



Universidad de Valladolid

ESCUELA DE INGENIERÍA INFORMÁTICA

Trabajo de fin de grado

**REDISEÑO Y REVISIÓN DE
PINTIADATA IMPLANTANDO EL
FRAMEWORK LARAVEL**

Grado en ingeniería informática

Mención en software

Autor: Rodrigo Núñez Paúl

Tutor: Joaquín Adiego Rodríguez

Resumen

Los catálogos arqueológicos digitales son el resultado del proceso de clasificación y documentación de las diferentes piezas y hallazgos encontrados en los yacimientos arqueológicos. Pintiadata es un proyecto concebido como herramienta de trabajo para el desarrollo de las investigaciones realizadas en el Centro de Estudios Vacceos 'Federico Wattenberg' de la Universidad de Valladolid situado, pero también como herramienta de consulta para que los resultados de estas investigaciones puedan ser accedidas de forma pública.

En este Trabajo de Fin de Grado se ha hecho una actualización del sitio web, integrándolo en un framework siguiendo patrones de diseño y añadiendo nuevas funcionalidades, así como el rediseño de algunas interfaces. Finalmente el sistema ha sido validado y verificado.

Tabla de contenidos

Resumen	I
Lista de tablas	IV
Lista de figuras	VI
1 Introducción y objetivos	1
1.1 Situación actual y motivación del trabajo	1
1.2 Objetivos	1
1.3 Estructura del documento	2
2 Estado del Arte	4
2.1 Catálogos arqueológicos	4
2.2 Servidores Web	4
2.3 Sistemas Gestores de Bases De Datos	6
2.4 HTML	7
2.5 CSS	9
2.6 JavaScript, jQuery y AJAX	10
2.6.1 JavaScript	10
2.6.2 jQuery	10
2.6.3 AJAX	10
2.7 PHP	11
2.8 Structured Query Language (SQL)	11
2.8.1 DDL	11
2.8.2 DML	11
2.9 Frameworks	12
2.9.1 Ventajas del uso de un framework	12
2.9.2 Desventajas del uso de un framework	12
2.10 Framework web Laravel	13
3 Análisis	14
3.1 Especificación de requisitos	14
3.1.1 Requisitos funcionales	14
3.1.2 Requisitos no funcionales	22
3.2 Actores del sistema	23

3.3	Diagrama de Casos de uso	24
3.4	Realización de casos de uso	26
3.5	Modelo de dominio	46
4	Arquitectura y diseño	48
4.1	Patrón MVC	48
4.1.1	Componentes principales	48
4.1.2	Flujo de control básico del patrón MVC	49
4.2	Active Record	49
4.3	Funcionamiento de una petición web en Laravel	50
4.4	Diagramas de secuencia	51
4.5	Diseño de la base de datos	54
4.5.1	Diagrama entidad relación de la base de datos	54
4.5.2	Descripción de las tablas	56
4.6	Rediseño de las interfaces	58
4.6.1	Rediseño del catálogo de las fichas de objetos	58
4.6.2	Rediseño de la lista de elementos multimedia	59
5	Implementación	61
5.1	Corrección de errores	61
5.2	Mejoras realizadas	62
5.3	Estructura de un proyecto Laravel	64
5.4	Control de versiones	65
5.4.1	Git	66
6	Pruebas	67
6.1	Introducción	67
6.2	Pruebas de caja negra	68
7	Planificación del proyecto	81
7.1	Gestión de los riesgos	81
7.1.1	Caracterización de los riesgos	81
7.1.2	Identificación de los riesgos y plan de acción	82
7.2	Plan de trabajo	84
7.2.1	Diagrama de Gantt	84
8	Conclusiones	89
8.1	Conclusiones finales del trabajo	89
8.2	Trabajo futuro	89
	Bibliografía	91
	Anexos	93

Lista de tablas

3.1. Descripción del CU-01	27
3.2. Descripción del CU-02	27
3.3. Descripción del CU-03	28
3.4. Descripción del CU-04	28
3.5. Descripción del CU-02	29
3.6. Descripción del CU-06	29
3.7. Descripción del CU-07	30
3.8. Descripción del CU-08	31
3.9. Descripción del CU-09	32
3.10. Descripción del CU-10	33
3.11. Descripción del CU-11	34
3.12. Descripción del CU-12	35
3.13. Descripción del CU-13	36
3.14. Descripción del CU-14	37
3.15. Descripción del CU-15	38
3.16. Descripción del CU-16	39
3.17. Descripción del CU-17	40
3.18. Descripción del CU-18	41
3.19. Descripción del CU-19	42
3.20. Descripción del CU-20	43
3.21. Descripción del CU-21	44
3.22. Descripción del CU-22	45
3.23. Descripción del CU-23	45
3.24. Descripción del CU-24	46
6.1. Pruebas de caja negra asociadas a un usuario web I	68
6.2. Pruebas de caja negra asociadas a un usuario web II	69
6.3. Pruebas de caja negra asociadas a un usuario regular	69
6.4. Pruebas de caja negra asociadas a un usuario arqueólogo novel I	70
6.5. Pruebas de caja negra asociadas a un usuario arqueólogo novel II	71
6.6. Pruebas de caja negra asociadas a un usuario arqueólogo novel III	72
6.7. Pruebas de caja negra asociadas a un usuario arqueólogo novel III	73
6.8. Pruebas de caja negra asociadas a un usuario arqueólogo experto I	74
6.9. Pruebas de caja negra asociadas a un usuario arqueólogo experto II	75

6.10. Pruebas de caja negra asociadas a un usuario arqueólogo experto III . . .	76
6.11. Pruebas de caja negra asociadas a un usuario arqueólogo experto IV . . .	77
6.12. Pruebas de caja negra asociadas a un usuario arqueólogo experto V . . .	78
6.13. Pruebas de caja negra asociadas a un usuario arqueólogo experto VI . . .	79
6.14. Pruebas de caja negra asociadas a un usuario administrador	80
7.1. Descripción del riesgo R01	82
7.2. Descripción del riesgo R02	82
7.3. Descripción del riesgo R03	83
7.4. Descripción del riesgo R04	83
7.5. Descripción del riesgo R05	83
7.6. Descripción del riesgo R06	84

Lista de figuras

1.	Petición – Respuesta entre Cliente – Servidor.	5
2.	Actores del sistema	23
3.	Actores del sistema	25
4.	Actores del sistema	26
5.	Modelo de dominio	47
6.	Modelo Vista Controlador	49
7.	Patrón Active Record	50
8.	Diagrama de comunicación de una petición web en Laravel	51
9.	Diagrama de secuencia del CU-08 Crear objeto	51
10.	Diagrama de secuencia del CU-04 Mostrar información de un objeto	52
11.	Diagrama de secuencia del CU-09 Gestionar un objeto	52
12.	Diagrama de secuencia del CU-10 Borrar un objeto del catálogo	53
13.	Diagrama de secuencia del CU-05 Realizar búsqueda con filtros	53
14.	Diagrama entidad relación	55
15.	Listado antiguo de las fichas de objetos	59
16.	Listado nuevo de las fichas de objetos	59
17.	Listado antiguo de los elementos multimedia	60
18.	Listado nuevo de los elementos multimedia	60
19.	Elementos duplicados en menús de selección	62
20.	Lista de fases y actividades del proyecto	85
21.	Diagrama de Gantt con la fase de inicio y análisis	86
22.	Diagrama de Gantt con la fase de diseño, implementación, despliegue y documentación	87
23.	Diagrama de Gantt con la replanificación de la fase de Diseño y la fase de Implementación	88

1 Introducción y objetivos

En este capítulo se presenta la situación actual y motivaciones que han llevado a desarrollar el trabajo, así como las metas y objetivos que se han decidido llevar a cabo. También se muestra la estructura de los capítulos de este documento para facilitar su comprensión.

1.1. Situación actual y motivación del trabajo

Una versión preliminar de la página web Pintiadata fue desarrollada como parte de un Trabajo de Fin de Grado anterior[1] con el fin de facilitar la gestión y la consulta del catálogo arqueológico. En la anterior versión se rediseñaron las interfaces y se implementaron nuevas funcionalidades para que los arqueólogos pudieran desarrollar correctamente su trabajo. Aunque se corrigieron errores, el código fuente no estaba optimizado para que fuese mantenido en el futuro, además de la limitación de algunas vistas.

