. Arreglo de *bug* de curso duplicado.
2. Algunos cambios en la visualización de las horas de plazas en grupos.
3. Crear el duplicado de la asignación de horas a grupos al duplicar el curso.
4. Arreglo de *bug* de dos plazas con el mismo docente.
5. Creación de *login*.

- ***Burndown Report***

Para la tarea «Arreglo de *bug* de curso duplicado» se fijaron 30 minutos de trabajo. La realidad fue un trabajo de aproximadamente el mismo tiempo. Fue un trabajo sencillo ya que principalmente se basa en comprobar si ya existía un curso con ese año y no permitir crear otro con el mismo año.

La tarea «Algunos cambios en la visualización de las horas de plazas en grupos» tenía una estimación de 4 horas. En esta tarea se pretendía realizar varios cambios en la tabla general de horas sobre como se mostraba la información de los docentes y también cambios dentro de la propia ventana de las plazas asignadas a una asignatura. El trabajo real fue de 2 horas ya que llevó menos trabajo del pensado en un inicio.

En cuanto a la tarea de «Crear el duplicado de la asignación de horas a grupos al duplicar el curso» se puso una estimación de 2 horas. En esta tarea se pretendía implementar en el duplicado de un curso académico la opción de duplicar también las asignaciones de horas de plazas a las asignaturas del curso. La tarea llevó aproximadamente las 2 horas estimadas para su realización.

La tarea «Arreglo de *bug* de dos plazas con el mismo docente» se estimó en una duración de media hora ya que tan solo era arreglar un pequeño fallo de funcionamiento en el que a un docente se le podía asignar más de una plaza sin que la plaza tuviese fecha de cese. La idea del funcionamiento es que un docente pueda tener como máximo una plaza sin fecha de cese, es decir, una plaza activa al mismo tiempo. La duración real fue de poco más de la media hora marcada inicialmente.

Por último, para la tarea «Creación de *login*» se puso una estimación de 3 horas. En esta tarea se implementó el *login* de la aplicación, utilizando para ello el inicio de sesión contra el Moodle de la UBU, el cual devuelve un *token* en caso de inicio de sesión correcto. Para el inicio de sesión primero se debe verificar que el usuario exista en esta aplicación, si no, por mucho que exista en Moodle, no dejará entrar a la aplicación web. Además de la creación del *login* se tuvo que implementar la gestión de permisos en todas las acciones de la web para verificar si el usuario estaba identificado o no, y de esta forma dejarle manejar la web o redirigir a la ventana de inicio de sesión.

- **Revisión del *sprint***

En la reunión del *sprint* 11 se dio el visto bueno a los cambios realizados, pero se estuvieron viendo algunos cambios que podrían mejorar la visualización de las horas asignadas a plazas.

También se explicó cómo debían funcionar los permisos en la aplicación web diferenciando de usuarios con permisos de modificación o permisos de lectura. Además de la posibilidad de tener usuarios sin ningún tipo de permiso, caso que se dará cuando se de un docente de baja.

También se vieron diferentes mejoras que se podrían añadir si daba tiempo como un botón para exportar e importar la información de la base de datos, la creación de gráficos con los datos de las horas de asignaturas, etc.

***Sprint* 12: Últimos cambios aplicación**

Fechas: 5 junio 2023 – 12 junio 2023.

■ **Planificación del *sprint***

En la reunión de planificación del *sprint* se fijaron las siguientes tareas:

1. Cambios en el login y creación de campos de lectura y escritura en docente.
2. Cambios visuales en el apartado de horas.
3. Creación de opción para importar y exportar la base de datos.
4. Arreglo de *bug* con los permisos de usuario.
5. Pequeños cambios de funcionamiento general.
6. Avanzar documentación.

■ ***Burndown Report***

La tarea «Cambios en el login y creación de campos de lectura y escritura en docente» tenía una estimación de 3 horas. En esta tarea lo que se quería hacer era la forma de hacer el *login* y, además, añadir dos campos en la tabla docente para identificar si se tienen permisos de lectura, modificación o ambos, y así mostrar diferentes elementos o vistas según los permisos. Este trabajo llevó finalmente 5 horas, algo más de lo previsto debido a algunos problemas con la forma de implementar los permisos que se vio que no era la mejor forma y se decidió cambiar de nuevo por una forma más eficiente. Además, inicialmente se guardaban el *token* y el id del docente en el lado del cliente, cosa que carece en gran medida de seguridad, ya que robar una

cookie es algo relativamente sencillo y esto podría provocar el acceso a la aplicación por usuarios no deseados. Para solucionar esto, se hizo un cambio para almacenar los datos de la sesión en la base de datos. Además, los datos se guardan cifrados, lo que añade muchas más seguridad de esta información sensible.

Para la tarea de «Cambios visuales en el apartado de horas» se dió una estimación de 1 hora y el trabajo se realizó en ese tiempo.

En cuanto a la tarea «Creación de opción para importar y exportar la base de datos» fue una de las más complejas. La estimación inicial fue de 3 horas ya que se pensó que iba a ser una tarea relativamente sencilla. La tarea pretendía añadir una nueva funcionalidad a la web en la que se permite exportar el contenido de la base de datos e importar el contenido descargado previamente. Esta funcionalidad es interesante pensando en la importación sencilla de los datos cuando se cree una nueva implantación de la web.

En un principio se pensó que sería una tarea sencilla, pero a la hora de la verdad fue una tarea complicada ya que la conexión y lectura de la base de datos no era algo tan sencillo. Había que controlar los diferentes tipos de datos y solventar algunos errores con la tabla de sesiones que contaba con información en binario y que se pensó que lo mejor era no exportar esa información ya que no es relevante.

De cara a la importación también llevó mucho tiempo debido a estudiar los diferentes ataques que se podían producir con esta funcionalidad, tanto en la subida del archivo a la web, como en las sentencias incluidas en el SQL. Finalmente se decidió permitir únicamente las sentencias *insert*, ya que la idea de la funcionalidad es únicamente añadir datos a la base de datos, y así evitar borrados o modificaciones no deseadas.

El trabajo final fue de unas 8 horas, bastante más de lo estimado.

Para la tarea «Arreglo de *bug* con los permisos de usuario» se dio una estimación de 20 minutos ya que lo único que se quería hacer era limitar las capacidades de un usuario sin permisos dentro de la aplicación, prohibiéndoles el acceso incluso a la ventana principal. Actualmente, un usuario sin permisos no puede pasar del inicio de sesión, donde se muestra un mensaje indicando la falta de permisos.

La tarea «Pequeños cambios de funcionamiento general» tenía una estimación de 1 hora. Esta tarea estaba diseñada para realizar algunos cambios visuales o de funcionalidad en la web. Así como carga interna de bibliotecas o datos. La duración real de la tarea también fue de 1 hora aproximadamente.

Por último, la tarea de «Avanzar documentación» tenía una estimación de 4 horas de trabajo que quedó en unas 3 horas de trabajo real donde se añadió documentación en los anexos.

■ Revisión del *sprint*

En la reunión del *sprint* 12 se estuvieron viendo los cambios realizados y se decidió realizar algunos cambios a la hora de importar los datos a la base de datos añadiendo una confirmación antes de importar y la visualización de los datos a importar. También se encontró un error en el funcionamiento de las sesiones que debía ser subsanado y por último, se estuvieron viendo los cambios y posibles adiciones que se podían realizar en la documentación.

***Sprint* 13: Avances en la documentación y arreglos**

Fechas: 12 junio 2023 – 19 junio 2023.

■ Planificación del *sprint*

En la reunión de planificación del *sprint* se fijaron las siguientes tareas:

1. Campo de reducciones por defecto a 0.
2. Solucionar bug con las sesiones en la base de datos.
3. Crear carga masiva de datos.
4. Cambios en la función de importar datos.
5. Añadir documentación y solucionar errores vistos.

■ *Burndown Report*

Para la tarea «Campo de reducciones por defecto a 0» se dio una estimación de 10 minutos y se hizo en ese mismo tiempo. Esta tarea consistía un pequeño cambio en poner un valor mínimo de 0 ya marcado en el campo reducciones de el formulario de docentes para hacer más cómoda la introducción de datos.

La tarea «Solucionar bug con las sesiones en la base de datos» tenía una previsión de 3 horas de trabajo que se convirtieron en 8 horas reales. Con esta tarea se buscaba arreglar un bug que hacía que las sesiones no se almacenasen en la base de datos. Al poco de mirarlo se vio que era un problema por el orden de carga de la biblioteca, pero al hacer pruebas se detectó un error entre las sesiones y la sesión de SQLAlchemy que hacía que en las eliminaciones no se esperase al

commit y las hiciese de forma directa. Tras darle muchas vueltas y buscar mucha información sin ningún resultado y revisar que todo estaba correcto, se decidió descartar la idea de utilizar SQLAlchemy para guardar las sesiones porque todo parecía indicar ser un bug de la biblioteca. Para poder seguir almacenando las sesiones en el servidor, se optó por guardarlas en el sistema de ficheros, cosa que en un principio dio problemas en Heroku, pero que se consiguió solucionar.

La tarea «Crear carga masiva de datos» fue propuesta por el tutor de proyecto Álar Arnáiz González para tener datos reales de los centros, titulaciones, asignaturas, etc. En un principio se pensó que con 2 horas de trabajo sería suficiente, pero la realidad fueron unas 6 horas debido a que el formato en el que podíamos obtener los datos no era el mismo que necesitábamos y esto implicaba tener que hacer varias modificaciones.

Para la tarea de «Cambios en la función de importar datos» se dio una estimación de 3 horas que se convirtieron en unas 7 horas de trabajo. Esto es debido a que se tuvo que crear una función de JavaScript que fuese capaz de leer un archivo SQL y extraer de él la información que se iba a introducir en la base de datos mostrándolo en forma de tabla en el HTML. También se creó un mensaje de confirmación antes de importar para evitar fallos y, por último, se modificó la función de Python encargada de la importación para mejorar la validación del fichero subido.

Finalmente, la tarea de «Añadir documentación y solucionar errores vistos» estaba creada para, como su nombre indica, avanzar con la documentación del trabajo y para corregir las cosas vistas en la reunión del *sprint*. Tenía una estimación de 8 horas que se quedó en unas 5 o 6 de trabajo real. No se pudo avanzar más debido a la falta de tiempo ya que otras tareas habían consumido más de lo esperado. La tarea de dejó abierta para continuar con ella en el siguiente *sprint*.

■ Revisión del *sprint*

En la reunión del *sprint* 13 estuvimos viendo algunos pequeños errores vistos por los tutores. También se vieron los cambios realizados y se comentó alguna cosa que podía añadir a la documentación.

***Sprint* 14: Arreglo de errores y documentación de anexos**

Fechas: 19 junio 2023 – 26 junio 2023.

■ Planificación del *sprint*

En la reunión de planificación del sprint se fijaron las siguientes tareas:

1. *Bug*: El campo plazas lleva a horas.
2. No permitir eliminar al docente que está en la sesión.
3. No permitir eliminar titulaciones con asignaturas.
4. Recarga de página tras cambio de grupos.
5. Cambiar el id por el nombre en las *breadcrumbs*.
6. Mejorar interfaz de *login*.
7. *Bug*: En la importación de datos permite cualquier fichero.
8. Añadir documentación en los anexos.

■ *Burndown Report*

Para la tarea «*Bug*: El campo plazas lleva a horas» se dio una estimación de aproximadamente 5 minutos y se realizó dentro del plazo estimado. En esta tarea simplemente había que cambiar el enlace de la opción «plazas» de la página inicial que estaba llevando a la vista de horas y no a la de plazas.

La siguiente tarea, «No permitir eliminar al docente que está en la sesión» tenía como finalidad no permitir que el docente con la cuenta activa pudiese eliminar o cambiar los permisos de su propia cuenta. Para realizar esta tarea se dio una estimación de media hora y se realizó en unos 20 minutos, algo menos de lo previsto.

La tarea de «No permitir eliminar titulaciones con asignaturas» estaba creada para quitar la posibilidad del borrado en cascada de asignaturas dependientes de una titulación al eliminar esta. Para esta tarea se hizo una estimación de 30 minutos que se quedó en la mitad de trabajo real, unos 15 minutos.

Para la tarea «Recarga de página tras cambio de grupos» se puso una estimación de 1 hora, pero finalmente el trabajo real fue de unas 3 horas. Se quería quitar la recarga al modificar los campos de una asignatura de un curso desde la vista principal de «grupos». Se preveía que iba a ser un trabajo menor, pero al tener que cambiar la forma de funcionar a hacer una petición mediante Ajax para evitar la recarga hubo que modificar la función del controlador prácticamente al completo y hubo que crear la función de JavaScript encargada de gestionar esta tarea.

Otra tarea que se realizó fue la de «Cambiar el id por el nombre en las *breadcrumbs*». Se dio una estimación de media hora y se realizó

dentro del plazo. Era una tarea sencilla ya que sólo se debía cambiar el identificador por el nombre, pero al ser un cambio en tantas acciones de distintos controladores se tardó un poco en completar la tarea.

La tarea de «Mejorar interfaz de *login*» pretendía mejorar un poco el diseño de la ventana del *login* haciéndola menos minimalista al añadirle algún contenido como el logo o un pie de página. Esta tarea tenía una estimación de media hora de trabajo, pero el trabajo real se fue hasta aproximadamente las 3 horas. Esto es debido a que surgió un problema con el pie de página que no se mostraba al final de la página y hubo que estudiar como hacer que se mostrase abajo del todo aunque no hubiese contenido suficiente en la página para rellenar hasta ahí. Además, al probar en un dispositivo móvil se detecto que no se mostraba del todo bien así que se cambió el formato completo del *login* para que se mostrase bien en dispositivos con pantallas más pequeñas.

Para la tarea «*Bug*: En la importación de datos permite cualquier fichero» se fue muy optimista ya que se estimó en 1 hora y finalmente fueron unas 4 horas de trabajo. En esta tarea se quería controlar mejor los archivos que se subían para hacer una importación en la base de datos ya que se detectó que era posible seleccionar cualquier fichero, aunque en el servidor si se comprobaba si era SQL y se permitía o no. La idea era que mediante JavaScript se hiciese una primera validación para directamente no dejar enviar el archivo y no mostrar la visualización.

Otro de los errores detectados era que si se cambiaba el formato a otro archivo, por ejemplo convertir un PDF en un archivo SQL, se permitía subir pero saltaba una excepción al intentar ejecutar la importación. Tras investigar se consiguió hacer una primera validación desde JavaScript para detectar estos archivos SQL mal formados y no permitir subirlos. Además, en Python también se controló el posible error que podía saltar.

Finalmente para la tarea «Añadir documentación en los anexos» se propuso trabajar 6 horas y al final se hicieron.....

■ Revisión del *sprint*

En la reunión del *sprint* 14.....

A.3. Estudio de viabilidad

El estudio de viabilidad es una etapa esencial en el desarrollo de cualquier proyecto, ya que permite evaluar la viabilidad económica, técnica y operativa de la propuesta.

En este apartado, se abordará el estudio de viabilidad del proyecto realizado. Se analizarán los aspectos económicos y legales del proyecto con el objetivo de determinar su viabilidad y establecer las bases para su exitosa implementación.

Viabilidad económica

Para determinar la viabilidad económica del proyecto es necesario analizar los costes que tiene y los posibles beneficios que puede otorgar.

Coste de personal

Para tener en cuenta los costes del proyecto, lo primero en lo que nos debemos fijar es el coste del personal involucrado en el proyecto realizado.

En este caso, ha habido un único desarrollador *junior* trabajando poco más de 4 meses dedicando aproximadamente 350 horas de trabajo en este periodo.

Se toma como salario medio de un programador *junior* los 19000 € brutos anuales². Suponiendo un trabajo de 1826 horas anuales³, se obtiene un precio de 10,40 € por cada hora de trabajo. Teniendo en cuenta las 350 horas trabajadas * $10,40 \frac{\text{€}}{\text{hora}} = 3640 \text{ €}$ totales, lo que es lo mismo que 910 € mensuales brutos.

Una vez calculado el salario es importante tener en cuenta que la empresa debe pagar impuestos adicionales a esta base. En primer lugar debe pagar un 23.60 % de la base como contingencias comunes, 5,50 % de desempleo, un 0,6 % como formación, un 0,2 % de cotización al Fondo de Garantía Salarial (FOGASA) y un 1,5 % por accidentes de trabajo y enfermedades profesionales⁴.

El total de lo que debe pagar la empresa por el salario mensual del trabajador se puede ver en la ecuación A.1.

²https://www.glassdoor.es/Sueldos/programador-junior-sueldo-SRCH_K00,18.htm

³<https://asesorias.com/empresas/normativas/laboral/jornada/horas-trabajo-anuales/>

⁴<https://getquipu.com/blog/cuanto-cuesta-contratar-un-trabajador/>

$$\frac{910 \frac{\text{€}}{\text{mes}}}{1 - (0,236 + 0,055 + 0,006 + 0,002 + 0,0150)} = 1326,53 \frac{\text{€}}{\text{mes}} \quad (\text{A.1})$$

Finalmente, el salario durante los 4 meses de trabajo asciende al total de $1326,53 \frac{\text{€}}{\text{mes}} * 4 \text{ meses} = 5306,12 \text{ €}$.

Coste de *hardware*

Para el desarrollo del proyecto es necesario contar con cierto *hardware*. En este caso, se ha utilizado un ordenador con un procesador Intel Core i5-8600, 16 GB de memoria RAM, tarjeta gráfica NVIDIA GeForce GTX 1060 y 1,5TB de almacenamiento. Es un ordenador tiene un valor de aproximado de 900€. En cuanto a periféricos y pantalla se estima un precio de 190 €.

Este conjunto de *hardware* se amortizará en 4 años. El coste anual de amortización se puede ver en la ecuación A.2.

$$\frac{1090 \text{ €}}{4 \text{ años}} = 272,5 \frac{\text{€}}{\text{año}} \quad (\text{A.2})$$

Teniendo en cuenta que la duración del proyecto ha sido de 4 meses, el coste durante estos meses del *hardware* es de 90,83 €.

Por último, sobre el servidor donde desplegar la aplicación, pondremos un precio estimado de unos 80 € mensuales para un servidor dedicado⁵. Por lo tanto, el coste durante estos 4 meses es de 320 €.

Coste de *software*

En este apartado vamos a exponer el coste del *software* utilizado para el desarrollo del proyecto.