El servidor se asienta sobre una infraestructura LAMP usando las siguientes herramientas: Linux como sistema operativo, Apache como servidor web, MySQL como SGBD, PHP como lenguaje de programación para el servidor y HTML y JavaScript como lenguajes ejecutándose en el cliente.

Teniendo en cuenta lo anterior, es necesario realizar una actualización de la web en su anterior versión, utilizando para ello un framework basado en PHP con el que se van a llevar a cabo el rediseño de algunas interfaces, la verificación y validación de la página para corregir errores y la implementación de nueva funcionalidad para simplificar el mantenimiento de la aplicación en el futuro.

1.2. Objetivos

En este trabajo se propone el estudio de los diferentes principios, reglas y estándares de diseño y usabilidad para la posterior integración de un framework web para la ampliación de la funcionalidad, facilitar la comprensión y el mantenimiento del código, rediseño de la interfaz, validación, verificación y prueba del catálogo online del sitio arqueológico de Pintia. Los objetivos principales se detallan a continuación:

- Integración de un framework web para facilitar el mantenimiento del software de la aplicación.
- Añadir mejoras y corregir errores de la anterior versión, para mejorar la usabilidad de la aplicación.
- Rediseño de algunas interfaces de usuario para fomentar el uso del portal a usuarios no pertenecientes al sistema y potenciar as aspecto visual de la aplicación.
- Validación, verificación, y prueba de la página web, realizando aquellas correcciones que sean oportunas.

1.3. Estructura del documento

El siguiente apartado describe la realización del Trabajo de Fin de Grado y está estructurado de la siguiente forma:

Capítulo 1: Introducción y objetivos : Se explica la situación actual del sistema del que se parte, así como los motivos y objetivos que persigue este trabajo, junto con los acrónimos empleados en él.

Capítulo 2: Estado del arte : En este capítulo se realiza un estudio del entorno tecnológico que rodea al sistema web y las bases teóricas de los patrones de diseño utilizados.

Capítulo 3: Análisis : Se enumeran los requisitos funcionales, los requisitos funcionales de información y requisitos no funcionales del sistema para a partir de ellos extraer los casos de uso del sistema, junto con la explicación de los actores implicados en ellos.

Capítulo 4: Arquitectura y diseño : Se describen los diagramas de secuencia correspondientes a las operaciones más importantes del sistema, los patrones de diseño utilizados en el desarrollo de la aplicación y el rediseño de interfaces llevado a cabo.

Capítulo 5: Implementación Se muestran aspectos de implementación básicos del proyecto de la aplicación, así como la estructura del código, además de los errores corregidos y las mejoras implementadas.

Capítulo 6: Pruebas Se detallan las pruebas llevadas a cabo en el sistema para verificar su funcionamiento.

Capítulo 7: Planificación del proyecto Este capítulo está dedicado a cómo se ha llevado a cabo la planificación del proyecto, con las actividades de las que se compone, así como un plan de riesgos.

Capítulo 8: Conclusiones En este apartado se hace balance del trabajo realizado, los objetivos conseguidos y posibles mejoras futuras.

2 Estado del Arte

Este apartado se centra en el estudio de las bases teóricas sobre las que se sustenta este trabajo, como son los Servidores Web, los frameworks web, los Sistemas Gestores de Bases de Datos y los patrones de diseño utilizados, con los cuales se pretende explicar los motivos de las decisiones tomadas en el desarrollo del trabajo. También se explican los diferentes lenguajes y tecnologías empleadas para llevar a cabo la implementación.

2.1. Catálogos arqueológicos

Los catálogos arqueológicos son el resultado de la clasificación de todos los elementos de un yacimiento arqueológico en función de sus diferentes características. Pueden ser internos, para ser usados como herramientas de trabajo en investigaciones arqueológicas, o públicos, los cuales tienen como finalidad dar a conocer los diferentes elementos resultantes de las investigaciones de forma pública, facilitando su localización.

En la actualidad existen diversos archivos y catálogos arqueológicos, como por ejemplo el Catálogo online del Museo Británico[2], el catálogo del Museo Arqueológico de Atenas[3] o el Museo Arqueológico Nacional de España[4]

2.2. Servidores Web

Un servidor web[5] es un programa informático que procesa una aplicación del lado del servidor, realizando conexiones bidireccionales o unidireccionales y síncronas o asíncronas con el cliente y generando o cediendo una respuesta en cualquier lenguaje o Aplicación del lado del cliente. El código recibido por el cliente es renderizado generalmente mediante un navegador web. Para transmitir los datos se suele utilizar algún protocolo siendo HTTP el más habitual para este tipo de comunicaciones.

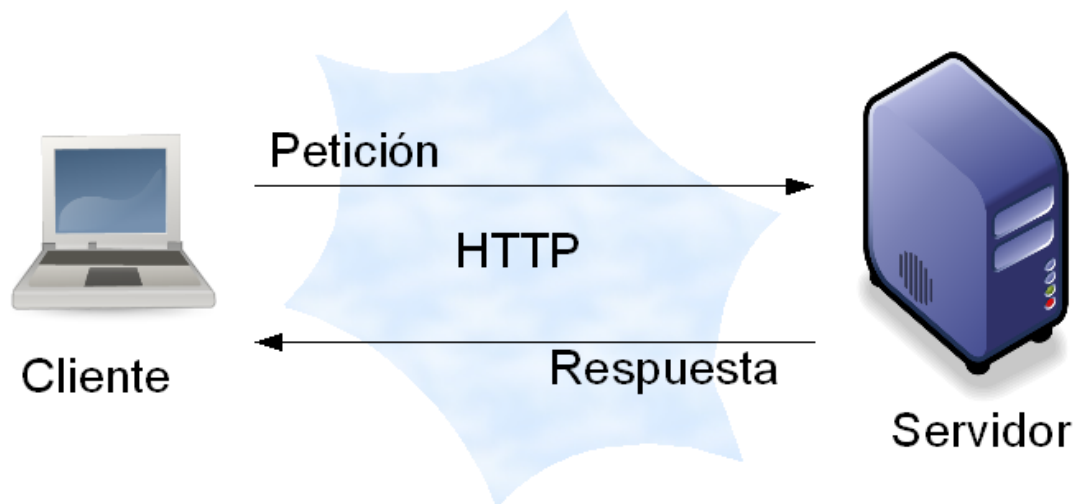


Figura 1: Petición – Respuesta entre Cliente – Servidor.

Cuando un cliente realiza una petición, el servidor la recibe y responde a ella de forma adecuada generando como respuesta una página web que se mostrará en el navegador web o mostrando un mensaje de error si lo hubiera.

En este proceso de petición – respuesta que se lleva a cabo entre cliente y servidor - se envía la información a través de la web. Los tipos de peticiones HTTP más habituales son GET, POST, PUT y DELETE.

GET : Pide al servidor el recurso especificado. Se recomienda por seguridad, que no se transmita información sensible con esta petición, ya que se agregan los parámetros a la URL, siendo visibles en la interfaz de usuario del navegador web.

POST : envía datos para que sean procesados en el servidor. Los datos se incluirán en el cuerpo de la petición. Se suele utilizar para envío de datos mediante formularios.

PUT : Se utiliza sobre todo para subir o cargar un recurso en el servidor, o actualizar el estado de un determinado recurso.

DELETE : borra el recurso especificado.

Durante el ciclo de vida de la petición, por lo general, se ejecutan aplicaciones web, tanto del lado del cliente como del lado del servidor:

Aplicaciones del lado del cliente Es en el cliente web, a través del navegador web donde se ejecuta el código proporcionado por el servidor. Es el caso del código HTML, hojas de estilos CSS y JavaScript.

Aplicaciones del lado del servidor El servidor web es el encargado de ejecutar las aplicaciones. Una vez procesada la petición, genera un código de respuesta, el cual es enviado al cliente. Las aplicaciones de servidor están escritas en lenguajes como PHP, Ruby o Python.