Para la realización de este proyecto se han utilizado bibliotecas gratuitas. El único coste adicional que puede tener es el del uso del IDE PyCharm, el cual tiene un coste mensual de 24,90 € en su versión profesional, por lo que el coste durante los 4 meses de utilización es de 99,60 €.

Recuento

El recuento de estos costes durante los 4 meses de desarrollo del proyecto se puede ver en la tabla A.1

⁵<https://www.ovhcloud.com/es-es/bare-metal/prices/>

Concepto	Coste (€)
Personal	5306,12
<i>Hardware</i>	410,83
<i>Software</i>	99,60
Total	5816,55

Tabla A.1: Total de costes durante el desarrollo del proyecto.

Beneficios

Este proyecto ha sido desarrollado para ser implementado en un departamento de la Universidad de Burgos sin obtener de ello ningún tipo de beneficio, ya que se trata de una aplicación interna de la propia universidad realizada con carácter educativo.

Viabilidad legal

La viabilidad legal de un proyecto permite evaluar y garantizar el cumplimiento de las leyes y regulaciones vigentes, tanto en el desarrollo como en el uso de la aplicación.

Licencias del software utilizado

En la tabla A.2 se puede ver el listado de bibliotecas y herramientas utilizadas junto a su licencia.

La explicación de las licencias que utilizan las bibliotecas se encuentran expuestas a continuación:

- **MIT License**

Licencia de *software* de código abierto que permite a los usuarios utilizar, copiar, modificar, fusionar, publicar, distribuir y sublicenciar el software bajo ciertas condiciones. Es una licencia flexible y no impone muchas restricciones [4].

- **BSD License**

Son varias licencias de *software* de código abierto con diferentes variantes, como la Licencia BSD de 3 cláusulas y la Licencia BSD de 2 cláusulas. Estas licencias permiten el uso, la redistribución y la modificación del software bajo ciertas condiciones. Se suelen considerar licencias permisivas [3].

Biblioteca/herramientas	Versión	Licencia
alembic	1.11.1	MIT License
asgiref	3.6.0	BSD-2 License
cachelib	0.10.2	MIT License
click	8.1.3	BSD-3 License
colorama	0.4.6	BSD-3 License
Django	4.2.1	BSD-3 License
djangoApiDec	1.7	MIT License
dnspython	2.3.0	ISC License
email-validator	2.0.0.post2	MIT License
Flask	2.2.3	BSD-3 License
Flask-Menu	0.7.2	BSD License
Flask-Migrate	4.0.4	BSD License
Flask-Session	0.5.0	BSD License
Flask-SQLAlchemy	3.0.3	BSD-3 License
Flask-WTF	1.1.1	BSD-3 License
greenlet	2.0.2	MIT License
gunicorn	20.1.0	MIT License
idna	3.4	BSD License
importlib-metadata	6.1.0	Apache License 2.0
itsdangerous	2.1.2	BSD License
jieba	0.38	MIT License
Jinja2	3.1.2	BSD-3 License
Mako	1.2.4	MIT License
MarkupSafe	2.1.2	BSD License
mysqlclient	2.1.1	GNU Library General Public License
ngram	4.0.3	Apache License 2.0
pymongo	3.4.0	Apache License 2.0
PyMySQL	1.0.3	MIT License
PyPrind	2.9.9	MIT License
requests	2.12.3	Apache License 2.0
simplejson	3.10.0	MIT License
six	1.16.0	MIT License
SQLAlchemy	2.0.9	MIT License
sqlparse	0.4.4	BSD License
typing_extensions	4.5.0	MIT License
tzdata	2023.3	MIT License
Werkzeug	2.2.3	BSD License
WTForms	3.0.1	BSD License
zipp	3.15.0	MIT License
grid.js	6.0.6	MIT License
select2	4.1.0-rc.0	MIT License
sortablejs	1.15.0	MIT License

Tabla A.2: Licencias de las bibliotecas utilizadas.

- **Apache License 2.0**

Licencia de *software* de código abierto permisiva. Permite a los usuarios utilizar, copiar, modificar, distribuir y sublicenciar el software bajo ciertas condiciones. Además, esta licencia incluye disposiciones específicas para las contribuciones de patentes [1].

- **GNU Library General Public License (LGPL)**

Licencia de *software* de código abierto. Esta licencia permite que el software se utilice y modifique, incluso en proyectos propietarios, siempre y cuando se respeten ciertas condiciones. Sin embargo, cualquier modificación realizada en el software LGPL debe estar disponible públicamente bajo la misma licencia LGPL [2].

- **ISC License**

Licencia de *software* de código abierto permisiva funcionalmente equivalente a las licencias BSD y MIT [5].



Apéndice B

Especificación de Requisitos

B.1. Introducción

Para comenzar con el desarrollo de la aplicación web sobre la que se basa este proyecto, se debe realizar un estudio de los requisitos que son necesarios cubrir. Con estos requisitos se podrán detectar las funcionalidades necesarias y se podrán generar los casos de uso de la aplicación.

Este es el punto de partida para realizar un buen diseño de la aplicación que facilitará las siguientes tareas a realizar.

B.2. Objetivos generales

El objetivo de este proyecto es desarrollar una aplicación web para poder llevar a cabo la gestión de las diferentes áreas de la Universidad de Burgos. El sistema gestionará el profesorado, asignaturas, reconocimiento de docencia...

B.3. Catálogo de requisitos

A continuación se van a exponer los requisitos de la aplicación web.

Requisitos funcionales

1. **RF-01.** Mantenimiento de titulaciones.
2. **RF-02.** Mantenimiento de asignaturas.

3. **RF-03.** Mantenimiento de grupos.
4. **RF-04.** Mantenimiento de docentes.
5. **RF-05.** Mantenimiento de centros.
6. **RF-06.** Mantenimiento de cursos académicos.
7. **RF-07.** Mantenimiento de plazas.
8. **RF-08.** Mantenimiento de tipos de contrato.
9. **RF-09.** Mantenimiento de áreas.
10. **RF-10.** Mantenimiento de departamentos.
11. **RF-11.** Un administrativo puede asignar horas a un docente (plaza) en un grupo de un curso.
12. **RF-12.** Un administrativo puede asignar una plaza a un docente.
13. **RF-13.** Un grupo puede ser asignado a una asignatura en un curso.

Requisitos no funcionales

- **RNF-01.** El sistema debe ser fácil de usar.
- **RNF-02.** El sistema no debe permitir el acceso no autorizado.

B.4. Especificación de requisitos

Casos de uso

A continuación se van a exponer los casos de uso de la aplicación separándolos en tres casos de uso principales para una mejor visualización y mayor organización. Estos casos de uso generales se pueden ver en la figura B.1.

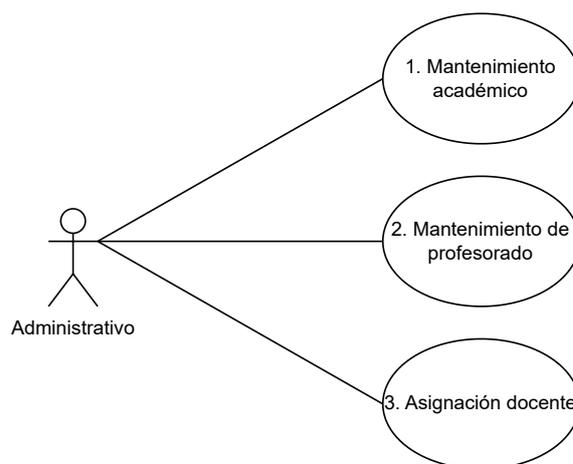


Figura B.1: Diagrama de casos de uso general

1. Mantenimiento académico

En este apartado se va a exponer el diagrama de casos de uso 2, el cual se encuentra definido en la ilustración B.2 y en las tablas B.2, B.3, B.4, B.5, B.6 y B.7.

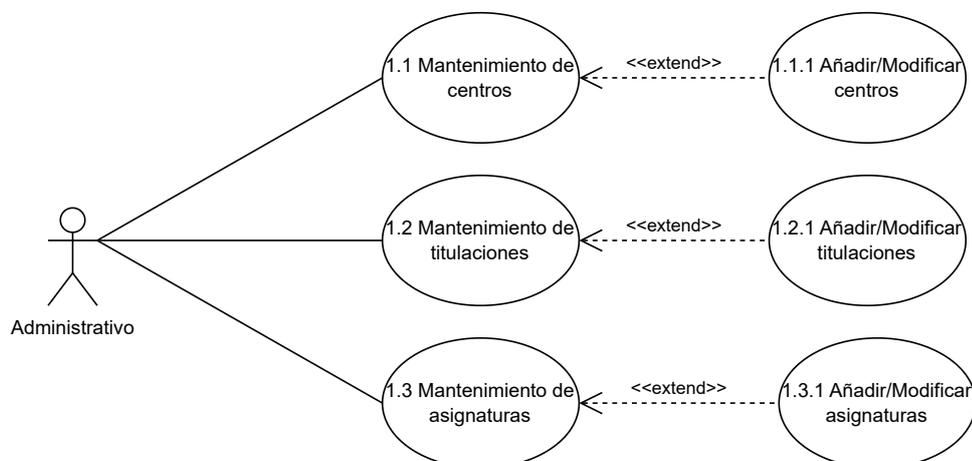


Figura B.2: Diagrama de caso de uso - 1. Mantenimiento académico

CU-1	Mantenimiento académico
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-05, RF-01, RF-02
Descripción	Un administrativo puede realizar labores de mantenimiento de los centros, titulaciones y asignaturas
Precondición	Tener iniciada sesión con una cuenta con permisos administrativos
Acciones	<ol style="list-style-type: none"> 1. Seleccionar la opción «Asignaturas» del menú principal de la web. 2. Se abre la ventana de mantenimiento de las asignaturas.
Postcondición	Ninguna
Excepciones	También es posible acceder la ventana de mantenimiento de centros y titulaciones seleccionando las opciones «Centros» (CU-1.1) o «Titulaciones» (CU-1.2) respectivamente. Si no se pulsa ninguna, se permanece en la ventana actual.
Importancia	Alta

Tabla B.1: CU-1 Mantenimiento académico.

CU-1.1	Mantenimiento de centros
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-05
Descripción	Un administrativo puede realizar el mantenimiento de los centros
Precondición	Realizar el CU-1
Acciones	<ol style="list-style-type: none"> 1. Se abre una ventana donde aparece una tabla con los centros creados desde donde se podrá realizar el mantenimiento.
Postcondición	Ninguna
Excepciones	Si se pulsa sobre el botón «Nuevo», se accede a la creación de centros (CU-1.1.1). Si se pulsa sobre el botón «Modificar» de un centro del listado, se accede a la ventana de modificación del centro. Por último, si se pulsa sobre el botón «Eliminar» de un centro el sistema pregunta si está seguro, y al pulsar en «Sí», este se elimina produciendo un borrado en cascada de las titulaciones vinculadas.
Importancia	Alta

Tabla B.2: CU-1.1 Mantenimiento de centros.

CU-1.1.1	Añadir/Modificar centros
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-05
Descripción	Un administrativo añade o modifica un nuevo centro
Precondición	Realizar el CU-1.1
Acciones	<ol style="list-style-type: none"> 1. Se abre una ventana con un formulario vacío donde aparecen los campos de la tabla «Centro» de la figura B.3, necesarios para crear un centro. 2. Rellenar el formulario con los datos del centro que se desea añadir. 3. Pulsar sobre el botón «Añadir».
Postcondición	El centro queda añadido/modificado y el sistema lleva al usuario a la ventana de centros donde se puede ver el listado de todos los centros creados.
Excepciones	Se dejan campos vacíos o se introducen datos con un formato incorrecto. Otra forma de finalizar el caso de uso es la modificación. En este caso, se pulsa sobre el botón «Modificar» de un centro y se accede al mismo formulario con los campos rellenos. Finalmente se pulsa en el botón «Modificar»
Importancia	Alta

Tabla B.3: CU-1.1.1 Añadir/Modificar centros.

CU-1.2	Mantenimiento de titulaciones
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-01
Descripción	Un administrativo puede realizar el mantenimiento de las titulaciones
Precondición	Realizar el CU-1
Acciones	<ol style="list-style-type: none"> 1. Se abre una ventana donde aparece una tabla con las titulaciones creadas desde donde se podrá realizar el mantenimiento.
Postcondición	Ninguna
Excepciones	Si se pulsa sobre el botón «Nuevo», se accede a la creación de titulaciones (CU-1.2.1). Si se pulsa sobre el botón «Modificar» de una titulación de la lista se accede a la ventana de modificación. Por último, si se pulsa sobre el botón «Eliminar» de una titulación el sistema pregunta si está seguro, y al pulsar en «Sí», esta se elimina produciendo un borrado en cascada de las asignaturas vinculadas.
Importancia	Alta

Tabla B.4: CU-1.2 Mantenimiento de titulaciones.

CU-1.2.1		Añadir/Modificar titulaciones	
Versión		1.0	
Autor		Ignacio Dávila García	
Requisitos asociados		RF-01	
Descripción		Un administrativo añade o modifica una titulación	
Precondición		Realizar el CU-1.2 y tener algún centro creado	
Acciones		<ol style="list-style-type: none"> 1. Se abre una ventana con un formulario vacío donde aparecen los campos de la tabla «Titulación» de la figura B.3, necesarios para crear una titulación. También aparece el campo «Centro» para seleccionar el centro al que pertenece. 2. Rellenar el formulario con los datos de la titulación que se desea añadir. 3. Pulsar sobre el botón «Añadir». 	
Postcondición		La titulación queda añadida/modificada y el sistema lleva al usuario a la ventana de titulaciones donde se puede ver el listado de todas las titulaciones añadidas.	
Excepciones		Se dejan campos vacíos o se introducen datos con un formato incorrecto. Otra forma de finalizar el caso de uso es la modificación. En este caso, se pulsa sobre el botón «Modificar» de una titulación de la tabla y se accede al mismo formulario, pero con los campos rellenos. Finalmente se pulsa en el botón «Modificar» y la titulación queda modificada.	
Importancia		Alta	

Tabla B.5: CU-1.2.1 Añadir/Modificar centros.

CU-1.3	Mantenimiento de asignaturas
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-02
Descripción	Un administrativo puede realizar el mantenimiento de asignaturas
Precondición	Realizar el CU-1
Acciones	<ol style="list-style-type: none"> 1. Se abre una ventana donde aparece una tabla con las asignaturas creadas desde donde se podrá realizar el mantenimiento.
Postcondición	Ninguna
Excepciones	Si se pulsa sobre el botón «Nuevo», se accede a la creación de asignaturas (CU-1.3.1). Si se pulsa sobre el botón «Modificar» de una asignaturas de la lista se accede a la ventana de modificación. Por último, si se pulsa sobre el botón «Eliminar» de una asignatura el sistema pregunta si está seguro, y al pulsar en «Sí» esta se elimina.
Importancia	Alta

Tabla B.6: CU-1.3 Mantenimiento de asignaturas.

CU-1.3.1		Añadir/Modificar asignaturas
Versión		1.0
Autor		Ignacio Dávila García
Requisitos asociados	aso-	RF-02
Descripción		Un administrativo añade o modifica una asignatura
Precondición		Realizar el CU-1.3 y tener alguna titulación creada
Acciones		<ol style="list-style-type: none"> 1. Se abre una ventana con un formulario vacío donde aparecen los campos de la tabla «Asignatura» de la figura B.3, necesarios para crear una asignatura. También se debe indicar la titulación a la que pertenece y rellenar el campo «Abreviatura» que permite la selección múltiple de abreviaturas separadas por comas. Las abreviaturas existentes aparecen al escribir y se pueden seleccionar. Si se escribe una nueva se almacena. 2. Rellenar el formulario con los datos de la asignatura que se desea añadir. 3. Pulsar sobre el botón «Añadir».
Postcondición		La asignatura queda añadida/modificada y el sistema lleva al usuario a la ventana de asignaturas donde se puede ver el listado de todas las asignaturas añadidas.
Excepciones		Se dejan campos vacíos, se introducen datos con un formato incorrecto o se introduce una id existente. Otra forma de finalizar el caso de uso es la modificación. En este caso, se pulsa sobre el botón «Modificar» de una asignatura de la tabla y se accede al mismo formulario, pero con los campos rellenos. Finalmente se pulsa en el botón «Modificar» y la asignatura queda modificada.
Importancia		Alta

Tabla B.7: CU-1.3.1 Añadir/Modificar asignaturas.

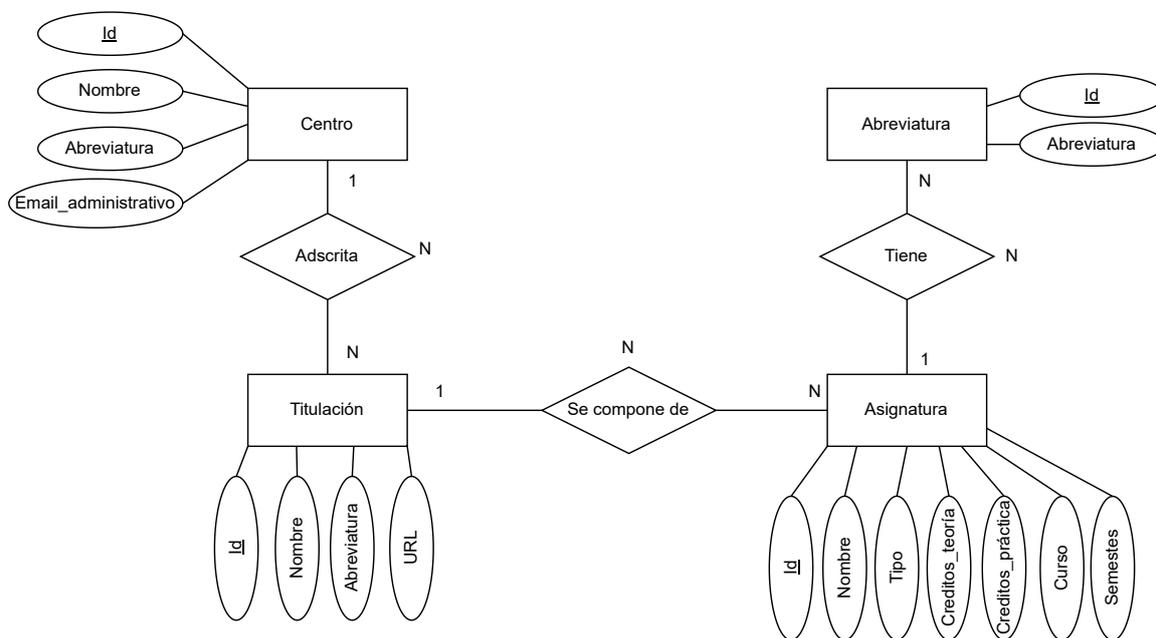


Figura B.3: Vista diagrama entidad relación para el CU-1

2. Mantenimiento de profesorado

En este apartado se va a exponer el diagrama de casos de uso 2, el cual se encuentra definido en la ilustración B.4 y en las tablas B.8, B.10, B.11, B.12, B.13, B.14, B.15, B.16, B.17, B.18 y B.19.

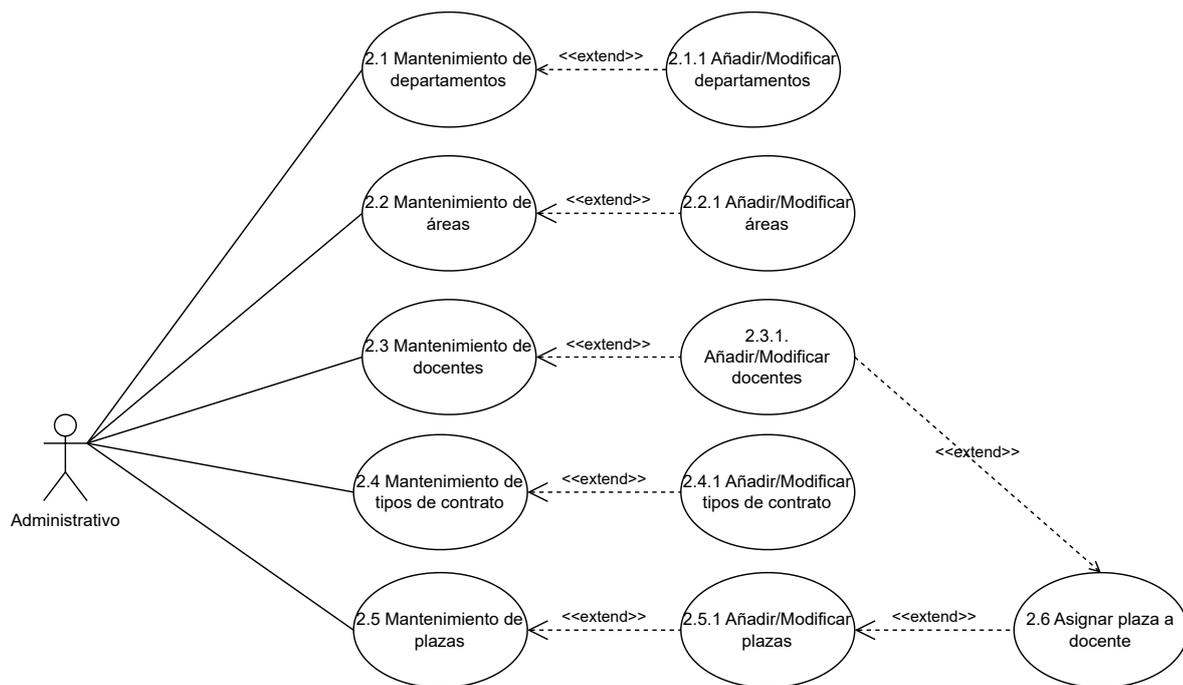


Figura B.4: Diagrama de caso de uso - 2. Mantenimiento de profesorado

CU-2	Mantenimiento de profesorado
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-04, RF-07, RF-08, RF-09, RF-10
Descripción	Un administrativo puede realizar labores de mantenimiento de los docentes, tipos de contrato, departamentos, áreas y plazas. Además, un administrativo puede asignar una plaza a un docente.
Precondición	Tener iniciada sesión con una cuenta con permisos administrativos
Acciones	<ol style="list-style-type: none"> 1. Seleccionar la opción «Plazas» del menú principal de la web. 2. Se abre la ventana de mantenimiento de plazas.
Postcondición	El sistema lleva al usuario a la ventana de la opción pulsada.
Excepciones	Otras opciones que se pueden seleccionar son «Docentes» (CU-2.3), «Contratos» (CU-2.4), «Departamentos» (CU-2.1) o «Áreas» (CU-2.2). Al seleccionar alguna de estas opciones el sistema lleva al usuario a la ventana de mantenimiento de la opción elegida.
Importancia	Alta

Tabla B.8: CU-2 Mantenimiento de profesorado.

CU-2.1		Mantenimiento de departamentos	
Versión		1.0	
Autor		Ignacio Dávila García	
Requisitos asociados		RF-10	
Descripción		Un administrativo puede realizar el mantenimiento de departamentos	
Precondición		Realizar el CU-2	
Acciones		<ol style="list-style-type: none"> 1. Se abre una ventana donde aparece una tabla con los departamentos creados desde donde se podrá realizar el mantenimiento. 	
Postcondición		Ninguna	
Excepciones		Si se pulsa sobre el botón «Nuevo», se accede a la creación de departamentos (CU-2.1.1). Si se pulsa sobre el botón «Modificar» de un departamento de la lista, se accede a la ventana de modificación. Por último, si se pulsa sobre el botón «Eliminar» de un departamento el sistema pregunta si está seguro, y al pulsar en «Sí», este se elimina produciendo un borrado en cascada de las áreas asociadas.	
Importancia		Alta	

Tabla B.9: CU-2.1 Mantenimiento de departamentos.

CU-2.1.1	Añadir/Modificar departamentos
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-10
Descripción	Un administrativo añade o modifica un departamento
Precondición	Realizar el CU-2.1
Acciones	<ol style="list-style-type: none"> 1. Se abre una ventana con un formulario vacío donde aparecen los campos de la tabla «Departamento» de la figura B.5, necesarios para crear un departamento. 2. Rellenar el formulario con los datos del departamento que se desea añadir. 3. Pulsar sobre el botón «Añadir».
Postcondición	El departamento queda añadido/modificado y el sistema lleva al usuario a la ventana de departamentos donde se puede ver el listado de todos los departamentos añadidos.
Excepciones	Se dejan campos vacíos o se introducen datos con un formato incorrecto. Otra forma de finalizar el caso de uso es la modificación. En este caso, se pulsa sobre el botón «Modificar» de un departamento de la tabla y se accede al mismo formulario, pero con los campos rellenos. Finalmente se pulsa en el botón «Modificar» y el departamento queda modificado.
Importancia	Alta

Tabla B.10: CU-2.1.1 Añadir/Modificar departamentos.

CU-2.2		Mantenimiento de áreas
Versión		1.0
Autor		Ignacio Dávila García
Requisitos asociados		RF-09
Descripción		Un administrativo puede realizar el mantenimiento de áreas
Precondición		Realizar el CU-2
Acciones		<ol style="list-style-type: none"> 1. Se abre una ventana donde aparece una tabla con las áreas creadas desde donde se podrá realizar el mantenimiento.
Postcondición		Ninguna
Excepciones		Si se pulsa sobre el botón «Nuevo», se accede a la creación de áreas (CU-2.2.1). Si se pulsa sobre el botón «Modificar» de un área de la lista, se accede a la ventana de modificación. Por último, si se pulsa sobre el botón «Eliminar» de un área el sistema pregunta si está seguro, y al pulsar en «Sí», esta se elimina produciendo un borrado en cascada de las plazas asociadas.
Importancia		Alta

Tabla B.11: CU-2.2 Mantenimiento de áreas.

CU-2.2.1	Añadir/Modificar áreas
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-09
Descripción	Un administrativo añade o modifica un área
Precondición	Realizar el CU-2 y tener algún departamento creado
Acciones	<ol style="list-style-type: none"> 1. Se abre una ventana con un formulario vacío donde aparecen los campos de la tabla «Área» de la figura B.5, necesarios para crear un área. También se debe indicar el departamento al que pertenece el área. 2. Rellenar el formulario con los datos del área que se desea añadir. 3. Pulsar sobre el botón «Añadir».
Postcondición	El área queda añadido/modificado y el sistema lleva al usuario a la ventana de áreas donde se puede ver el listado de todas las áreas añadidas.
Excepciones	Se dejan campos vacíos o se introducen datos con un formato incorrecto. Otra forma de finalizar el caso de uso es la modificación. En este caso, se pulsa sobre el botón «Modificar» de un área de la tabla y se accede al mismo formulario, pero con los campos rellenos. Finalmente se pulsa en el botón «Modificar» y el área queda modificada.
Importancia	Alta

Tabla B.12: CU-2.2.1 Añadir/Modificar áreas.

CU-2.3	Mantenimiento de docentes
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-04
Descripción	Un administrativo puede realizar el mantenimiento de docentes
Precondición	Realizar el CU-2
Acciones	<ol style="list-style-type: none"> 1. Seleccionar la opción «Docentes» del menú principal de la web. 2. Se abre una ventana donde aparece una tabla con los docentes creados desde donde se podrá realizar el mantenimiento.
Postcondición	Ninguna
Excepciones	Si se pulsa sobre el botón «Nuevo», se accede a la creación de docentes (CU-2.3.1). Si se pulsa sobre el botón «Modificar» de un docente de la lista, se accede a la ventana de modificación. Por último, si se pulsa sobre el botón «Eliminar» de un docente el sistema pregunta si está seguro, y al pulsar en «Sí», este se elimina.
Importancia	Alta

Tabla B.13: CU-2.3 Mantenimiento de docentes.

CU-2.3.1	Añadir/Modificar docentes
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-04
Descripción	Un administrativo añade o modifica un docente
Precondición	Realizar el CU-2.3
Acciones	<ol style="list-style-type: none"> 1. Se abre una ventana con un formulario vacío donde aparecen los campos de la tabla «Docente» de la figura B.5, necesarios para crear un docente. 2. Rellenar el formulario con los datos del docente que se desea añadir. 3. Pulsar sobre el botón «Añadir».
Postcondición	El docente queda añadido/modificado y el sistema lleva al usuario a la ventana de docentes donde se puede ver el listado de todos los docentes añadidos.
Excepciones	Se dejan campos vacíos o se introducen datos con un formato incorrecto. Otra forma de finalizar el caso de uso es la modificación. En este caso, se pulsa sobre el botón «Modificar» de un docente de la tabla y se accede al mismo formulario, pero con los campos rellenos. Finalmente se pulsa en el botón «Modificar» y el docente queda modificado.
Importancia	Alta

Tabla B.14: CU-2.3.1 Añadir/Modificar docentes.

CU-2.4	Mantenimiento de tipos de contrato
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-08
Descripción	Un administrativo puede realizar el mantenimiento de tipos de contrato
Precondición	Realizar el CU-2
Acciones	<ol style="list-style-type: none"> 1. Seleccionar la opción «Contratos» del menú principal de la web. 2. Se abre una ventana donde aparece una tabla con los tipos de contrato creados desde donde se podrá realizar el mantenimiento.
Postcondición	Ninguna
Excepciones	Si se pulsa sobre el botón «Nuevo», se accede a la creación de un nuevo tipo de contrato (CU-2.4.1). Si se pulsa sobre el botón «Modificar» de un tipo de contrato de la lista, se accede a la ventana de modificación. Por último, si se pulsa sobre el botón «Eliminar» de un tipo de contrato el sistema pregunta si está seguro, y al pulsar en «Sí», este se elimina.
Importancia	Alta

Tabla B.15: CU-2.4 Mantenimiento de tipos de contrato.

CU-2.4.1	Añadir/Modificar tipos de contrato
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-08
Descripción	Un administrativo añade o modifica un docente
Precondición	Realizar el CU-2.4
Acciones	<ol style="list-style-type: none"> 1. Se abre una ventana con un formulario vacío donde aparecen los campos de la tabla «Tipo Contrato» de la figura B.5, necesarios para crear un tipo de contrato. 2. Rellenar el formulario con los datos del tipo de contrato que se desea añadir. 3. Pulsar sobre el botón «Añadir».
Postcondición	El tipo de contrato queda añadido/modificado y el sistema lleva al usuario a la ventana de tipos de contrato donde se puede ver el listado de todos los tipos de contrato añadidos.
Excepciones	Se dejan campos vacíos o se introducen datos con un formato incorrecto. Otra forma de finalizar el caso de uso es la modificación. En este caso, se pulsa sobre el botón «Modificar» de un tipo de contrato de la tabla y se accede al mismo formulario, pero con los campos rellenos. Finalmente se pulsa en el botón «Modificar» y el tipo de contrato queda modificado.
Importancia	Alta

Tabla B.16: CU-2.4.1 Añadir/Modificar tipos de contrato.

CU-2.5	Mantenimiento de plazas
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-07
Descripción	Un administrativo puede realizar el mantenimiento de plazas
Precondición	Realizar el CU-2
Acciones	<ol style="list-style-type: none"> 1. Se abre una ventana donde aparece una tabla con las plazas creadas desde donde se podrá realizar el mantenimiento.
Postcondición	Ninguna
Excepciones	Si se pulsa sobre el botón «Nuevo», se accede a la creación de una plaza (CU-2.5.1). Si se pulsa sobre el botón «Modificar» de una plaza de la lista, se accede a la ventana de modificación. Desde la ventana de modificación también se puede acceder al CU-2.6 de asignar una plaza a un docente. Por último, si se pulsa sobre el botón «Eliminar» de una plaza el sistema pregunta si está seguro, y al pulsar en «Sí», esta se elimina.
Importancia	Alta

Tabla B.17: CU-2.5 Mantenimiento de plazas.

CU-2.5.1	Añadir/Modificar plazas
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-07
Descripción	Un administrativo añade o modifica una plaza
Precondición	Realizar el CU-2.5 y tener algún área y tipo de contrato creados
Acciones	<ol style="list-style-type: none"> 1. Se abre una ventana con un formulario vacío donde aparecen los campos de la tabla «Plaza» de la figura B.5, necesarios para crear una plaza. También se debe indicar el tipo de contrato y se puede asignar la plaza a un docente (CU-2.6). 2. Rellenar el formulario con los datos de la plaza que se desea añadir. 3. Pulsar sobre el botón «Añadir».
Postcondición	La plaza queda añadida/modificada y el sistema lleva al usuario a la ventana de plazas donde se puede ver el listado de todas las plazas añadidas.
Excepciones	Se dejan campos vacíos o se introducen datos con un formato incorrecto. Otra forma de finalizar el caso de uso es la modificación. En este caso, se pulsa sobre el botón «Modificar» de una plaza de la tabla y se accede al mismo formulario, pero con los campos rellenos. Finalmente se pulsa en el botón «Modificar» y la plaza queda modificada.
Importancia	Alta

Tabla B.18: CU-2.5.1 Añadir/Modificar plazas.

CU-2.6	Asignar plaza a docente
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-07, RF-12
Descripción	Un administrativo puede asignar una plaza a un docente. Este caso de uso es una extensión del CU-2.5.1, ya que la asignación de la plaza a un docente se realiza en la creación o modificación de la misma
Precondición	Realizar el CU-2.5 y tener creado un docente
Acciones	<ol style="list-style-type: none"> 1. Si la plaza que se desea asignar ya está creada, pulsar en el botón «Modificar» de la fila de la tabla que corresponde a la plaza. 2. Se abre una ventana con un formulario que tendrá los campos de la tabla «Plaza» de la figura B.5 rellenos. En el campo llamado «Docente», se podrá seleccionar el docente al que se quiere asignar la plaza desde un seleccionable con búsqueda.
Postcondición	La plaza queda asignada al docente seleccionado y el sistema lleva al usuario a la vista del mantenimiento de plazas.
Excepciones	La asignación se puede realizar a la hora de crear una plaza. Para ello, se debe pulsar en el botón «Nuevo» desde la ventana de mantenimiento de plazas y se abrirá una ventana con el mismo formulario, pero vacío. Después de rellenar los datos, se pulsa en el botón «Añadir» y la plaza queda creada y asignada. Si el docente al que se desea vincular la plaza todavía no está creado, este se puede crear pulsando sobre el botón «Nuevo docente» que se encuentra dentro del formulario. Pulsar el botón hará que se abra una ventana flotante que cubre el CU-2.3.1 para añadir un nuevo docente.
Importancia	Alta

Tabla B.19: CU-2.6 Asignar plaza a docente.