En nuestro caso, se ha usado Apache como servidor web ya que es de distribución libre y soporta una gran variedad de sistemas operativos.

Además, existen diversos paquetes de software que permiten instalar de forma conjunta servidores web Apache y otras herramientas adicionales como el gestor de Bases de Datos MySQL y PHP (entre otros). De esta forma, reciben el nombre LAMP aquellos paquetes instalados en plataformas Linux (tecnología empleada para el desarrollo de la web de este trabajo), WAMP cuando se instalan en Windows y MAMP en sistemas MAC OS.

Los servidores web pueden ser locales o remotos. Los servidores web locales son aquellos que residen en un equipo perteneciente a una red local, y que por tanto sólo son accesibles por los equipos de dicha red. Por otra parte, los servidores web remotos tienen un alcance global y no se limitan a una red local.

Para la fase de implementación y pruebas se eligió un paquete WAMP, llamado Laragon que trabaja con varios frameworks de PHP y que permite la instalación de los mismos de manera intuitiva y sencilla.

2.3. Sistemas Gestores de Bases De Datos

Un sistema gestor de base de datos[6] es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información contenida en una base de datos, además de proporcionar herramientas para añadir, borrar, modificar y analizar los datos.

Toda base de datos soportada por un SGBD debe tener un modelo de datos. Este es un conjunto de estructuras descriptivas de datos de alto nivel que oculta muchos detalles de almacenamiento de bajo nivel. De esta forma, el SGBD permite a los usuarios definir los datos que se van a almacenar en términos de este modelo de datos. Existen diversos modelos, aunque la mayor parte de sistemas actuales utilizan el modelo relacional. En

este, los datos se almacenan en tablas formadas por filas (tuplas) y columnas (atributos). Las tablas pueden relacionarse entre sí mediante claves comunes.

Estos sistemas también proporcionan métodos para mantener la integridad de los datos, para administrar el acceso de usuarios a los datos y para recuperar la información si el sistema se corrompe. Generalmente se accede a los datos mediante lenguajes de consulta, como SQL que es uno de los más reconocidos y extendidos, lenguajes de alto nivel que simplifican la tarea de construir las aplicaciones. También simplifican las consultas y la presentación de la información.

2.4. HTML

HTML[7] Es el lenguaje de marcado para la elaboración de contenidos en las páginas web, siendo de hecho el estándar que se ha impuesto en la visualización de páginas web y el que todos los navegadores actuales han adoptado. La sintaxis se describe con «etiquetas», rodeadas por corchetes angulares (<, >, /).

Los componentes principales son los elementos los cuales a su vez se componen de atributos y contenido. De forma general esta sería la estructura de un elemento.

```
<nombre-de-elemento atributo="valor">Contenido</nombre-de-elemento>
```

En cuanto a la estructura de un documento HTML, hay tres secciones bien diferenciadas:

DOCTYPE : en esta línea se especifica la versión de HTML utilizada. Ahora mismo la más extendida es HTML5

HEAD : se indican los metadatos del documento como el título y descripción de la página o elementos relacionados que no se representarán visualmente en el navegador del usuario. Además es donde se suelen inyectar los estilos CSS que usará la página.

BODY : En esta zona se incluyen todas las etiquetas HTML que van a ser visualizadas en el navegador del usuario, generalmente utilizadas para estructuras los contenidos de una página.

Algunas de las etiquetas básicas son las siguientes:

<html> : define el inicio del documento y le indica al navegador que el código que viene a continuación debe ser interpretado como HTML.

<head> : contiene información importante de documento como la codificación de caracteres empleada o el título de la página.

<body> : el contenido principal del documento. Es la parte visual que mostrará el navegador.

<script> : incrusta un script en el documento, o bien se importa uno externo mediante el atributo 'src'. El atributo 'type' indica el tipo MIME del script. En HTML5 si es de tipo JavaScript no hace falta especificarlo ya que es el considerado por defecto.

<link> : sirve para vincular a la página web hojas de estilo externas o iconos.

<style> : para colocar el estilo interno de la página; ya sea usando CSS u otros lenguajes similares.

<h1>a <h6> : encabezados o títulos del documento de distinto tamaño.

<table> : define una tabla

<tr> : define una fila de una tabla

<td> : define una celda de una tabla

<div> : division o sección de la página.

**** : etiqueta que representa una imagen. Es necesario declarar el atributo 'src' que indica la ruta a la imagen, además de declarar el atributo 'alt' con un texto, en el caso de que el servidor no encontrara la imagen.

<a> : enlace, interno a un recurso interno de la web, o bien un recurso hacia externo a otro sitio web. Debe incluirse el atributo 'href' que indica la ruta al recurso.

`<form>` se utiliza para crear un formulario HTML que recoja la entrada de datos del usuario, para que sean procesados por el servidor.

Un formulario puede contener elementos de entrada como los campos de texto, casillas de verificación, botones de radio, botones de envío y mucho más.

El atributo 'action' indica la URL del servidor que se encarga de procesar dichos datos. Otro atributo útil es 'method', que indica cómo se envía la información. En HTML5 si no se incluye, por defecto se considera que la petición será de tipo GET

2.5. CSS

CSS[8] es un lenguaje de diseño gráfico para definir y crear la presentación (apariciencia visual y formato) de un documento estructurado escrito en un lenguaje de marcado. El principal objetivo de CSS es separar el contenido del documento de su presentación. Con esta separación se logra una mayor mantenimiento y control del código, haciendo más fácil la modificación de estilos. Las hojas de estilo están formadas por reglas de estilo, que tienen dos partes principales, el selector y la declaración. La declaración puede estar formada por uno o más pares clave-valor:

```
Selector { propiedad: valor; propiedad: valor; }
```

En CSS los selectores más utilizados son:

Selector de etiqueta : Selecciona todos los elementos de la página cuya etiqueta HTML coincide con el selector.

Selector de clase : para usar este selector se debe crear en un elemento HTML, el atributo class con un valor. Después, en la hoja de estilos se define la regla CSS, debiera llevar como selector el valor del atributo 'class' del elemento, precedido de un punto (.) , para que se apliquen correctamente las propiedades.

Los selectores de clase son imprescindibles para diseñar páginas web complejas, ya que permiten disponer de una precisión total al seleccionar los elementos. Además, estos selectores permiten reutilizar los mismos estilos para varios elementos diferentes.

Selector de ID : a veces es necesario aplicar estilos a un único elemento. Con el selector de ID podemos seleccionar un elemento a través del valor de su atributo 'id'. En este caso, como no puede haber el mismo id para dos elementos, se selecciona un único elemento. Para definir la regla CSS en el selector debemos introducir el valor del atributo 'id' precedido del símbolo (#)

Para utilizar un fichero una hoja de estilos se puede hacer de tres maneras distintas:

Estilo como atributo del elemento : Se añade el atributo `style` al elemento y se añaden las reglas que se deseen, y solo afectarían a ese elemento.

CSS interno : dentro del elemento `<style>` se definen las reglas. Afectaría a los elementos del documento HTML donde se ha definido.

CSS externo : a través del elemento `<link>`, se enlazan las hojas de estilo que se deseen. Las reglas CSS afectarían a los elementos de los ficheros HTML donde se ha enlazado la hoja de estilos

2.6. JavaScript, jQuery y AJAX

Para crear contenido dinámico en el sitio web, se han utilizado las siguientes tecnologías del lado del cliente.

2.6.1. JavaScript

JavaScript[9], es un lenguaje de programación interpretado, principalmente utilizado en el lado del cliente y que entre sus usos más comunes es su interacción con los elementos HTML del documento para crear contenido web dinámico. Dado que el código JavaScript se ejecuta en el navegador del usuario, las acciones del usuario pueden ser respondidas con rapidez.

2.6.2. jQuery

jQuery es una biblioteca de JavaScript ampliamente reconocida, que agiliza el desarrollo y simplifica la manipulación de los documentos del DOM. Asimismo, hay multitud de plugins implementados que pueden ser descargados y añadidos fácilmente al proyecto. En nuestro caso, se ha utilizado el plugin `jquery.easyPaginate` que permite la paginación de listas de elementos para cargar solamente una parte de dicha lista.