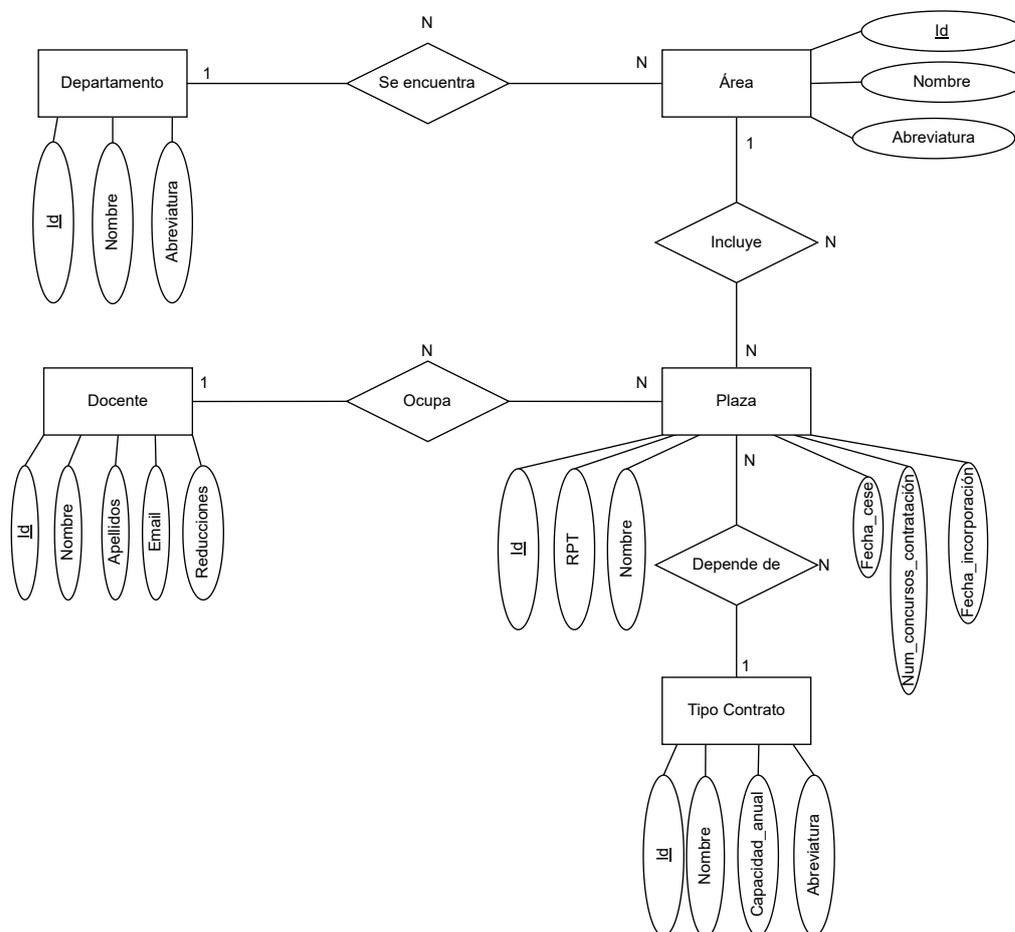


Figura B.5: Vista diagrama entidad relación para el CU-2

3. Asignación docente

En este apartado se va a exponer el diagrama de casos de uso 3, el cual se encuentra definido en la ilustración B.6 y en las tablas B.20, B.21, B.22, B.23, B.24, B.25, B.26 y B.27

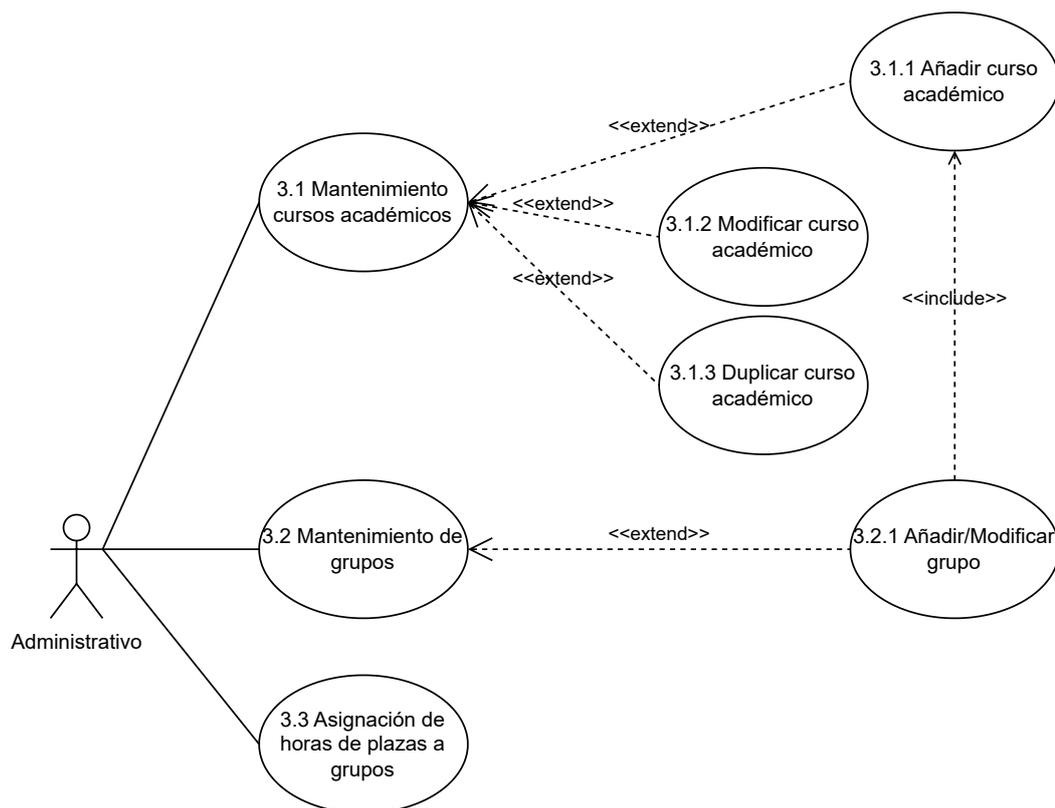


Figura B.6: Diagrama de caso de uso - 3. Asignación docente

CU-3	Asignación docente
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-06, RF-07, RF-08, RF-09, RF-10
Descripción	Un administrativo puede realizar las labores de mantenimiento de cursos y grupos, además de las asignaciones de horas de plazas a grupos
Precondición	Tener iniciada sesión con una cuenta con permisos administrativos
Acciones	<ol style="list-style-type: none"> 1. Seleccionar la opción del menú «Grupos» para realizar el mantenimiento de los mismos. 2. Se abre la ventana de mantenimiento de cursos.
Postcondición	El sistema lleva al usuario a la ventana de la opción pulsada.
Excepciones	Otras posibilidades del caso de uso son el mantenimiento de cursos académicos, que se puede realizar seleccionando la opción del menú «Cursos» (CU-3.1) y la asignación de horas de una plaza a un grupo, que se puede realizar seleccionando la opción «Horas» (CU-3.3).
Importancia	Alta

Tabla B.20: CU-3 Asignación docente.

CU-3.1	Mantenimiento cursos académicos
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-06
Descripción	Un administrativo puede realizar el mantenimiento de los cursos académicos
Precondición	Realizar el CU-3
Acciones	<ol style="list-style-type: none"> 1. Seleccionar la opción «Cursos» del menú principal de la web. 2. Se abre una ventana donde aparece una tabla con los cursos creados desde donde se podrá realizar el mantenimiento.
Postcondición	Ninguna
Excepciones	Si se pulsa sobre el botón «Nuevo», se accede a la creación de un curso (CU-3.1.1), si se pulsa sobre el botón «Modificar» de un curso de la lista, se accede a la ventana de modificación (CU-3.1.2), si se pulsa sobre el botón «Modificar Año» de un curso de la lista, se accede a la ventana de modificación del campo año de inicio, si se pulsa sobre el botón «Eliminar» de un curso el sistema pregunta si está seguro, y al pulsar en «Sí», este se elimina produciendo un borrado en cascada de sus relaciones con asignaturas y grupos. Por último, si se pulsa sobre «Duplicar», se duplica el curso junto a todas sus vinculaciones (CU-3.1.3)
Importancia	Alta

Tabla B.21: CU-3.1 Mantenimiento cursos académicos.

CU-3.1.1	Añadir curso académico
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-03, RF-06
Descripción	Un administrativo añade un curso académico
Precondición	Realizar el CU-3.1
Acciones	<ol style="list-style-type: none"> 1. Se abre una ventana con un formulario. El formulario contiene un campo para indicar el año de comienzo del curso. 2. Pulsar en el botón «Siguiente» 3. Se abre una ventana que contiene una tabla con las asignaturas existentes desde donde se podrán seleccionar las deseadas. También contiene tres bloques para las diferentes modalidades donde se podrá indicar el número de alumnos previstos, el número de grupos de teoría y el número de grupos de práctica que se deben crear para las asignaturas seleccionadas. 4. Pulsar en el botón «Añadir» 5. Aparece la misma pantalla con los campos limpios para poder añadir más asignaturas y grupos.
Postcondición	El curso se crea junto a las asignaturas y grupos elegidos.
Excepciones	Se dejan campos vacíos o se introducen datos con un formato incorrecto.
Importancia	Alta

Tabla B.22: CU-3.1.1 Añadir curso académico.

CU-3.1.2		Modificar curso académico
Versión		1.0
Autor		Ignacio Dávila García
Requisitos asociados	aso-	RF-03, RF-06
Descripción		Un administrativo modifica un curso académico
Precondición		Realizar el CU-3.1
Acciones		<ol style="list-style-type: none"> 1. Se abre una ventana con el listado de asignaturas del curso. Desde esa ventana se pueden añadir nuevas asignaturas al curso pulsando sobre el botón «Añadir asignaturas». 2. Al pulsar sobre «Añadir asignaturas» se abre una ventana flotante desde la que se pueden seleccionar las asignaturas e indicar el número de alumnos previstos y el número de grupos de teoría y práctica a crear para cada modalidad. 3. Pulsar sobre el botón «Añadir». 4. La ventana se cierra y se puede ver en la tabla como las nuevas asignaturas aparecen vinculadas al curso.
Postcondición		El curso se modifica y el sistema permanece en la ventana de modificación del curso desde donde se pueden ver los cambios realizados.
Excepciones		Si se pulsa sobre el botón «Quitar del curso» de una de las asignaturas, el sistema pregunta si está seguro de la eliminación, y al pulsar en la opción «Sí», la asignatura desaparece del listado y se produce un borrado en cascada de sus grupos. Si se pulsa sobre el botón «Editar grupos» de una de las asignaturas se abre una ventana desde donde realizar la modificación de los grupos de la asignatura para ese curso. (CU-3.2.1)
Importancia		Alta

Tabla B.23: CU-3.1.2 Modificar curso académico.

CU-3.1.3	Duplicar curso académico
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-06
Descripción	Un administrativo duplica un curso académico con toda la información asociada
Precondición	Realizar el CU-3.1
Acciones	<ol style="list-style-type: none"> 1. Se abre una ventana modal de confirmación donde se pregunta si está seguro de duplicar el curso académico. 2. Pulsar en la opción «Sí».
Postcondición	El curso y todas sus asociaciones se duplican y el sistema permanece en la ventana de mantenimiento de cursos desde donde se puede ver como el nuevo curso se ha creado. El nuevo curso se crea con un año más en el campo año de inicio que el último existente.
Excepciones	Ninguna
Importancia	Alta

Tabla B.24: CU-3.1.3 Duplicar curso académico.

CU-3.2		Mantenimiento de grupos	
Versión			1.0
Autor			Ignacio Dávila García
Requisitos asociados			RF-03
Descripción		Un administrativo puede realizar el mantenimiento de grupos	
Precondición			Realizar el CU-3
Acciones			<ol style="list-style-type: none"> 1. Se abre una ventana donde aparece un selector de curso, con el último curso seleccionado por defecto. También aparece una tabla con todas las asignaturas del curso seleccionado junto al número de grupos de cada tipo que tienen vinculados.
Postcondición			Ninguna
Excepciones			Si se pulsa sobre el botón «Gestionar grupos» de uno de los registros de la tabla se accede a una ventana con la información de la asignatura y sus grupos. Desde esta ventana se pueden crear, modificar o eliminar grupos para esa asignatura. (CU-3.2.1)
Importancia			Alta

Tabla B.25: CU-3.2 Mantenimiento de grupos.

CU-3.2.1	Añadir/Modificar grupo
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	RF-03, RF-13
Descripción	Un administrativo añade o modifica los grupos de una asignatura
Precondición	Realizar el CU-3.2
Acciones	<ol style="list-style-type: none"> 1. Pulsar sobre el botón «Gestionar grupos» de una de las asignaturas de la tabla. 2. Se abre una nueva ventana con la información de la asignatura y un listado con los grupos que tiene vinculados. 3. Pulsar sobre el botón «Añadir grupo». 4. Se abre una ventana flotante con un formulario que contiene los campos de la tabla «Grupo» de la figura B.7. 5. Completar los campos y pulsar en el botón «Añadir»
Postcondición	El grupo se crea y queda vinculado a la asignatura. El usuario termina el caso de uso en la ventana que contiene los grupos de la asignatura.
Excepciones	Se dejan campos vacíos o se introducen datos con un formato incorrecto. Otra forma de finalizar el caso de uso es la modificación o eliminación. En el primer caso, se pulsa sobre el botón «Modificar» de un grupo de la tabla y se accede al mismo formulario de creación, pero con los campos rellenos. Si se pulsa en el botón «Modificar» la información del grupo queda actualizada. Por último, para el caso de eliminar, se pulsa sobre el botón «Eliminar» de un grupo de la tabla, el sistema pregunta si está seguro, y al pulsar en «Sí», este se elimina.
Importancia	Alta

Tabla B.26: CU-3.2.1 Añadir/Modificar grupo.

CU-3.3	Asignación de horas de plazas a grupos
Versión	1.0
Autor	Ignacio Dávila García
Requisitos asociados	aso- RF-11
Descripción	Un administrativo puede asignar horas de una plaza a un grupo durante un curso académico
Precondición	Realizar el CU-3
Acciones	<ol style="list-style-type: none"> 1. Se abre una ventana donde aparece un selector de curso, con el último curso seleccionado por defecto. También aparece una tabla con todos los grupos del curso y las plazas/docentes que tienen vinculados. 2. Pulsar sobre el botón «Editar» de una de las filas de la tabla. 3. Se abre una nueva ventana con la información del grupo y un listado con las plazas vinculadas. 4. Para vincular una nueva plaza pulsar sobre el botón «Añadir plaza». 5. Se abre una ventana flotante donde se puede buscar por el nombre de la plaza o del docente que se quiere vincular. 6. Pulsar sobre la plaza/docente a vincular e indicar en el campo horas el número de horas asignadas al grupo. 7. Pulsar sobre el botón «Vincular».
Postcondición	La plaza seleccionada se vincula al grupo con el número de horas indicado. El sistema permanece en la ventana del grupo donde aparecen las plazas asignadas.
Excepciones	Otra forma de finalizar el caso de uso es desvincular una plaza de un grupo. Para ello se pulsa en el botón «Eliminar» del listado de plazas del grupo. Aparece una ventana modal que pregunta si está seguro de la eliminación y, al pulsar en «Sí», desaparece la vinculación de la plaza al grupo. Finalmente, las horas que tiene asignadas una plaza/docente se pueden modificar. Para ello, se cambia el número de horas de la plaza desde la tabla (columna «Horas anuales») y para guardar los cambios se pulsa en el botón «Modificar».
Importancia	Alta

Tabla B.27: CU-3.3 Asignación de horas de plazas a grupos.

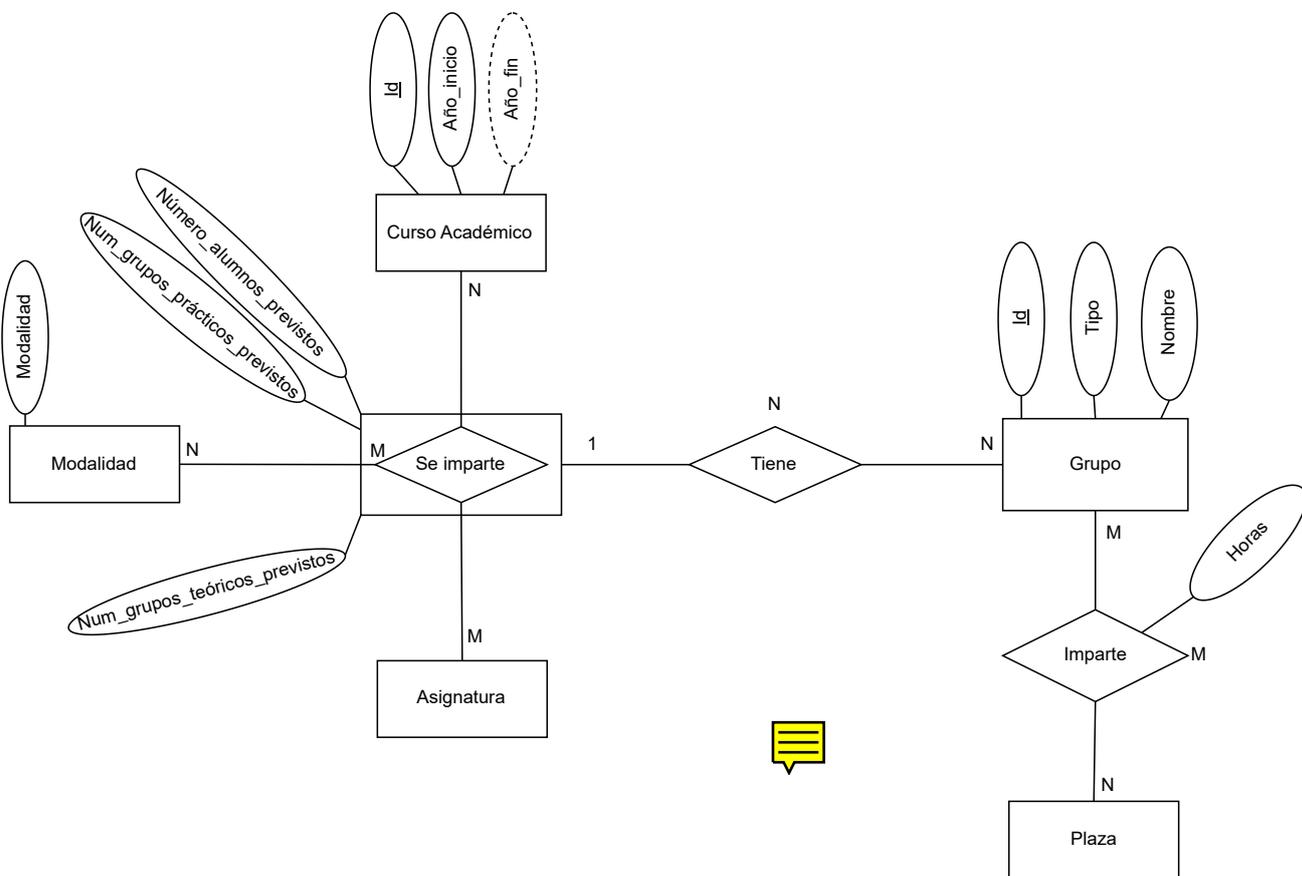


Figura B.7: Vista diagrama entidad relación para el CU-3

Prototipos de vistas/pantallas de los casos de uso

- CU-1. Mantenimiento académico.
 - CU-1.1 Mantenimiento de centros. Ver figura B.8.

Centros

Id	Nombre	Abreviatura	Email Administrativo	Botones acción	
1	Nombre 1	AA	Email 1	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
2	Nombre 2	BB	Email 2	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
3	Nombre 3	CC	Email 3	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
4	Nombre 4	DD	Email 4	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>

Figura B.8: Mantenimiento de centros

- CU-1.1.1 Añadir/Modificar centros. Ver figura B.9.

Añadir Centro

Centros > Añadir Centro

Nombre

Abreviatura

Email del Administrativo

Figura B.9: Añadir/Modificar centros

- **CU-1.2** Mantenimiento de titulaciones. Ver figura B.10.

Titulaciones

Nuevo

Id	Nombre	Abreviatura	URL	Centro	Botones acción
1	Nombre 1	AA	https://www.ubu.es/	Centro 1	Modificar Eliminar
2	Nombre 2	BB	https://www.ubu.es/	Centro 5	Modificar Eliminar
3	Nombre 3	CC	https://www.ubu.es/	Centro 1	Modificar Eliminar
4	Nombre 4	DD	https://www.ubu.es/	Centro 3	Modificar Eliminar

Figura B.10: Mantenimiento de titulaciones

- **CU-1.2.1** Añadir/Modificar titulaciones. Ver figura B.11.

Añadir Titulación

Titulaciones > Añadir Titulación

Nombre

Abreviatura

URL

Centro

Añadir

Figura B.11: Añadir/Modificar titulaciones

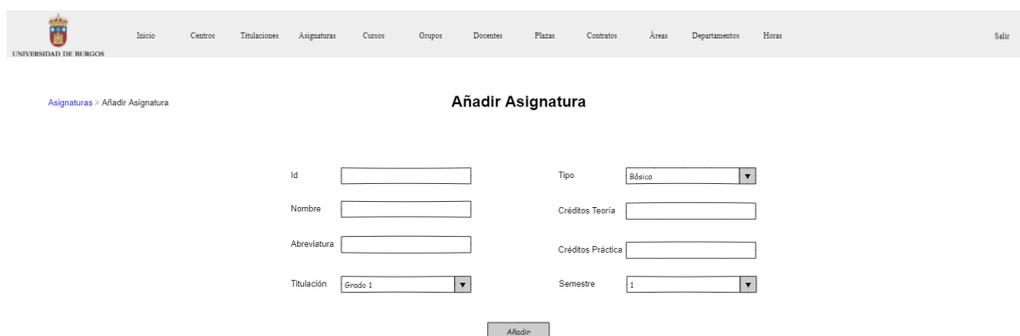
- **CU-1.3** Mantenimiento de asignaturas. Ver figura B.12.



UNIVERSIDAD DE BURGOS									
Inicio Cursos Titulaciones Asignaturas Cursos Grupos Docentes Plazas Contratos Áreas Departamentos Horas Salir									
Asignaturas									
Nuevo									
id	Nombre	Tipo	Titulación	Curso	Semestre	Créditos Teoría	Créditos Práctica	Abreviatura	Botones acción
1	Nombre 1	Obligatoria	Grado 1	3º	1	3	3	AA, A1	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
2	Nombre 2	Básica	Master 5	1º	2	3	3	BB	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
3	Nombre 3	Obligatoria	Grado 4	3º	1	3	3	CC	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
4	Nombre 4	Opcativa	Grado 3	4º	1/2	3	3	DD, D1	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>

Figura B.12: Mantenimiento de asignaturas

- **CU-1.3.1** Añadir/Modificar asignaturas. Ver figura B.13.



Asignaturas - Añadir Asignatura

Añadir Asignatura

id

Nombre

Abreviatura

Titulación

Tipo

Créditos Teoría

Créditos Práctica

Semestre

Figura B.13: Añadir/Modificar asignaturas

- **CU-2.** Mantenimiento de profesorado.
 - **CU-2.1** Mantenimiento de departamentos. Ver figura B.14.



Figura B.14: Mantenimiento de departamentos

- **CU-2.1.1** Añadir/Modificar departamentos. Ver figura B.15.



Figura B.15: Añadir/Modificar departamentos

- **CU-2.2** Mantenimiento de áreas. Ver figura B.16.

UNIVERSIDAD DE BURGOS

Inicio Centros Titulaciones Asignaturas Cursos Grupos Docentes Plazas Contratos **Áreas** Departamentos Horas Salir

Áreas

Nuevo

Id	Nombre	Abreviatura	Departamento	Botones acción	
1	Nombre 1	AA	Departamento	Modificar	Eliminar
2	Nombre 2	AA	Departamento	Modificar	Eliminar
3	Nombre 3	AA	Departamento	Modificar	Eliminar
4	Nombre 4	AA	Departamento	Modificar	Eliminar

Figura B.16: Mantenimiento de áreas

- **CU-2.2.1** Añadir/Modificar áreas. Ver figura B.17.

UNIVERSIDAD DE BURGOS

Inicio Centros Titulaciones Asignaturas Cursos Grupos Docentes Plazas Contratos **Áreas** Departamentos Horas Salir

Áreas > Nueva

Añadir Área

Nombre

Abreviatura

Departamento

Añadir

Figura B.17: Añadir/Modificar áreas

- **CU-2.3** Mantenimiento de docentes. Ver figura B.18.

UNIVERSIDAD DE BURGOS

Inicio Centros Titulaciones Asignaturas Cursos Grupos **Docentes** Plazas Contratos Áreas Departamentos Horas Salir

Docentes

Nuevo

Id	Nombre	Apellidos	Email	Reducciones	Botones acción	
1	Nombre 1	Apellidos 1	Email 1	10	Modificar	Eliminar
2	Nombre 2	Apellidos 2	Email 2	2	Modificar	Eliminar
3	Nombre 3	Apellidos 3	Email 3	60	Modificar	Eliminar
4	Nombre 4	Apellidos 4	Email 4	53	Modificar	Eliminar

Figura B.18: Mantenimiento de docentes

- **CU-2.3.1** Añadir/Modificar docentes. Ver figura B.19.

UNIVERSIDAD DE BURGOS

Inicio Centros Titulaciones Asignaturas Cursos Grupos Docentes Plazas Contratos Áreas Departamentos Horas Salir

Docentes > Añadir Docente

Añadir Docente

Nombre Email

Apellidos Reducciones

Figura B.19: Añadir/Modificar docentes

- **CU-2.4** Mantenimiento de tipos de contrato. Ver figura B.20.

UNIVERSIDAD DE BURGOS

Inicio Centros Titulaciones Asignaturas Cursos Grupos Docentes Plazas **Contratos** Áreas Departamentos Horas Salir

Tipos de contrato

Id	Nombre	Abreviatura	Capacidad anual	Botones acción
1	Nombre 1	AA	500	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
2	Nombre 2	AA	1000	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
3	Nombre 3	AA	300	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
4	Nombre 4	AA	800	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>

Figura B.20: Mantenimiento de tipos de contrato

- **CU-2.4.1** Añadir/Modificar tipos de contrato. Ver figura B.21.

UNIVERSIDAD DE BURGOS

Inicio Centros Titulaciones Asignaturas Cursos Grupos Docentes Plazas Contratos Áreas Departamentos Horas Salir

Contratos > Nuevo

Añadir tipo contrato

Nombre

Abreviatura

Capacidad anual

Figura B.21: Añadir/Modificar tipos de contrato

- **CU-2.5** Mantenimiento de plazas. Ver figura B.22.

UNIVERSIDAD DE BURGOS

Inicio Centros Titulaciones Asignaturas Cursos Grupos Docentes **Plazas** Contratos Áreas Departamentos Horas Salir

Plazas

Id	Nombre	RPT	Fecha incorporación	Fecha cese	Nº concursos contratación	Docente	Tipo contrato	Botones acción
1	Nombre 1	RPT	01/01/2023	01/01/2024	3	Docente 1	Tipo 1	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
2	Nombre 2	RPT	01/01/2023	01/01/2024	2	Ninguno	Tipo 1	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
3	Nombre 3	RPT	01/01/2023	01/01/2024	2	Docente 15	Tipo 3	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
4	Nombre 4	RPT	01/01/2023	01/01/2024	1	Docente 2	Tipo 2	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>

Figura B.22: Mantenimiento de plazas

- **CU-2.5.1** Añadir/Modificar plazas. Ver figura B.23.

The screenshot shows the 'Añadir Plaza' form in the University of Bergamos system. The form is titled 'Añadir Plaza' and is located under the 'Plazas > Nueva' menu. The form contains the following fields:

- Nombre: Text input field.
- RTP: Text input field.
- Fecha incorporación: Text input field.
- Fecha cese: Text input field.
- Nº concursos contratación: Text input field.
- Docente: Dropdown menu with 'ninguno' selected and a 'Nuevo docente' button.
- Tipo contrato: Dropdown menu.

There is an 'Añadir' button at the bottom of the form.

Figura B.23: Añadir/Modificar plazas

- **CU-2.6** Asignar plaza a docente. Ver figura B.24.

The screenshot shows the 'Añadir Plaza' form in the University of Bergamos system, which is identical to the one in Figure B.23. The form is titled 'Añadir Plaza' and is located under the 'Plazas > Nueva' menu. The form contains the following fields:

- Nombre: Text input field.
- RTP: Text input field.
- Fecha incorporación: Text input field.
- Fecha cese: Text input field.
- Nº concursos contratación: Text input field.
- Docente: Dropdown menu with 'ninguno' selected and a 'Nuevo docente' button.
- Tipo contrato: Dropdown menu.

There is an 'Añadir' button at the bottom of the form.

Figura B.24: Asignar plaza a docente

- **CU-3.** Asignación docente.
 - **CU-3.1** Mantenimiento cursos académicos. Ver figuras B.25 y B.26.

Cursos

Nuevo

Id	Año inicio	Año fin	Botones acción			
<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	Modificar Año	Modificar	Duplicar	Eliminar
1	2019	2020	Modificar Año	Modificar	Duplicar	Eliminar
2	2020	2021	Modificar Año	Modificar	Duplicar	Eliminar
3	2021	2022	Modificar Año	Modificar	Duplicar	Eliminar
4	2022	2023	Modificar Año	Modificar	Duplicar	Eliminar

Figura B.25: Mantenimiento cursos académicos

Curso 2022/2023

Año inicio

Figura B.26: Modificar año curso académico

- **CU-3.1.1** Añadir curso académico. Ver figuras B.27 y B.28.

UNIVERSIDAD DE BURGOS

Inicio Centros Titulaciones Asignaturas Cursos Grupos Docentes Plazas Contratos Áreas Departamentos Horas Salir

Cursos > Nuevo

Añadir Curso

Año inicio

Siguiente

Figura B.27: Añadir curso académico

UNIVERSIDAD DE BURGOS

Inicio Centros Titulaciones Asignaturas Cursos Grupos Docentes Plazas Contratos Áreas Departamentos Horas Salir

Cursos > Nuevo

Añadir Curso

Listado Asignaturas								
Sele	Id	Asignatura	Abreviatura	Tipo	Créditos teoría	Créditos práctica	Titulación	Centro
<input type="checkbox"/>		<input type="text"/>						
<input checked="" type="checkbox"/>	1	Nombre 1	AA	Tipo 1	3	3	Grado 1	Centro 1
<input type="checkbox"/>	2	Nombre 2	BB	Tipo 1	3	3	Grado 5	Centro 1
<input type="checkbox"/>	3	Nombre 3	CC	Tipo 2	3	3	Master 1	Centro 3
<input checked="" type="checkbox"/>	4	Nombre 4	DD	Tipo 5	3	3	Master 6	Centro 2

Grupos

Presencial

Nº Alumnos previstos

Número previsto de grupos de teoría:

Número previsto de grupos de práctica:

Online

Nº Alumnos previstos

Número previsto de grupos de teoría:

Número previsto de grupos de práctica:

Inglés

Nº Alumnos previstos

Número previsto de grupos de teoría:

Número previsto de grupos de práctica:

Añadir

Figura B.28: Añadir curso académico 2

- CU-3.1.2 Modificar curso académico. Ver figuras B.29 y B.30.



Modificar Curso

Listado Asignaturas

Id	Asignatura	Abreviatura	Tipo	Créditos teoría	Créditos práctica	Titulación	Centro	
<input type="checkbox"/>	<input type="text"/>							
1	Nombre 1	AA	Tipo 1	3	3	Grado 1	Centro 1	<input type="button" value="Editar grupos"/> <input type="button" value="Quitar del curso"/>
2	Nombre 2	BB	Tipo 1	3	3	Grado 5	Centro 1	<input type="button" value="Editar grupos"/> <input type="button" value="Quitar del curso"/>
3	Nombre 3	CC	Tipo 2	3	3	Master 1	Centro 3	<input type="button" value="Editar grupos"/> <input type="button" value="Quitar del curso"/>
4	Nombre 4	DD	Tipo 5	3	3	Master 6	Centro 2	<input type="button" value="Editar grupos"/> <input type="button" value="Quitar del curso"/>

Figura B.29: Modificar curso académico

Listado Asignaturas								
Selel	Id	Asignatura	Abreviatura	Tipo	Créditos teoría	Créditos práctica	Titulación	Centro
<input type="checkbox"/>	<input type="text"/>							
<input checked="" type="checkbox"/>	1	Nombre 1	AA	Tipo 1	3	3	Grado 1	Centro 1
<input type="checkbox"/>	2	Nombre 2	BB	Tipo 1	3	3	Grado 5	Centro 1
<input type="checkbox"/>	3	Nombre 3	CC	Tipo 2	3	3	Master 1	Centro 3
<input checked="" type="checkbox"/>	4	Nombre 4	DD	Tipo 5	3	3	Master 6	Centro 2

Grupos

Presencial

Nº Alumnos previstos:

Número previsto de grupos de teoría:

Número previsto de grupos de práctica:

Online

Nº Alumnos previstos:

Número previsto de grupos de teoría:

Número previsto de grupos de práctica:

Inglés

Nº Alumnos previstos:

Número previsto de grupos de teoría:

Número previsto de grupos de práctica:

Figura B.30: Añadir asignaturas al curso

- **CU-3.2** Mantenimiento de grupos. Ver figura B.31.

Grupos

Curso 2022/2023

Nuevo

Centro	Titulación	Asignatura	Semestre	Nº grupos teoría	Nº grupos práctica	Nº grupos inglés teoría	Nº grupos inglés práctica	Nº grupos online teoría	Nº grupos online práctica	Acciones
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>							
Centro	Titulación	Asignatura	Semestre	1	1	1	1	1	1	Gestionar grupos
Centro	Titulación	Asignatura	Semestre	1	1	1	1	1	1	Gestionar grupos
Centro	Titulación	Asignatura	Semestre	1	1	1	1	1	1	Gestionar grupos
Centro	Titulación	Asignatura	Semestre	1	1	1	1	1	1	Gestionar grupos
Centro	Titulación	Asignatura	Semestre	1	1	1	1	1	1	Gestionar grupos
Centro	Titulación	Asignatura	Semestre	1	1	1	1	1	1	Gestionar grupos
Centro	Titulación	Asignatura	Semestre	1	1	1	1	1	1	Gestionar grupos
Centro	Titulación	Asignatura	Semestre	1	1	1	1	1	1	Gestionar grupos
Centro	Titulación	Asignatura	Semestre	1	1	1	1	1	1	Gestionar grupos

Figura B.31: Mantenimiento de grupos

- **CU-3.2.1** Añadir/Modificar grupo. Ver figuras B.32 y B.33.

Curso 2022/2023

Nombre centro Asignatura Nombre titulación

Grupos:

Id	Tipo	Nombre		
Id	Teoría	Nombre	Modificar	Eliminar
Id	Teoría	Nombre	Modificar	Eliminar
Id	Práctica	Nombre	Modificar	Eliminar

Añadir grupo

Figura B.32: Añadir/Modificar grupo

Añadir Grupo

Tipo

Nombre

Figura B.33: Añadir/Modificar grupo 2

- **CU-3.3** Asignación de horas de plazas a grupos. Ver figuras [B.34](#), [B.35](#) y [B.36](#).

UNIVERSIDAD DE BURGOS

Inicio Centros Titulaciones Asignaturas Cursos Grupos Docentes Plazas Contratos Áreas Departamentos **Horas** Salir

Curso

Listado de grupos

Centro	Titulación	Asignatura	Semestre	Id Grupo	Plaza/Docente	Horas	Horas totales docente	Plaza/Docente	Horas	Horas totales docente	Acciones
<input type="text"/>											
Centro	Titulación	Asignatura	Semestre	Id Grupo	Plaza/Docente	Horas	Horas totales docente	Plaza/Docente	Horas	Horas totales docente	<input type="button" value="Editar"/>
Centro	Titulación	Asignatura	Semestre	Id Grupo	Plaza/Docente	Horas	Horas totales docente	Plaza/Docente	Horas	Horas totales docente	<input type="button" value="Editar"/>
Centro	Titulación	Asignatura	Semestre	Id Grupo	Plaza/Docente	Horas	Horas totales docente	Plaza/Docente	Horas	Horas totales docente	<input type="button" value="Editar"/>
Centro	Titulación	Asignatura	Semestre	Id Grupo	Plaza/Docente	Horas	Horas totales docente	Plaza/Docente	Horas	Horas totales docente	<input type="button" value="Editar"/>
Centro	Titulación	Asignatura	Semestre	Id Grupo	Plaza/Docente	Horas	Horas totales docente	Plaza/Docente	Horas	Horas totales docente	<input type="button" value="Editar"/>
Centro	Titulación	Asignatura	Semestre	Id Grupo	Plaza/Docente	Horas	Horas totales docente	Plaza/Docente	Horas	Horas totales docente	<input type="button" value="Editar"/>
Centro	Titulación	Asignatura	Semestre	Id Grupo	Plaza/Docente	Horas	Horas totales docente	Plaza/Docente	Horas	Horas totales docente	<input type="button" value="Editar"/>
Centro	Titulación	Asignatura	Semestre	Id Grupo	Plaza/Docente	Horas	Horas totales docente	Plaza/Docente	Horas	Horas totales docente	<input type="button" value="Editar"/>

Figura B.34: Asignación de horas de plazas a grupos

Apéndice C

Especificación de diseño

C.1. Introducción

Una vez realizado el estudio y especificación de los requisitos de la aplicación web, se debe realizar el diseño de la misma. En este anexo se pretende aportar información sobre el diseño de los datos que utiliza la aplicación junto al diseño procedimental y arquitectónico del proyecto.

C.2. Diseño de datos

Gracias a la especificación de requisitos y casos de uso, se puede obtener una visión global de la aplicación que permite deducir las entidades, acompañadas de sus datos, necesarias para poder cumplir con lo requerido.

En primer lugar, podemos obtener la visión global de las entidades relacionadas mediante el diagrama general de Entidad-Relación de la figura C.1.

El diagrama obtenido tiene un gran tamaño y, para mejorar la visualización y comprensión del mismo, se ha decidido dividir en vistas donde se incluyan los datos de cada una de las entidades.

La primera vista hace referencia al apartado de mantenimiento académico y se puede ver en la figura B.3, la segunda al mantenimiento de profesorado B.5 y la última a la asignación docente B.7.

Del diagrama entidad-relación se puede obtener el diagrama relacional de la figura C.2 en el que se pueden ver las tablas que contendrá la base de datos de la aplicación web.