2.6.3. AJAX

AJAX [10], es una técnica de desarrollo web para crear aplicaciones interactivas. Permite, a través de peticiones asíncronas al servidor, que la página se actualice sin necesidad de recargarla, lo cual mejora la usabilidad de la página web.

Para ello, se declara la petición AJAX en un script donde se indica la petición (generalmente una URL), el servidor la procesa, y los datos que llegan del servidor se cargan en segundo plano sin interferir en el comportamiento de la página. Además es una técnica

soportada por múltiples plataformas lo cual hace que casi cualquier navegador soporte esta tecnología.

2.7. PHP

Es un lenguaje de programación de código abierto que es muy utilizado en el ámbito del desarrollo web. Aunque se pueda incrustar en código HTML, todo el código PHP se ejecuta en el lado del servidor, lo que hace que el usuario reciba el resultado, pero no puede acceder al código que lo generó.

Otra de sus ventajas es que se puede utilizar en casi cualquier sistema operativo y soporta la mayoría de servidores web, aparte de dar soporte a numerosos sistemas de gestión de bases de datos.

2.8. SQL

SQL[11], es un lenguaje específico para el manejo de Bases de Datos Relacionales, con el fin de acceder, manipular o borrar datos de las mismas. Está formado por dos lenguajes, que tiene diferentes propósitos:

2.8.1. DDL

Se encarga de gestionar la estructura de los esquemas de la base de datos. Entre las operaciones más importantes se encuentran:

CREATE : Permite la creación de nuevas bases de datos o tablas dentro de ellas.

ALTER : Permite modificar la estructura de una tabla, como la agregación o modificación de alguno de sus campos.

DROP : Elimina un objeto de la base de datos.

2.8.2. DML

Es el lenguaje proporcionado por el gestor para llevar a cabo las consultas y manipulaciones de datos necesarios sobre la base de datos, y que cumple con el modelo de datos.

SELECT : Permite consultar los datos almacenados en las tablas de la base de datos.

INSERT : Añade registros a una tabla de la base de datos

UPDATE : Modifica el valor de unos determinados registros de una tabla de la base de datos.

DELETE : Permite el borrado de los registros de las tablas de la base de datos.

2.9. Frameworks

Un framework [12] es un conjunto de clases cooperativas que construyen un diseño reutilizable para un tipo específico de software. El framework proporciona la arquitectura partiendo de sus clases abstractas y definiendo cómo van a cooperar para poder llevar a cabo esa reutilización. Así, el desarrollador realiza su aplicación implementando subclases a partir de las clases definidas por el framework.

Los frameworks se guían por un principio de la ingeniería de software denominado "principio de Hollywood"(no nos llame, ya le llamaremos). Esto se puede resumir en que las clases del framework son las que usarán las de la aplicación, en lugar de ser al revés como sucede en las bibliotecas.

En un framework, el flujo del programa está en el código del framework que realiza llamadas al código de la aplicación.

2.9.1. Ventajas del uso de un framework

Entre las ventajas que podemos encontrar en un framework:

Reducción de la codificación : el framework al proporcionar las clases y módulos que están verificados, reduce el código a mantener y sobrescribir por el desarrollador.

Proporciona una arquitectura para las aplicaciones : al ser una arquitectura proporcionada común, hace que sea fácil de aprender y mantener, ya que las clases que proporciona el framework ocultan su complejidad dando un nivel alto de abstracción.

2.9.2. Desventajas del uso de un framework

Poca flexibilidad : los componentes desarrollados a partir del framework deben residir dentro de las restricciones impuestas por el framework.

Dificultad de aprendizaje en fases iniciales : el usuario debe estudiar los servicios y clases que proporciona el framework para hacer un uso adecuado de él. Aunque

sea costoso inicialmente, las tareas de desarrollo se pueden agilizar en las etapas posteriores

2.10. Framework web Laravel

Laravel [13] es un PHP-framework de código abierto para desarrollar aplicaciones web, que fue creado por Taylor Otwell en 2011. La filosofía del mismo es desarrollar código de forma simple y legible. Actualmente se encuentra en su versión 5.5 de desarrollo. Este proyecto se ha desarrollado con la versión 5.3

Algunas de sus funcionalidades más importantes son:

Sistemas de rutas : URI para acceder a los recursos de la página web.

Motor de plantillas Blade : evita la incrustación de código PHP en las vistas lo que facilita un código mucho más limpio.

Soporte para MVC : se puede aplicar fácilmente el patrón Modelo-Vista-Controlador para su desarrollo, que veremos más adelante

Eloquent ORM : Facilita la creación de modelos que se van a comunicar con la base de datos.

Composer como gestor de dependencias : Composer[14] resuelve las dependencias que necesite el proyecto para descargarlas e instalarlas en nuestro proyecto.

Se eligió Laravel para desarrollar este proyecto, ya que lo conocía con anterioridad y es uno de los PHP-frameworks más en boga de los últimos años, superando a otros como Symfony o CodeIgniter. Además, su comunidad de usuarios no para de crecer por lo que la documentación y soporte es abundante[15].

3 Análisis

En el siguiente capítulo se desarrollan las diferentes tareas llevadas a cabo durante la fase de análisis del trabajo, lo cual implica el estudio y definición de los requisitos funcionales y no funcionales del sistema, tanto aquellos que ya se han alcanzado en el sistema sobre el que se parte como los nuevos requisitos, junto con los actores del sistema y un análisis de los casos de uso extraídos a partir de los requisitos.

3.1. Especificación de requisitos

Los requisitos son las condiciones que debe cumplir un sistema para satisfacer un contrato, una norma o una especificación.

3.1.1. Requisitos funcionales

Los requisitos funcionales definen los servicios que va a proporcionar el sistema y su comportamiento ante determinadas situaciones. A continuación se indican los requisitos funcionales de partida ya logrados, antes de comenzar este trabajo.

- El sistema deberá permitir el inicio de sesión.
- El sistema deberá permitir cerrar sesión del mismo.
- El sistema deberá permitir gestionar la cuenta personal del usuario.
- El sistema deberá tener varios niveles de usuarios con sus permisos y restricciones.
- El sistema deberá permitir el registro de nuevos usuarios.
- El sistema deberá permitir el borrado de usuarios.
- El sistema deberá proporcionar información sobre los usuarios del sistema.
- El sistema deberá permitir la administración de las cuentas de los usuarios.
- El sistema mostrará listas de: tumbas,inhumaciones,cremaciones,unidades estratigráficas,matrices de Harris,analíticas de faunas,artículos,autores, objetos multimedia,muestras,relaciones estratigráficas y análisis metalográficos.

- El sistema mostrará un listado de objetos visibles a aquellos usuarios que no estén registrados.
- El sistema permitirá crear objetos,tumbas,inhumaciones,cremaciones,unidades estratigráficas,matrices de Harris,analíticas de faunas,artículos,autores, objetos multimedia,muestras,relaciones estratigráficas y análisis metalográficos.
- El sistema permitirá gestionar objetos,tumbas,inhumaciones,cremaciones,unidades estratigráficas,matrices de Harris,analíticas de faunas,artículos,autores, objetos multimedia,muestras,relaciones estratigráficas y análisis metalográficos.
- El sistema permitirá el borrado de objetos,tumbas,inhumaciones,cremaciones,unidades estratigráficas,matrices de Harris,analíticas de faunas,artículos,autores, objetos multimedia,muestras,relaciones estratigráficas y análisis metalográficos.
- El sistema permitir deberá la búsqueda de objetos,tumbas,inhumaciones,cremaciones, autores,artículos,muestras,usuarios,registros,multimedias,analíticas de fauna,mensajes,unidades estratigráficas y localizaciones.
- El sistema permitirá crear notas asociadas a un objeto, una unidad estratigráfica y a una tumba.
- El sistema permitirá gestionar las notas asociadas a un objeto, una unidad estratigráfica y a una tumba.
- El sistema permitirá la creación de palabras clave, materias primas, tipos de tumbas, tipos de muestras, componentes artificiales, componentes geológicos, componentes orgánicos, artefactos y superficies.
- El sistema permitirá la gestión palabras clave, materias primas, tipos de tumbas, tipos de muestras, componentes artificiales, componentes geológicos, componentes orgánicos, artefactos y superficies.
- El sistema permitirá borrar palabras clave, materias primas, tipos de tumbas, tipos de muestras, componentes artificiales, componentes geológicos, componentes orgánicos, artefactos y superficies.
- El sistema permitirá asociar palabras clave a un artículo
- El sistema permitirá asociar componentes artificiales, componentes geológicos, componentes orgánicos, analíticas de fauna,matrices de Harris,superficies,arterfactos, una localización, muestras y fases a una unidad estratigráfica.
- El sistema permitirá asociar materiales, medidas,artículos, objetos multimedia, una localización, una unidad estratigráfica, un análisis estratigráfico y una tumba a un objeto.
- El sistema permitirá asociar tipos de tumba,cremaciones, inhumaciones, una localización, una UE, analíticas de fauna y objetos multimedia a una tumba.