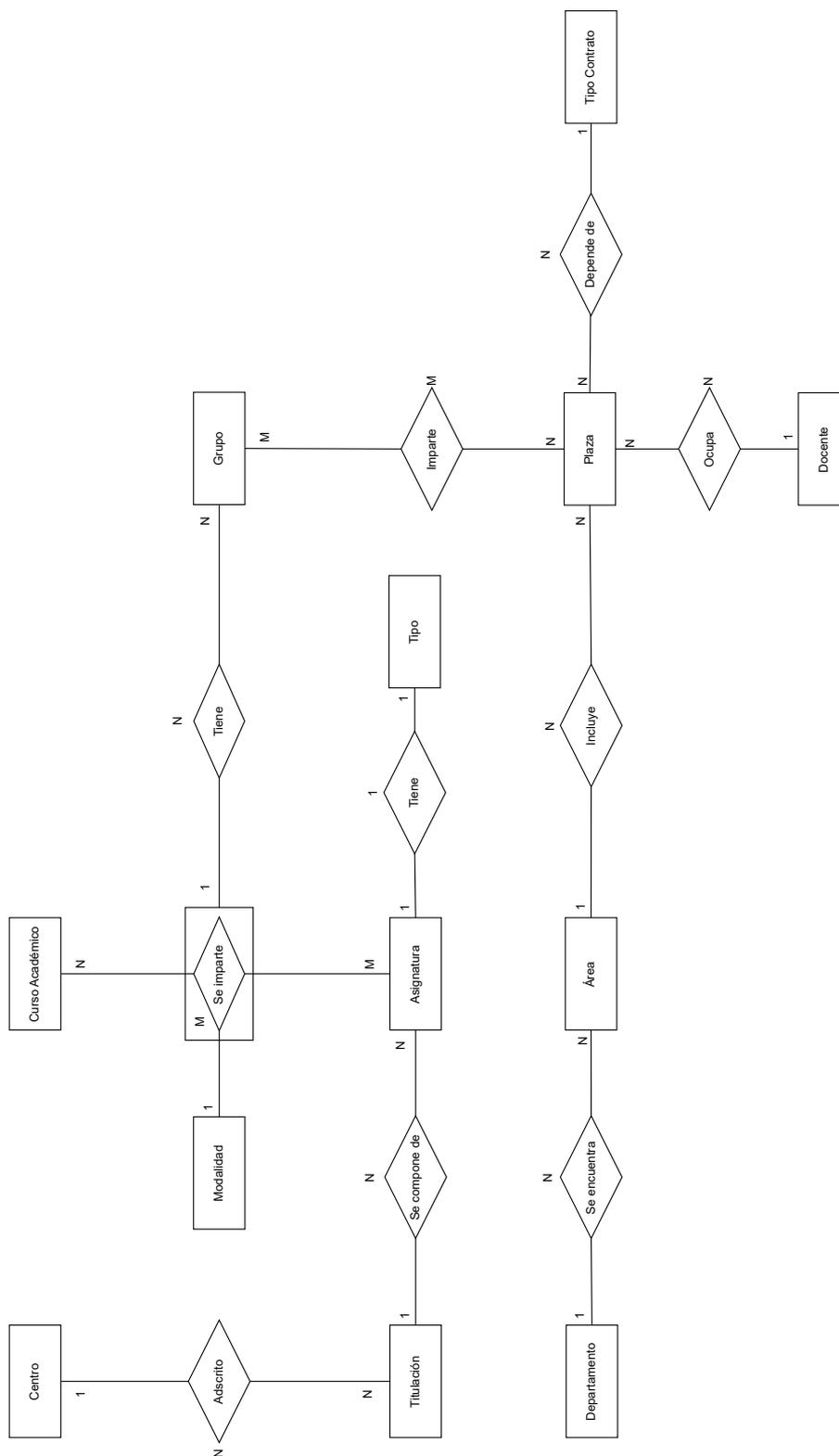


Figura C.1: Diagrama general entidad-relación

Campo	Tipo	Descripción
id	Entero <u>PK</u>	Identificador. Clave primaria creada de forma autoincremental.
codigo	Entero	Código interno del centro de la universidad. Sólo tiene carácter informativo.
abreviatura	Texto	Abreviatura del centro.
email	Texto	Email administrativo del centro.

Tabla C.1: Diccionario de datos. Tabla Centro

En este diagrama se pueden ver las tablas de la base de datos juntos los distintos campos que tendrá cada una. Como se puede ver en la figura C.2, las tablas centro, titulación y asignatura tienen un campo llamado código. Este código campo podría haber sido utilizado como clave primaria, pero al ser un campo que introduce el usuario, se decidió mantener una clave primaria auto-incremental con la que se hacen las relaciones de las tablas, y además, añadir ese campo para que se puedan hacer búsquedas o filtrar por el mismo sin que su uso pueda afectar a la consistencia del sistema.

Otro aspecto relevante es que en las relaciones que dan lugar a la creación de una tabla intermedia se siguió el mismo patrón que antes. Aunque la teoría diga que las claves de las tablas que se relacionan pasan a ser claves primarias de la nueva tabla que se genera, se decidió tener una única clave primaria auto-incremental y tener las claves de las tablas relacionadas como claves foráneas. De esta manera, las búsquedas en la base de datos estarán más optimizadas al buscar como clave primaria un único campo y no la composición de varios. Además, se evita cualquier tipo de error de clave primaria al ser la propia base de datos la que asigna esta y no el código creado.

Diccionario de datos

A continuación se va a especificar, para cada tabla de la base de datos, el diccionario de datos correspondiente. De esta forma se pretende facilitar la comprensión de los datos almacenados en la base de datos.

La tabla `centro` almacena los diferentes centros de la universidad. Se puede ver su información en la tabla C.1.

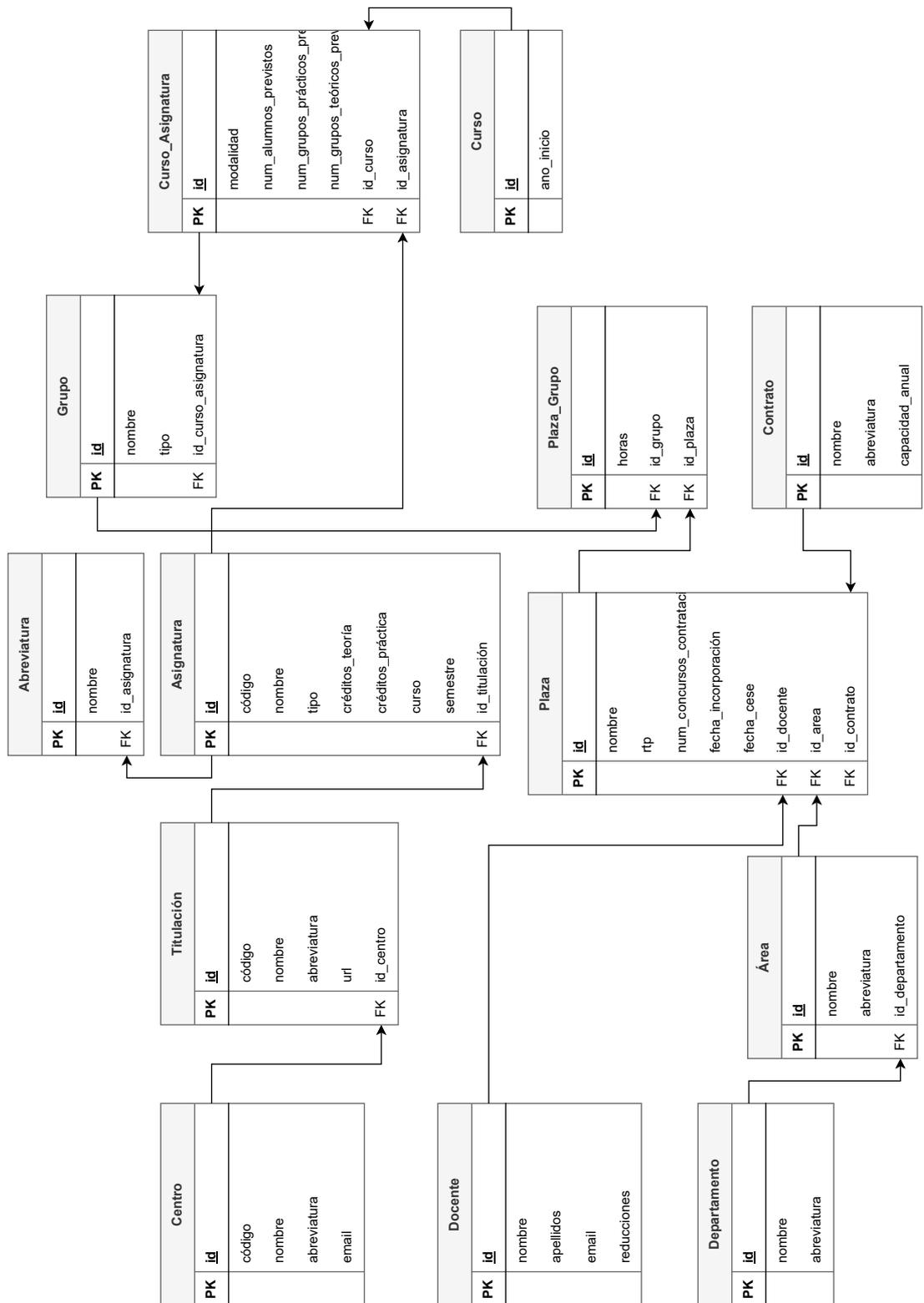


Figura C.2: Diagrama relacional

Campo	Tipo	Descripción
<u>id</u>	Entero <u>PK</u>	Identificador de la titulación. Clave primaria creada de forma autoincremental.
codigo	Entero	Código interno de la titulación.
nombre	Texto	Nombre de la titulación.
abreviatura	Texto	Abreviatura de la titulación.
url	Texto	Dirección web de la titulación.
id_centro	Entero FK(Centro)	Identificador del centro asociado a la titulación.

Tabla C.2: Diccionario de datos. Tabla Titulación

La tabla `titulación` almacena las titulaciones que son creadas y que pertenecen a un centro. Se puede ver su información en la tabla [C.2](#).

La tabla `asignatura` almacena información sobre las asignaturas, cada asignatura pertenece a una titulación. Se puede ver su información en la tabla [C.3](#).

La tabla `abreviatura` almacena las abreviaturas de las asignaturas. De esta forma se permite que una asignatura pueda tener varias abreviaturas. Se puede ver su información en la tabla [C.4](#).

La tabla `curso` almacena información sobre los cursos. Se puede ver su información en la tabla [C.5](#).

La tabla `curso_asignatura` almacena la relación entre cursos y asignaturas. Se puede ver su información en la tabla [C.6](#).

La tabla `grupo` almacena información sobre los grupos de asignaturas. Se puede ver su información en la tabla [C.7](#).

La tabla `docente` almacena información sobre los docentes. Se puede ver su información en la tabla [C.8](#).

La tabla `departamento` almacena información sobre los departamentos de la universidad. Se puede ver su información en la tabla [C.9](#).

La tabla `área` almacena información sobre las áreas de un departamento. Se puede ver su información en la tabla [C.10](#).

Campo	Tipo	Descripción
id	Entero <u>PK</u>	Identificador de la asignatura. Clave primaria creada de forma autoincremental.
codigo	Entero	Código interno de la asignatura.
nombre	Texto	Nombre de la asignatura.
tipo	Texto	Tipo de la asignatura.
creditos_teoría	Entero	Créditos de teoría de la asignatura.
creditos_práctica	Entero	Créditos de práctica de la asignatura.
curso	Texto	Curso al que pertenece la asignatura.
semestre	Texto	Semestre en el que se imparte la asignatura.
id_titulación	Entero FK(Titulación)	Identificador de la titulación a la que pertenece la asignatura.

Tabla C.3: Diccionario de datos. Tabla Asignatura

Campo	Tipo	Descripción
id	Entero <u>PK</u>	Identificador de la abreviatura. Clave primaria creada de forma autoincremental.
abreviatura	Texto	Abreviatura de la asignatura.
id_asignatura	Entero FK(Asignatura)	Identificador de la asignatura asociada a la abreviatura.

Tabla C.4: Diccionario de datos. Tabla Abreviatura

Campo	Tipo	Descripción
id	Entero <u>PK</u>	Identificador del curso. Clave primaria creada de forma autoincremental.
ano_inicio	Texto	Año de inicio del curso.

Tabla C.5: Diccionario de datos. Tabla Curso

Campo	Tipo	Descripción
<u>id</u>	Entero <u>PK</u>	Identificador de la relación curso-asignatura. Clave primaria creada de forma autoincremental.
id_asignatura	Entero FK(Asignatura)	Identificador de la asignatura relacionada.
id_curso	Entero FK(Curso)	Identificador del curso relacionado.
modalidad	Texto	Modalidad de la relación curso-asignatura.
num_alumnos_previstos	Entero	Número de alumnos previstos para la relación curso-asignatura.
num_grupos_teoricos_previstos	Entero	Número de grupos teóricos previstos para la relación curso-asignatura.
num_grupos_practicos_previstos	Entero	Número de grupos prácticos previstos para la relación curso-asignatura.

Tabla C.6: Diccionario de datos. Tabla Curso-Asignatura

Campo	Tipo	Descripción
<u>id</u>	Entero <u>PK</u>	Identificador del grupo. Clave primaria creada de forma autoincremental.
nombre	Texto	Nombre del grupo.
tipo	Texto	Tipo de grupo.
id_curso_asignatura	Entero FK(Curso-Asignatura)	Identificador de la relación curso-asignatura asociada al grupo.

Tabla C.7: Diccionario de datos. Tabla Grupo

Campo	Tipo	Descripción
<u>id</u>	Entero <u>PK</u>	Identificador del docente. Clave primaria generada de forma autoincremental.
nombre	Texto	Nombre del docente.
apellidos	Texto	Apellidos del docente.
email	Texto	Email del docente
reducciones	Entero	Número de horas de reducciones del docente.

Tabla C.8: Diccionario de datos. Tabla Docente

Campo	Tipo	Descripción
<u>id</u>	Entero <u>PK</u>	Identificador del departamento. Clave primaria generada de forma autoincremental.
nombre	Texto	Nombre del departamento.
abreviatura	Texto	Abreviatura del departamento.

Tabla C.9: Diccionario de datos. Tabla Departamento

Campo	Tipo	Descripción
<u>id</u>	Entero <u>PK</u>	Identificador del área. Clave primaria creada de forma autoincremental.
nombre	Texto	Nombre del área.
abreviatura	Texto	Abreviatura del área.
id_departamento	Entero FK(Departamento)	Identificador del departamento al que pertenece el área.

Tabla C.10: Diccionario de datos. Tabla Área

Campo	Tipo	Descripción
<u>id</u>	Entero <u>PK</u>	Identificador del tipo de contrato. Clave primaria generada de forma autoincremental.
nombre	Texto	Nombre del tipo de contrato.
abreviatura	Texto	Abreviatura del tipo de contrato.
capacidad_anual	Entero	Capacidad anual en horas del tipo de contrato.

Tabla C.11: Diccionario de datos. Tabla Tipo Contrato

La tabla `contrato` almacena información sobre los tipos de contratos. Se puede ver su información en la tabla [C.11](#).

La tabla `plaza` almacena información sobre las plazas de contratación. Se puede ver su información en la tabla [C.12](#).

La tabla `plaza-grupo` almacena información sobre la asignación de horas de plazas a grupos. Se puede ver su información en la tabla [C.13](#).

Existen otras dos tablas llamadas `alemic_version` y `sessions` que son creadas por algunas de las bibliotecas que se utilizan en la aplicación. La primera guarda un registro de las migraciones de la base de datos y la segunda almacena la sesión de los usuarios.

Campo	Tipo	Descripción
<u>id</u>	Entero <u>PK</u>	Identificador de la plaza. Clave primaria generada de forma autoincremental
nombre	Texto	Nombre de la plaza
rpt	Texto	RPT de la plaza
num_concursos_contratacion	Entero	Número de concursos de contratación de la plaza
fecha_incorporacion	Fecha	Fecha de incorporación a la plaza.
fecha_cese	Fecha	Fecha de cese de la plaza.
id_docente	Entero	Identificador del docente asignado a la plaza.
id_area	Entero	Identificador del área asociada a la plaza.
id_contrato	Entero	Identificador del tipo de contrato asociado a la plaza.

Tabla C.12: Diccionario de datos. Tabla Plaza

Campo	Tipo	Descripción
<u>id</u>	Entero <u>PK</u>	Identificador de la asignación. Clave primaria generada de forma autoincremental.
horas	Entero	Horas de la plaza asignadas al grupo.
id_grupo	Entero	Identificador del grupo.
id_plaza	Entero	Identificador de la plaza.

Tabla C.13: Diccionario de datos. Tabla Plaza-Grupo

C.3. Diseño procedimental

El diseño procedimental es una fase fundamental en el desarrollo de proyectos informáticos, donde se establecen los pasos necesarios para realizar una actividad dentro de la aplicación.

■ Creación de centro

El diagrama C.3 muestra los pasos necesarios para crear un nuevo centro en la aplicación web.

Se parte de que el usuario está en la ventana de centros. Desde ahí el usuario debe pulsar en «Nuevo», lo que provocará la redirección a la ventana que contiene el formulario de creación. El usuario debe rellenar el formulario y pulsar sobre «Añadir».

Al hacer esto los datos se mandan al servidor. Si todo está correcto el nuevo centro se almacena en la base de datos, mientras que si hay algún error en los datos introducidos, se recarga la página del formulario indicando los errores de cada campo.

El método `save` se encarga de comprobar si el objeto centro tiene identificador, ya que si no tiene, se trata de un objeto no almacenado en la base de datos y debe llamar al método `add` para guardarlo, mientras que si ya tiene `id`, quiere decir que se trata de una modificación y tan sólo debe hacer el `commit` para guardar los datos modificados.

Una vez se ha guardado el nuevo centro, la aplicación redirecciona al usuario a la página de centros.

El diagrama C.3 se va a utilizar como ejemplo de creación para las tablas de «Titulación», «Asignatura», «Docente», «Plaza», «Contrato», «Área» y «Departamento». Esto es debido a que la creación en estas tablas es igual y sólo cambia el nombre de los modelos, controladores y rutas, pero el funcionamiento es el mismo.

■ Modificación de centro

En el diagrama C.4 se muestra el proceso seguido para modificar un centro. Se parte de la base de que el centro ya se encuentra creado. Al igual que en el caso anterior, este diagrama sirve de ejemplo para la modificación de «Titulación», «Asignatura», «Docente», «Plaza», «Contrato», «Área» y «Departamento».

Suponiendo que el usuario se encuentra en la ventana de centros y que ahí por lo menos un centro creado, pulsa en el icono de lápiz de uno de los centros de la lista.

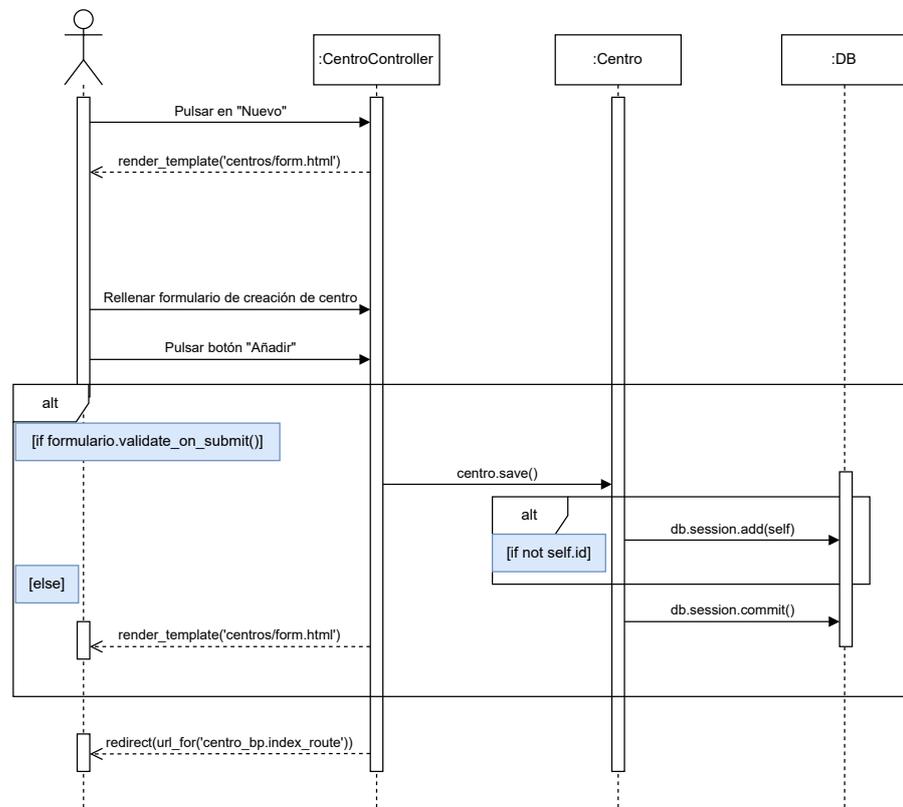


Figura C.3: Diagrama de secuencia referente a la creación de centros.

Al realizar esta acción se abre un formulario con los datos del centro seleccionado. Desde aquí el usuario puede cambiar los campos que quiera. Una vez realizados los cambios debe pulsar sobre «Modificar». Una vez modificados los campos y pulsada la opción indicada anteriormente, el servidor recogerá los datos. Si todo está bien, las modificaciones se almacenarán en la base de datos y el usuario será redirigido a la ventana principal de los centros. En caso contrario, se recargará la página del formulario indicando los campos que tienen errores y qué errores son.

■ Eliminación de centro

El diagrama C.5 muestra la secuencia de pasos que se siguen para eliminar un centro de la aplicación.

Partiendo de la base de que existe al menos un centro creado y que el usuario se encuentra en la ventana de centros, el usuario pulsará

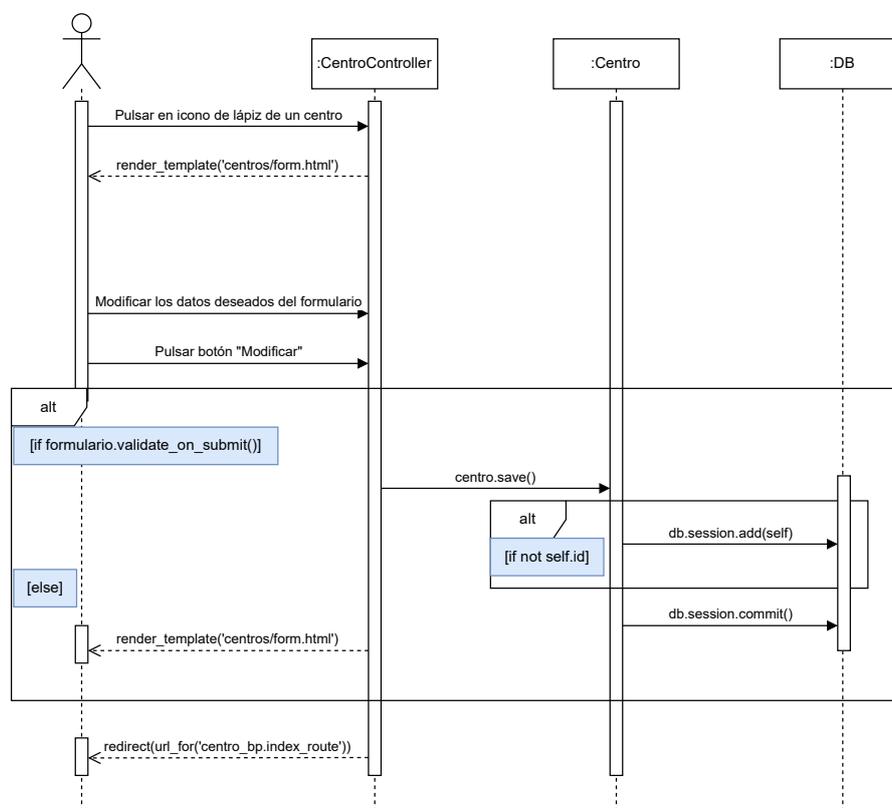


Figura C.4: Diagrama de secuencia referente a la modificación de centros.

sobre el icono de una papelerita de un centro de la tabla. Al realizar esta acción aparece una ventana de confirmación para afirmar que se quiere eliminar el centro. Al pulsar en aceptar se manda al servidor la petición de eliminación.

En el servidor se comprobará si el centro existe. Si no existe se devolverá un error 404 mientras que si existe se comprobará si el centro tiene titulaciones que dependan de él. En caso afirmativo se redireccionará al usuario a la ventana principal de centros indicando que no se puede eliminar el centro porque tiene titulaciones asociadas. En caso de que el centro no tenga titulaciones asociadas, se eliminará de la base de datos y se redireccionará al usuario a la vista principal de centros indicando en un mensaje que el centro se ha eliminado correctamente.

- **Asignar plaza a docente**

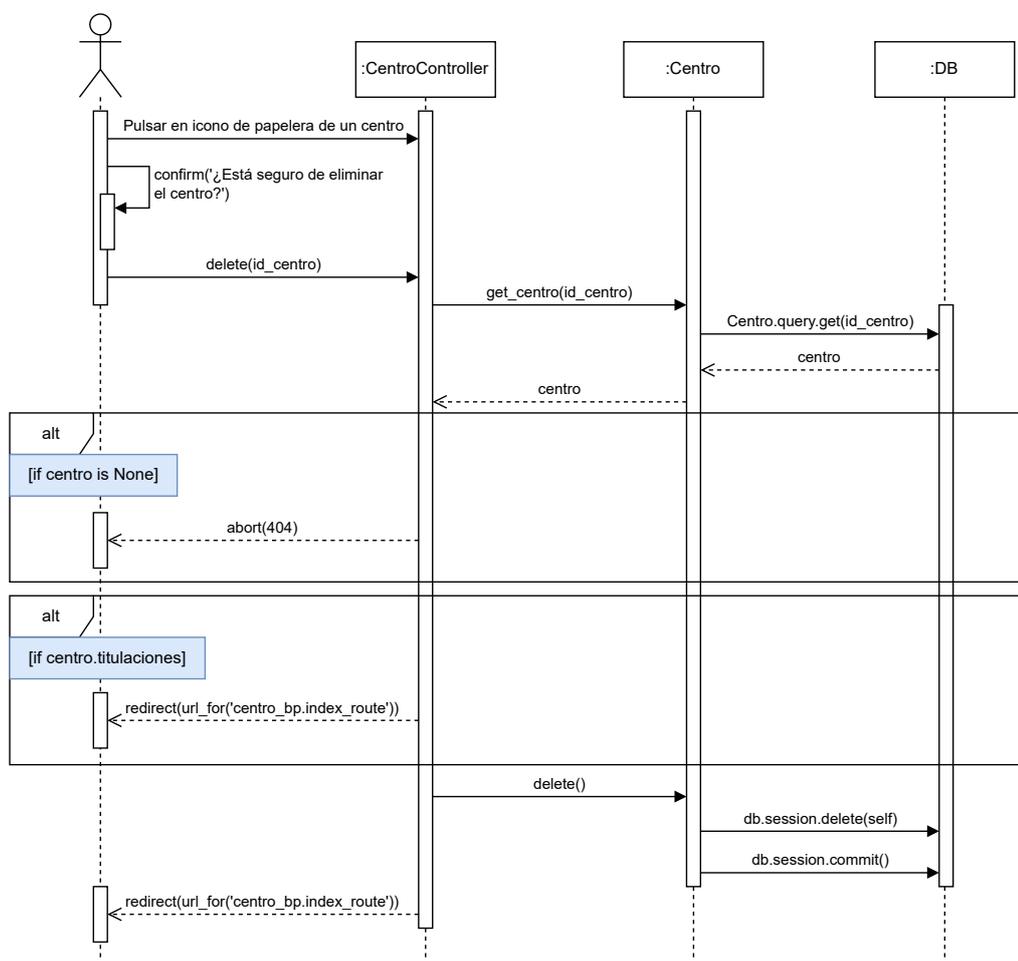


Figura C.5: Diagrama de secuencia de la eliminación de centros.

En el diagrama C.6 se puede ver el conjunto de pasos seguidos para asignar una plaza a un docente. Para este caso se supone que se tiene una plaza y un docente creados previamente.

El cliente pulsa en el icono del lápiz de una de las plazas. Esto abre el formulario de edición de la plaza desde donde puede seleccionar el docente a través del campo de búsqueda de docentes escribiendo el nombre del mismo. Con el docente seleccionado se pulsa en «Modificar».

Si todos los campos del formulario son correctos, se comprueba si el docente tiene alguna plaza sin fecha de cese, en ese caso no se puede

vincular con esta plaza y se mostrará un mensaje informando del error. En caso contrario, la plaza se asigna.

Finalmente, los cambios se guardan en la base de datos y se redirecciona al usuario a la vista de plazas donde se muestran los mensajes correspondientes a la correcta modificación de la plaza y, si no ha sido posible asignar la plaza por culpa de poseer alguna plaza sin fecha de cese, también se muestra en un mensaje de error. En caso de no haber pasado la validación del formulario, se redirecciona a la ventana del formulario mostrando los errores detectados.

■ **Creación de curso académico**

El diagrama C.7 contiene los pasos necesarios para crear un nuevo curso académico.

El usuario parte de la ventana de cursos donde debe pulsar sobre «Nuevo». Esto abre una ventana donde se indica el año de inicio del nuevo curso a crear y se pulsa sobre «Añadir».

Con el paso anterior realizado, el usuario se encuentra en la ventana donde puede seleccionar el número de alumnos de cada modalidad y las asignaturas que quiere incluir al curso. Después de realizar la selección deseada pulsa sobre «Añadir». Esto hará que se cree la vinculación entre el curso y aquellas asignaturas que no están ya vinculadas. Además, se crea un grupo de tipo presencial y otro de tipo práctico por cada asignatura.

En caso de producirse algún error con el formulario de introducción de año, como por ejemplo añadir un año de un curso existente, se devolverá al usuario a la ventana del formulario indicando el error. Lo mismo ocurre en el formulario de selección de asignaturas, si el formulario no se valida correctamente, se le mostrará el error al usuario en el propio formulario.

Si al final todo ha ido bien, el usuario será redirigido a la ventana principal de cursos donde podrá ver el curso creado y un mensaje indicando la correcta creación y asignación de asignaturas realizada.

■ **Creación de grupo**

En el diagrama C.8 se muestra el proceso seguido para crear un nuevo grupo en una asignatura vinculada a un curso académica. Se parte de la base de que existe un curso con al menos una asignatura y que el usuario se encuentra en la ventana de grupos donde aparece un listado de las asignaturas del curso académico.

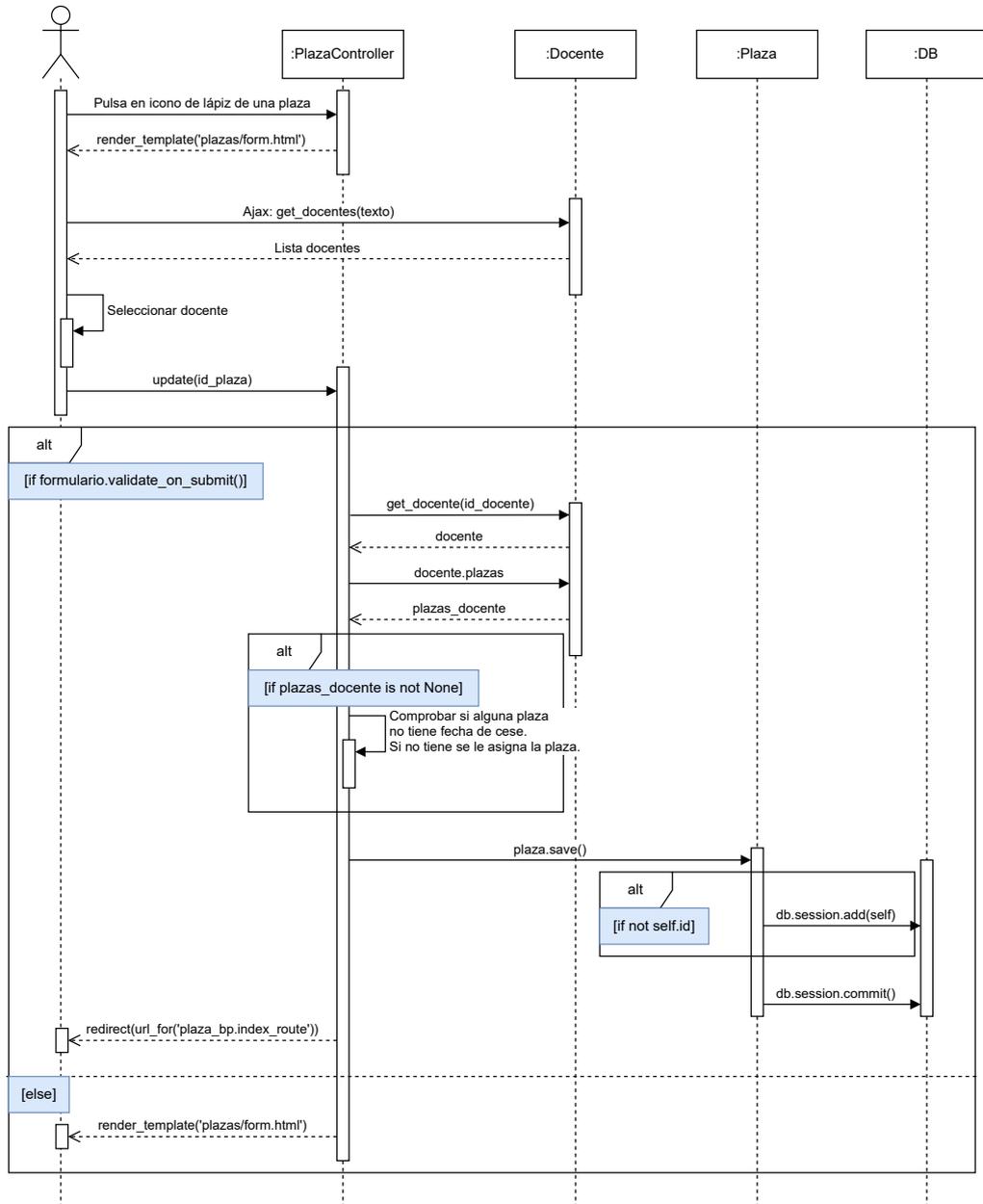


Figura C.6: Diagrama de secuencia de la asignación de una plaza a un docente.

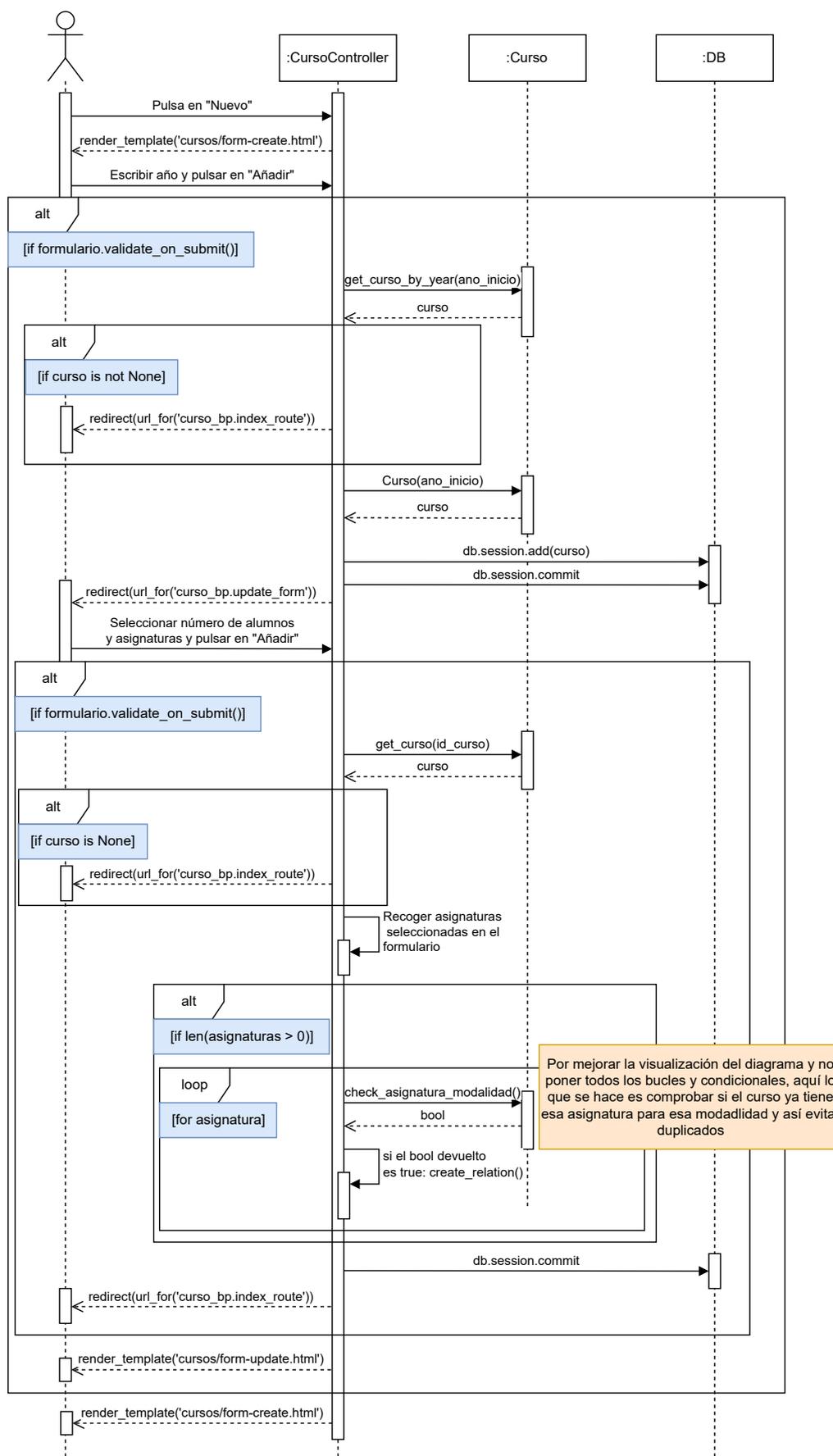


Figura C.7: Diagrama de secuencia de la creación de un curso académico.