- El sistema permitirá marcar/desmarcar campos pendientes de objetos, tumbas y unidades estratigráficas.
- El sistema permitirá gestionar medidas, categorías y subcategorías de objeto, lugares y localizaciones.
- El sistema permitirá el borrado de medidas, categorías y subcategorías de objeto, lugares y localizaciones.
- El sistema permitirá acceder a las imágenes de objetos multimedia en sus dimensiones originales.
- El sistema deberá proporcionar ayuda en tareas de gestión.
- El sistema mostrará un listado con los registros pendientes de validar.
- El sistema permitirá validar los registros asociados a objetos, tumbas e inhumaciones.
- El sistema permitirá el envío de correos electrónicos.
- El sistema permitirá el envío de mensajes entre los usuarios del sistema.
- El sistema permitirá imprimir la información relativa a un objeto.

Requisitos funcionales de información

Los requisitos funcionales de información indican la información que se va a almacenar en el sistema.

- El sistema deberá almacenar la información correspondiente a los objetos. En concreto:
 - (a) Referencia
 - (b) Visibilidad
 - (c) Año de campaña
 - (d) UE
 - (e) Localización
 - (f) Numero de Serie
 - (g) Tumba asociada
 - (h) Analisis estratigráfico asociado
 - (i) Multimedia asociado
 - (j) Almacen

- (k) Observaciones
 - (l) Decoración
 - (m) Forma
 - (n) Descripción
 - (ñ) Cronología
 - (o) Campos pendientes
 - (p) Materiales
 - (q) Partes asociadas
 - (r) Campos pendientes
 - (s) Notas asociadas
- El sistema deberá almacenar la información correspondiente a la parte de un objeto. En concreto:
 - (a) ID Parte
 - (b) ID Categoría
 - (c) ID Subcategoría
 - (d) Denominación
 - El sistema deberá almacenar la información correspondiente a las medidas de un objeto. En concreto:
 - (a) Siglas medida
 - (b) Denominación
 - (c) Unidades
 - El sistema deberá almacenar la información correspondiente a las tumbas. En concreto:
 - (a) ID Tumba
 - (b) UE asociada
 - (c) Cremaciones asociadas
 - (d) Inhumaciones asociadas
 - (e) Multimedia asociado
 - (f) Tipos de tumba asociados
 - (g) Conservación

- (h) Estructura
 - (i) Composición
 - (j) Organización y jerarquía
 - (k) Restos humanos
 - (l) Ofrendas animales
 - (m) Neonato
 - (n) Año de campaña
 - (ñ) Notas asociadas
 - (o) Campos pendientes
- El sistema deberá almacenar la información correspondiente a las unidades estratigráficas. En concreto:
 - (a) ID UE
 - (b) Localización asociada
 - (c) Unidad de acción
 - (d) Tipo de unidad
 - (e) Color del estrato
 - (f) Constitución del estrato
 - (g) Descripción
 - (h) Excavada
 - (i) Alzada
 - (j) Técnica de excavación
 - (k) Fiabilidad
 - (l) Observaciones
 - (m) Interpretación
 - (n) Componentes geológicos
 - (ñ) Componentes artificiales
 - (o) Componentes orgánicos
 - (p) Superficies
 - (q) Muestras
 - (r) Fases

- (s) Notas asociadas
- (t) Campos pendientes
- El sistema deberá almacenar la información correspondiente a las cremaciones. En concreto:
 - (a) ID Cremación
 - (b) UE asociada
 - (c) Código propio
 - (d) Presentación
 - (e) Peso
 - (f) Descripción
 - (g) Sexo
 - (h) Edad
 - (i) Calidad de la combustión
 - (j) Análisis posdeposicional
 - (k) Observaciones
 - (l) Tumba asociada
- El sistema deberá almacenar la información correspondiente a las inhumaciones. En concreto:
 - (a) ID Enterramiento
 - (b) UE cadáver
 - (c) UE Fosa
 - (d) UE Estructura
 - (e) UE Relleno
 - (f) Ajuar
 - (g) Fecha
 - (h) Orientación
 - (i) Conservación
 - (j) Conexión anatómica
 - (k) Posición
 - (l) Actitud

- (m) Medidas del esqueleto
 - (n) Sexo
 - (ñ) Edad
 - (o) Descripción
 - (p) Adscripción cultural cronológica
 - (q) Observaciones
 - (r) Tumba asociada
- El sistema deberá almacenar la información correspondiente a los registros. En concreto:
 - (a) Número de control
 - (b) ID Usuario
 - (c) Referencia
 - (d) ID Enterramiento
 - (e) ID Tumba
 - (f) Admin_level
 - (g) Fecha
 - El sistema deberá almacenar la información correspondiente a los objetos multimedia. En concreto:
 - (a) ID Multimedia
 - (b) Título
 - (c) Descripción
 - (d) Tipo
 - (e) Nombre del archivo
 - El sistema deberá almacenar la información correspondiente a las analíticas de fauna. En concreto:
 - (a) ID Analítica
 - (b) Descripción
 - (c) Partes Óseas Especie Edad
 - El sistema deberá almacenar la información correspondiente a las matrices de Harris. En concreto:
 - (a) ID Elemento Harris

- (b) UE
 - (c) Relacionada con UE
 - (d) Posición X
 - (e) Posición Y
 - (f) Posición Z
- El sistema deberá almacenar la información correspondiente a un artículo. En concreto:
 - (a) ID Artículo
 - (b) Título
 - (c) Publicación
 - (d) Número
 - (e) Volumen
 - (f) Páginas
 - (g) ISBN_ISSN
 - El sistema deberá almacenar la información correspondiente a un autor. En concreto:
 - (a) ID Autor
 - (b) Nombre
 - (c) Apellidos
 - (d) Filiación
 - El sistema deberá almacenar la información correspondiente a un mensaje. En concreto:
 - (a) ID Mensaje
 - (b) ID Usuario
 - (c) Admin_level
 - (d) Fecha
 - (e) Comentario
 - (f) Usuario destino
 - (g) Categoría
 - El sistema deberá almacenar la información correspondiente a un usuario. En concreto:

- (a) ID Usuario
- (b) Username
- (c) Contraseña
- (d) Nombre
- (e) Apellidos
- (f) email
- (g) Ciudad
- (h) País
- (i) Hobbies

3.1.2. Requisitos no funcionales

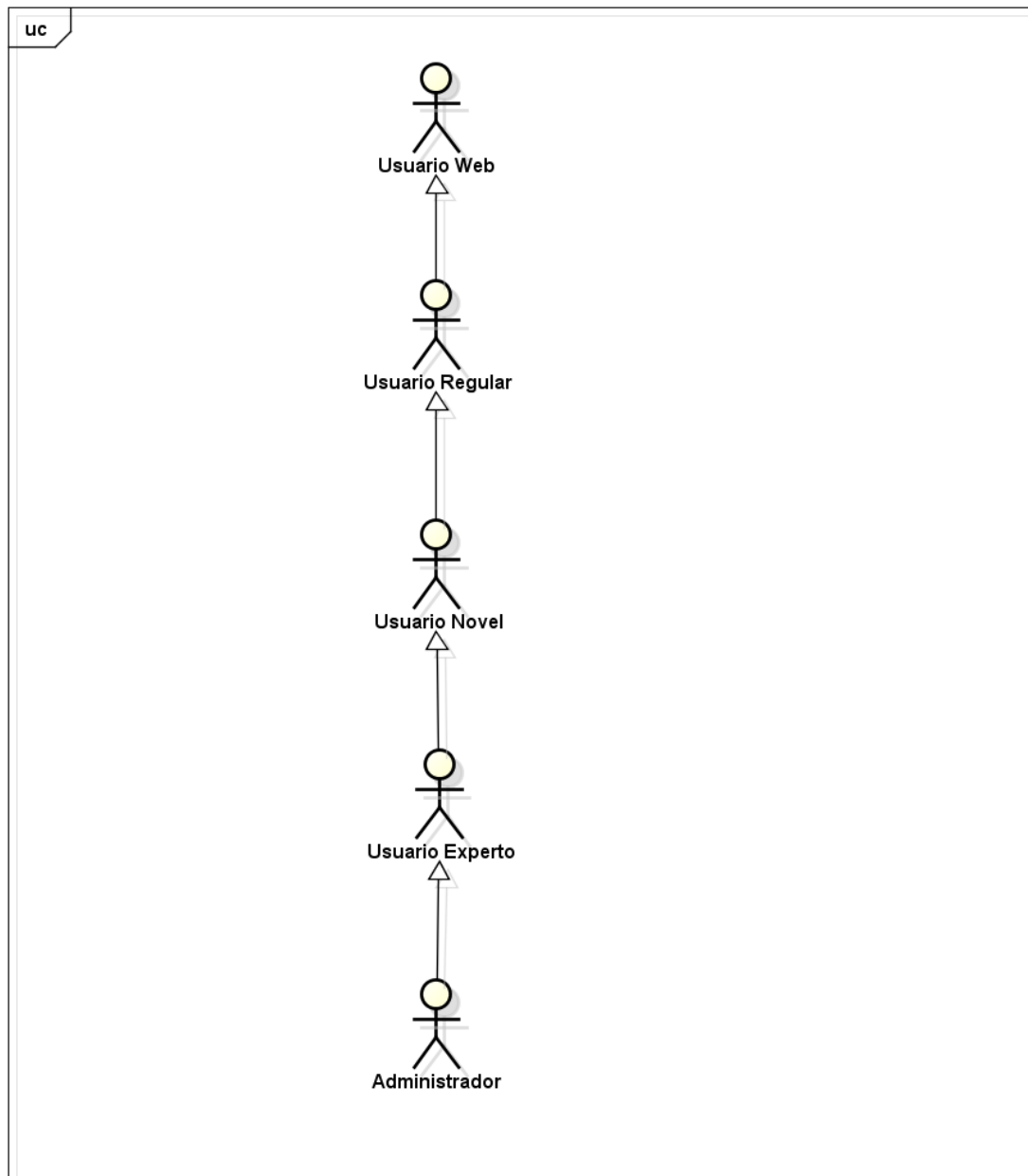
Son requisitos debido a las limitaciones que afectan a los servicios o funciones del sistema. Estos pueden ser más críticos que los requisitos funcionales ya que si no se cumplen, el sistema podría quedar inutilizable. Los requisitos no funcionales ya extraídos son los siguientes:

- El sistema deberá almacenar la información en una base de datos MySQL.
- El sistema estará alojado en un servidor LAMP.
- El sistema deberá responder de forma fiable las 24 horas y 365 días al año.
- El sistema deberá ser accesible mediante cualquier navegador web.

A los que se han añadido:

- El sistema deberá permitir que en menos de 20 minutos, un usuario inexperto pueda realizar búsquedas.
- El sistema deberá potenciar el apartado multimedia y el catálogo de objetos con una interfaz visualmente atractiva.
- El sistema estará implementado con un framework basado en PHP.

3.2. Actores del sistema



powered by Astah

Figura 2: Actores del sistema

La descripción de cada uno de los actores es la siguiente:

Usuario Web : es cualquier usuario que navegue por la red que acceda al sistema pero no posea una cuenta en el mismo.

Usuario Regular : usuario registrado en el sistema con una cuenta de tipo 'regular'. No tiene acceso a la gestión.

Usuario Novel : usuario registrado en el sistema con una cuenta de tipo 'novel'. Tiene acceso a más información del catálogo y a la gestión de algunos objetos.

Usuario Experto : usuario registrado en el sistema con una cuenta de tipo 'experto'. Tiene acceso a más información del catálogo que un novel e incluso puede gestionar objetos creados por un novel, además de poder gestionar más información.

Administrador : como su nombre indica, es el usuario con una cuenta de administrador el cual tiene acceso a toda la información y gestión del sistema. En definitiva cuenta con todos los permisos existentes del mismo

A continuación, se detallan los casos de uso asociados a cada uno de los actores, que detallan las acciones que pueden llevar a cabo en el diagrama de casos de uso.

3.3. Diagrama de Casos de uso

Mediante el diagrama de casos de uso se describe el sistema, su entorno y cómo ambos se relacionan. En el siguiente diagrama se van a detallar los casos de uso ya desarrollados ahora y en anteriores desarrollos del sistema. Al contener el sistema abundantes casos de uso y varios actores implicados, se ha optado por dividirlo en dos diagramas: uno con los casos de uso asociados a un usuario web, un usuario regular y un usuario novel, y otro con los casos de uso asociados a un usuario experto y a un administrador, siempre teniendo en cuenta la relación de jerarquía/especialización mostrada en la figura 3.