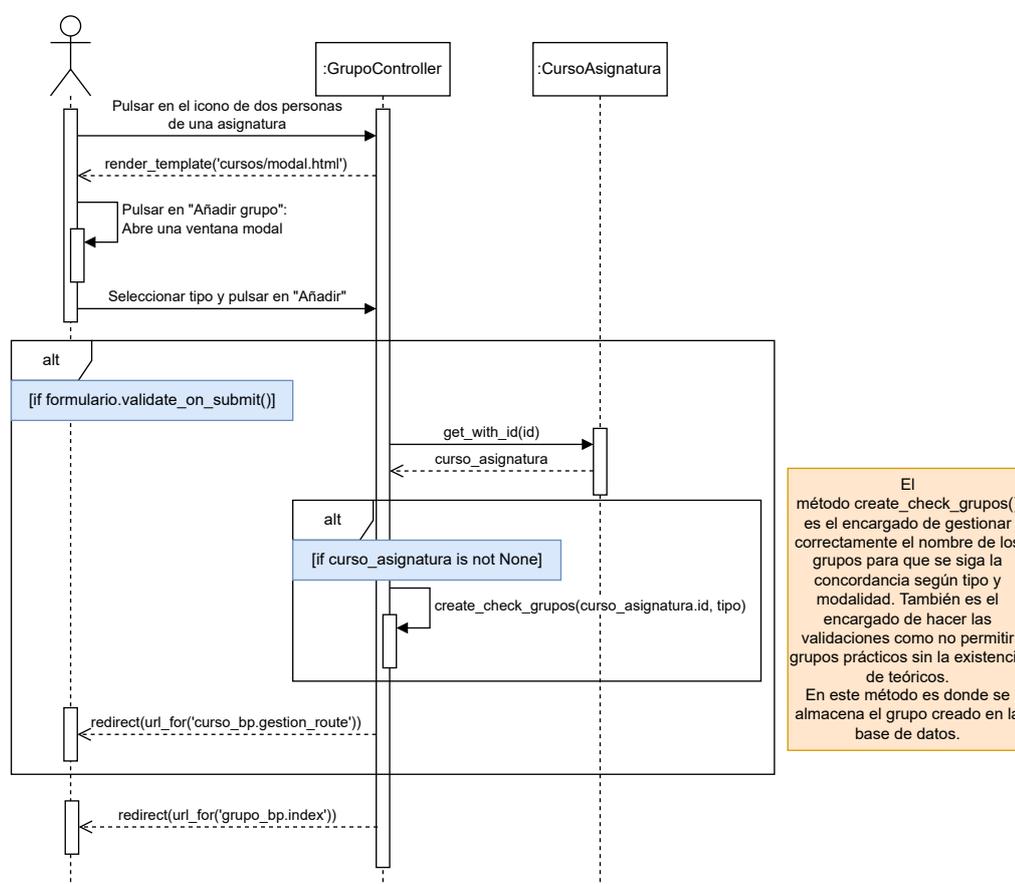


Figura C.8: Diagrama de secuencia de creación de grupos.

El usuario comienza el proceso pulsando sobre el icono de dos personas de una asignatura del listado. De esta manera será redirigido a la vista de los grupos de la asignatura. Acto seguido pulsa sobre el botón «Añadir grupo», lo que abre una ventana modal donde podrá seleccionar entre las modalidades teórico o presencial para el nuevo grupo a crear. Una vez seleccionada esta opción, pulsa sobre el botón «Añadir» y la ventana modal se cerrará. Además, aparecerá un mensaje informando de la creación y se podrá ver el grupo en la lista.

■ Asignación de horas de una plaza a un grupo

El diagrama C.9 representa la secuencia seguida para vincular una plaza a un grupo indicando el número de horas de docencia que va a impartir.

Se parte de que el usuario se encuentra en la ventana llamada «Horas» y que existe al menos un grupo y una plaza en la aplicación.

El usuario pulsa sobre el icono del reloj de uno de los grupos de la lista que aparece en la pantalla. Esto abre una nueva ventana donde se pueden ver todos las plazas que cuentan con horas en ese grupo. El usuario pulsa en «Añadir Plaza», lo que abre una ventana modal con un buscador mediante Ajax de la plaza y un campo para indicar las horas. El usuario selecciona una plaza e indica las horas y pulsa en «Añadir».

De forma interna, el controlador de la plaza realiza una búsqueda de la plaza y grupo para asegurarse de que existen, con esta comprobación hecha se recogen las plazas del grupo. Si la plaza que se quiere asignar ya se encuentra en ese grupo no se permite, en caso contrario la plaza se vincula con las horas indicadas.

C.4. Diseño arquitectónico

El diseño arquitectónico define la estructura del *software* creado respecto a la distribución de las clases y el patrón de diseño utilizado.

En este proyecto se ha seguido la arquitectura cliente-servidor bajo el patrón de diseño Modelo-vista-controlador (MVC). Flask, que ha sido el *framework* elegido para realizar el código de la aplicación, no obliga al uso de este patrón; sin embargo, se decidió utilizarlo se decidió utilizarlo debido a la organización que proporciona. Esto permite que el código quede separado de una forma fácil de entender, de modo que cualquier programador pueda identificar claramente la ubicación de cada elemento.

- **Modelo:** Es el encargado del manejo de los datos de la aplicación.
- **Vista:** Es la representación de los datos que ve el cliente de la web. En este caso mediante código HTML bajo el uso del motor de plantillas Jinja2.
- **Controlador:** Es la parte de la aplicación donde se encuentra la parte importante de la lógica. Actúa de intermediario entre el cliente y los datos, recogiendo las peticiones de los clientes y haciendo las peticiones necesarias a los modelos para procesar la información requerida y devolver la respuesta al cliente a través de la vista.

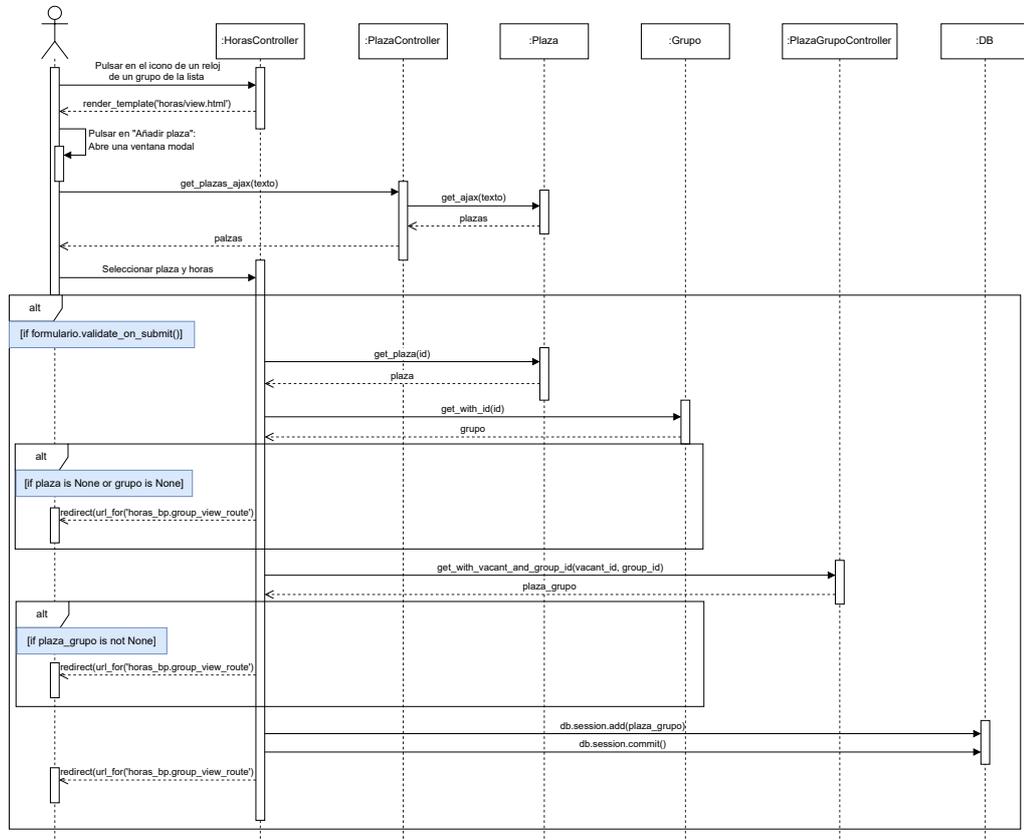


Figura C.9: Diagrama de secuencia de la asignación de horas de una plaza a un grupo.

En el diagrama C.10 se representa el funcionamiento del MVC donde el usuario de la web interactúa, a través de la vista, con el controlador que se encarga de hacer peticiones y respuestas de datos con el modelo. De esta forma el controlador es capaz de recuperar, modificar o eliminar los datos de un modelo. Además, el controlador es el encargado de formar el código necesario para mostrar la vista que visualizará el usuario de la web.

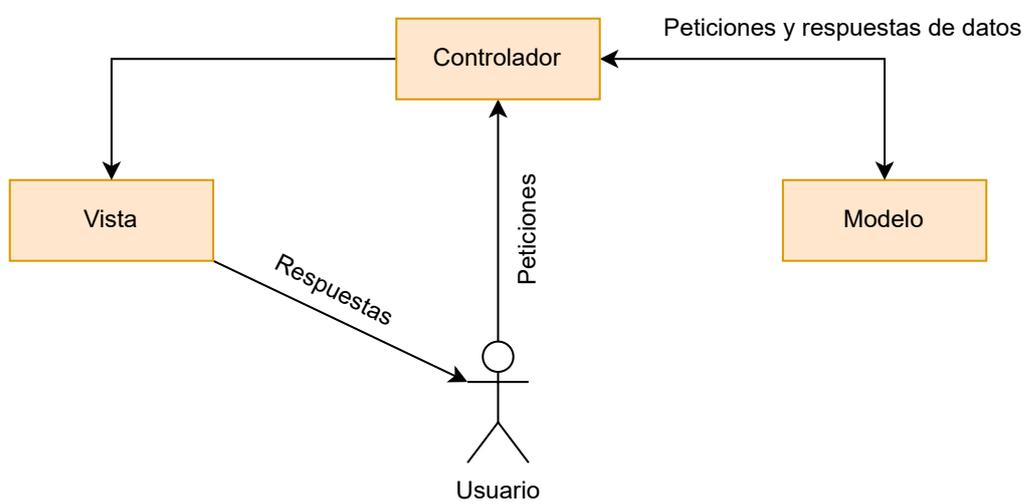


Figura C.10: *Modelo-Vista-Controlador*

Apéndice *D*

Documentación técnica de programación

D.1. Introducción

En este anexo se muestra toda la información necesaria para que un desarrollador pueda poner en funcionamiento el *software* de este proyecto. Además, se da una explicación sobre la estructura del código y aquellos aspectos que puedan ser importantes para cualquier desarrollador que vea por primera vez el código desarrollado.

D.2. Estructura de directorios

La estructura de los directorios del proyecto es la siguiente:

- **docs:** Contiene la documentación del proyecto.
- **src:** Dentro de este directorio se encuentra el código fuente del proyecto desarrollado.

Este anexo va a centrarse en la explicación del contenido del directorio **src**. En la figura [D.1](#) se puede ver la distribución de los directorios que se encuentran dentro de **src**.

A continuación se da una explicación de lo contenido en cada directorio:

- **src:** Es el directorio principal del código fuente del proyecto que contiene a todos los demás. Además, en su interior se encuentra el

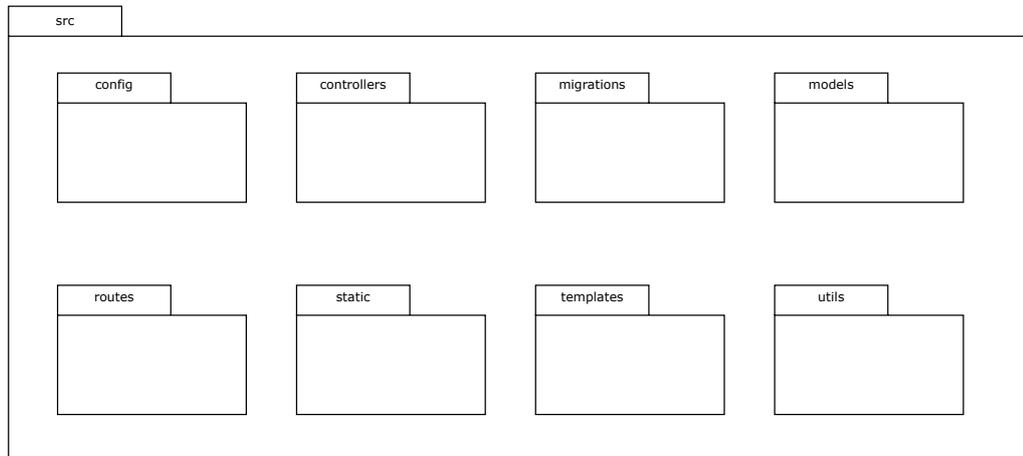


Figura D.1: Directorios del proyecto

fichero `app.py` desde el que se carga la configuración y se arranca la aplicación, el fichero `forms.py` que contiene todos los formularios generados a partir de la biblioteca WTForms y el fichero `decorators.py` que contiene aquellas funciones encargadas de detectar los permisos del usuario activo y su `token`. Por otro lado, también se encuentran los ficheros `Procfile` necesario para desplegar la aplicación en Heroku, `requirements.txt` que contiene todas las bibliotecas junto a su versión requerida y el fichero `runtime.txt` también necesario para desplegar en Heroku donde se indica la versión de Python.

- **config**: Contiene tres ficheros necesarios para la configuración del proyecto. El fichero `config.py` indica la configuración general del proyecto mientras que los ficheros `development.py` y `production.py` contienen la configuración necesaria para el despliegue en modo desarrollo y modo producción respectivamente.
- **controllers**: En este directorio se encuentran todos los controladores del proyecto.
- **migrations**: Es un directorio perteneciente a la librería Flask-Migrate. Este directorio es utilizado para almacenar la configuración de la biblioteca y las diferentes migraciones creadas para actualizar la base de datos evitando la pérdida de información almacenada. (En principio, no sería necesario **tocar** nada de este directorio).

- **models**: Aquí se encuentran todos los ficheros respectivos a los modelos de la aplicación web. Es decir, aquellos objetos que se crean y almacenan en la aplicación. Desde cada uno de estos ficheros también se define la estructura de sus tablas en la base de datos.
- **routes**: Este directorio almacena los diferentes ficheros que definen las rutas de la aplicación, organizadas por *blueprints*.

En estos ficheros se indican los decoradores de cada ruta y la acción del controlador asociada que se debe ejecutar con la petición a esa ruta. Además, se define el tipo de petición HTTP permitida por ruta.

- **static**: Como su nombre indica, este directorio contiene todos aquellos ficheros estáticos de la aplicación que van a ser cargados por las vistas. Los ficheros que contiene son accesibles desde un navegador y aquí se almacenan imágenes, ficheros CSS, ficheros de JavaScript y las librerías de JavaScript cargadas desde el sistema de gestión de paquetes NPM, junto al fichero de configuración donde se indican las bibliotecas y su versión llamado `package.json`.
- **templates**: Este directorio agrupa todas aquellas vistas de la aplicación web. Dentro se encuentra dividido a su vez por otros directorios con el nombre de los diferentes modelos para tener más organizadas las vistas.

En este directorio también se encuentra el fichero `base_template.html` que define la plantilla base de todas las demás vistas. Desde este fichero se define la carga de los ficheros CSS y *scripts* de JavaScript, además de otros elementos como el menú de la aplicación.

En este fichero todos los elementos se encuentran divididos por bloques para cargar solo aquellos necesarios. Por ejemplo, el footer solo se carga en la vista de *login*, pero el bloque de carga se define aquí.

- **utils**: Este directorio está creado para mejorar la carga y organización de la aplicación. Dentro del directorio se encuentra el fichero `db.py` donde se cargan las bibliotecas relativas a la base de datos como SQLAlchemy o Flask-Migrate.

D.3. Manual del programador**D.4. Compilación, instalación y ejecución del proyecto****D.5. Pruebas del sistema**

Apéndice E

Documentación de usuario

- E.1. Introducción
- E.2. Requisitos de usuarios
- E.3. Instalación
- E.4. Manual del usuario

Bibliografía

- [1] Apache. Licencia apache, versión 2.0. <https://www.apache.org/licenses/LICENSE-2.0>, 2023. [Internet].
- [2] GNU. Gnu library general public license, version 2.0. <https://www.gnu.org/licenses/old-licenses/lgpl-2.0.html>, 2023. [Internet].
- [3] Wikipedia. Licencia bsd — wikipedia, la enciclopedia libre. https://es.wikipedia.org/w/index.php?title=Licencia_BSD&oldid=146098617, 2022. [Internet; descargado 21-septiembre-2022].
- [4] Wikipedia. Licencia mit — wikipedia, la enciclopedia libre. https://es.wikipedia.org/w/index.php?title=Licencia_MIT&oldid=151165122, 2023. [Internet; descargado 15-mayo-2023].
- [5] Wikipedia contributors. Isc license — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=ISC_license&oldid=1151785929, 2023. [Online; accessed 13-June-2023].