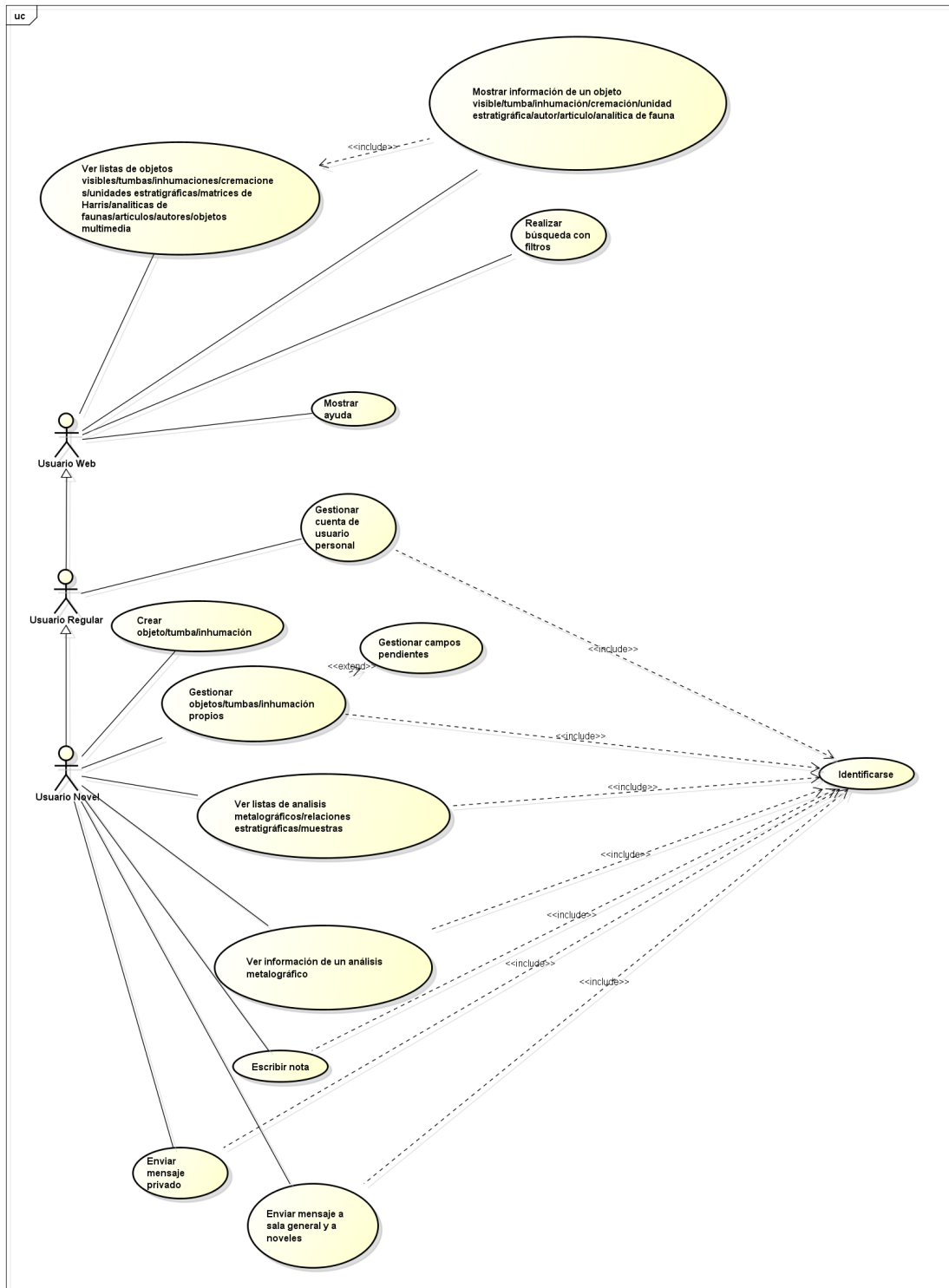
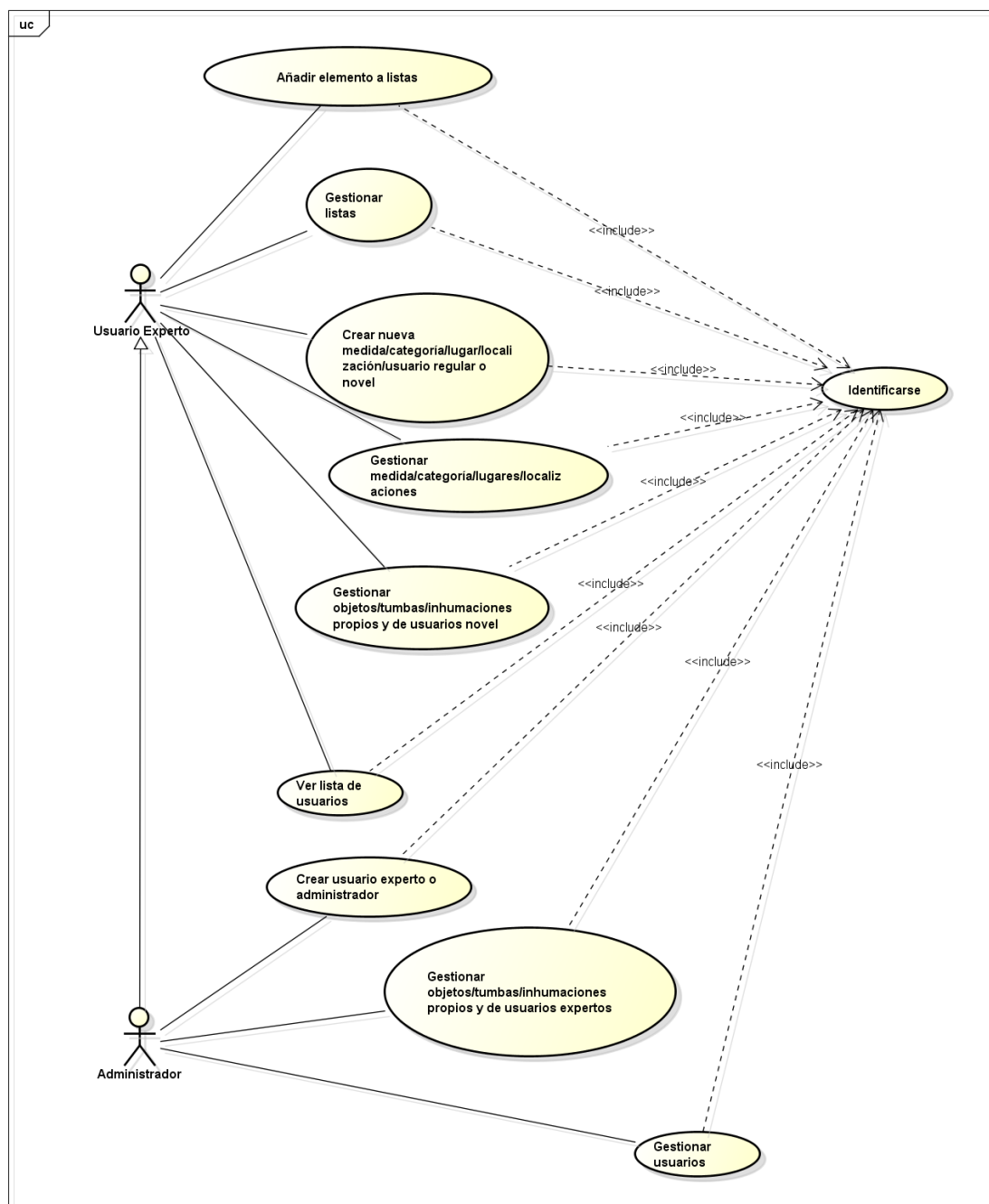


Figura 3: Diagrama de casos de uso de los actores usuario web,regular y novel



powered by Astah

Figura 4: Diagrama de casos de uso de los actores usuario experto y administrador

3.4. Realización de casos de uso

A continuación se describe una descripción de los casos de uso con las interacciones de los diferentes elementos del sistema en cada uno de ellos.

CU-01	Identificarse	
Descripción	El sistema deberá permitir a los actores Usuario Regular, Usuario Novel, Usuario Experto y Administrador poder loguearse para acceder al Sistema tal y como se describe en el siguiente caso de uso.	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	El Sistema muestra los campos de usuario y contraseña.
	2	El actor introduce usuario y contraseña.
	3	El sistema comprueba, para los datos introducidos, que existe ese usuario y la contraseña es válida.
Excepciones	Paso	Acción
	3a	Si el usuario no está registrado, el Sistema muestra el correspondiente mensaje de error y el caso de uso continúa en el paso 1
	3b	Si el usuario está registrado, pero la contraseña es incorrecta, el Sistema muestra el mensaje de error correspondiente y el caso de uso continúa en el paso 1

Cuadro 3.1: Descripción del CU-01.

CU-02	Mostrar ayuda	
Descripción	El sistema deberá mostrar la ayuda necesaria a los actores Usuario Web, Usuario Regular, Usuario Novel, Usuario Experto y Administrador	
Secuencia normal	Paso	Acción
	1	El actor solicita obtener ayuda necesaria para el lugar de la web en el que se encuentra
	2	El sistema muestra los datos de ayuda solicitados

Cuadro 3.2: Descripción del CU-02.

CU-03	Ver listas de objetos del catálogo	
Descripción	El sistema deberá mostrar a los actores Usuario Web, Usuario Regular, Usuario Novel, Usuario Experto y Administrador las listas de un tipo de objeto del catálogo tal y como se describe en el siguiente caso de uso.	
Secuencia normal	Paso	Acción
	1	El actor solicita el listado de objetos para el tipo seleccionado.
	2	El Sistema muestra una lista con los objetos de ese tipo

Cuadro 3.3: Descripción del CU-03.

CU-04	Mostrar información de un objeto del catálogo	
Descripción	El sistema deberá mostrar a los actores Usuario Web, Usuario Regular, Usuario Novel, Usuario Experto y Administrador la información de un objeto del catálogo tal y como se describe en el siguiente caso de uso.	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	Se realiza el caso de uso Ver listas de objetos del catálogo (CU-03)
	2	El actor selecciona un objeto.
	3	El sistema muestra para ese objeto, su información asociada
Excepciones	Paso	Acción
	2a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.

Cuadro 3.4: Descripción del CU-04.

CU-05	Realizar búsqueda con filtros	
Descripción	El sistema deberá permitir a los actores Usuario Web, Usuario Regular, Usuario Novel, Usuario Experto y Administrador la búsqueda con filtros de un tipo de objeto del catálogo	
Secuencia normal	Paso	Acción
	1	Se realizar el caso de uso Ver listas de objetos del catálogo(CU-03).
	2	El actor introduce los datos para filtrar resultados del listado.
	3	El Sistema muestra los objetos que poseen esos datos.
Excepciones	Paso	Acción
	2a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.

Cuadro 3.5: Descripción del CU-05.

CU-06	Enviar email	
Descripción	El sistema deberá permitir a los actores Usuario Web, Usuario Regular, Usuario Novel, Usuario Experto y Administrador el envío de emails	
Secuencia normal	Paso	Acción
	1	El actor solicita enviar un email.
	2	El Sistema solicita el nombre, el correo electrónico y el contenido del mensaje.
	3	El actor introduce los datos solicitados por el sistema.
	4	El Sistema comprueba que los datos son válidos y envía el contenido del email
Excepciones	Paso	Acción
	1a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	3a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.

Cuadro 3.6: Descripción del CU-06.

CU-07	Gestionar cuenta de usuario personal	
Descripción	El sistema deberá permitir a los actores Usuario Regular, Usuario Novel, Usuario Experto y Administrador la gestión de su cuenta de usuario	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	El actor solicita ver su cuenta de usuario
	2	El Sistema muestra un formulario con sus datos personales
	3	El actor modifica sus datos personales.
	4	El Sistema notifica los cambios realizados.
Excepciones	Paso	Acción
	2a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	4a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.

Cuadro 3.7: Descripción del CU-07.

CU-08	Crear objeto/tumba/inhumación	
Descripción	El sistema deberá permitir a los actores Usuario Novel, Usuario Experto y Administrador la creación de un objeto/tumba/inhumación	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	El actor solicita la creación de un nuevo objeto/tumba/inhumación
	2	El Sistema solicita los datos necesarios para crear un objeto/tumba/inhumación
	3	El actor introduce los datos solicitados.
Excepciones	4	El Sistema notifica los cambios realizados y guarda el objeto/tumba/inhumación creado.
	Paso	Acción
	1a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	3a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
Postcondición	Se ha creado un nuevo objeto/tumba/inhumación en el sistema	

Cuadro 3.8: Descripción del CU-08.

CU-09	Gestionar objeto/tumba/inhumación	
Descripción	<p>El sistema deberá permitir a los actores Usuario Novel, Usuario Experto y Administrador la gestión de un objeto/tumba/inhumación:</p> <ul style="list-style-type: none"> • actor Novel: solo permite gestión de objetos/tumbas/inhumaciones propios • actor Experto: permite gestión de objetos/-tumbas/inhumaciones propios y de noveles • actor Administrador: permite gestión de todos los objetos/tumbas/inhumaciones 	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	Se realiza el caso de uso ver listas de objetos del catálogo(CU-03)
	2	El actor selecciona un objeto/tumba/inhumación.
	3	El Sistema muestra un formulario con los datos actuales
	4	El actor modifica los datos del objeto/tumba/inhumación.
Excepciones	5	El Sistema notifica los cambios
	Paso	Acción
	2a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	4a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
Postcondición	Se ha modificado un nuevo objeto/tumba/inhumación en el sistema	

Cuadro 3.9: Descripción del CU-09.

CU-10	Borrar un objeto del catálogo	
Descripción	<p>El sistema deberá permitir a los actores Usuario Novel, Usuario Experto y Administrador borrar objetos del catálogo</p> <ul style="list-style-type: none"> • actor Novel: solo permite el borrado de objetos/tumbas/inhumaciones propios • actor Experto: permite borrado objetos/tumbas/inhumaciones propios y de noveles, y de objetos que pueda gestionar. • actor Administrador: permite el borrado de cualquier objeto del catálogo 	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	Se realiza el caso de uso ver listas de objetos del catálogo(CU-03)
	2	El actor selecciona el objeto del catálogo que desea eliminar.
	3	El Sistema borra el objeto del sistema y lo notifica.
Excepciones	Paso	Acción
	2a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
Postcondición	Se ha borrado el objeto del sistema	

Cuadro 3.10: Descripción del CU-10.

CU-11	Clasificar objeto en parte(s)	
Descripción	El sistema deberá permitir a los actores Usuario Novel, Usuario Experto y Administrador la clasificación de un objeto y sus partes	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	Se realiza el caso de uso ver listas de objetos del catálogo(CU-03)
	2	El actor selecciona gestionar un objeto.
	3	El Sistema muestra la sección de gestión del objeto.
	4	El actor solicita clasifica el objeto y sus partes.
	5	El Sistema muestra las partes asociadas al objeto.
	6	El actor selecciona una parte.
	7	El Sistema muestra las opciones de clasificación.
	8	El actor selecciona una opción.
Excepciones	Paso	Acción
	2a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	4a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	4b	El objeto no tiene partes, a continuación el caso de uso queda sin efecto.
	6a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	8a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
Postcondición	Se ha clasificado el objeto con su parte	

Cuadro 3.11: Descripción del CU-11.

CU-12	Gestionar campos pendientes	
Descripción	El sistema deberá permitir a los actores Usuario Novel, Usuario Experto y Administrador la gestión de los campos pendientes de un determinado objeto del catálogo	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	Se realiza el caso de uso ver listas de objetos del catálogo(CU-03)
	2	El actor selecciona un objeto/tumba/inhumación.
	3	El Sistema muestra un formulario con los datos actuales
	4	El actor solicita ver sus campos pendientes.
	5	El Sistema muestra los campos pendientes.
	6	El actor marca/desmarca un campo de la sección de pendientes.
7	El Sistema guarda los cambios y lo notifica	
Excepciones	Paso	Acción
	2a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	4a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
6a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.	
Postcondición	Se han modificado los campos pendientes	

Cuadro 3.12: Descripción del CU-12.

CU-13	Ver listas de análisis metalográficos/relaciones estratigráficas/muestras	
Descripción	El sistema deberá permitir a los actores Usuario Novel, Usuario Experto y Administrador ver listas de análisis metalográficos/relaciones estratigráficas/muestras	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	El actor solicita el listado del tipo seleccionado
	2	El Sistema muestra los objetos de ese tipo
Excepciones	Paso	Acción
	1a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
Postcondición	Se muestra un listado del tipo seleccionado	

Cuadro 3.13: Descripción del CU-13.

CU-14	Escribir nota	
Descripción	El sistema deberá permitir a los actores Usuario Novel, Usuario Experto y Administrador escribir una nota en un objeto/ue/tumba	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	El actor solicita escribir una nota.
	2	El Sistema muestra un formulario donde escribir la nota.
	3	El actor introduce la nota asociada al objeto.
	4	El Sistema guarda la nota y lo notifica.
Excepciones	Paso	Acción
	1a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	3a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
Postcondición	Se ha guardado la nota	

Cuadro 3.14: Descripción del CU-14.

CU-15	Enviar mensaje privado	
Descripción	El sistema deberá permitir a los actores Usuario Novel, Usuario Experto y Administrador escribir una nota	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	El actor solicita enviar un mensaje privado a un usuario.
	2	El Sistema muestra un formulario donde escribir la nota e indicar el usuario de destino.
	3	El actor introduce el contenido del mensaje y el usuario
	4	El Sistema guarda el mensaje.
Excepciones	Paso	Acción
	1a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	3a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
Postcondición	Se ha enviado el mensaje privado	

Cuadro 3.15: Descripción del CU-15.

CU-16	Enviar mensaje a sala	
Descripción	<p>El sistema deberá permitir a los actores Usuario Novel, Usuario Experto y Administrador enviar un mensaje a una sala</p> <ul style="list-style-type: none"> • actor Novel: solo a sala general y de noveles. • actor Experto: solo a sala de general y de expertos. • actor Administrador: a todas las salas. <p>(noveles: , resto de actores: a todas las salas)</p>	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	El actor solicita enviar un mensaje a una sala
	2	El Sistema muestra un formulario donde escribir la nota e indicar a la sala que va dirigido el mensaje.
	3	El actor introduce el contenido del mensaje y la sala.
	4	El Sistema guarda el mensaje.
Excepciones	Paso	Acción
	1a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	3a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
Postcondición	Se ha enviado el mensaje a la sala	

Cuadro 3.16: Descripción del CU-16.

CU-17	Crear nueva cremación/unidad estratigráfica/- Matriz de Harris/ analítica de faunas/ multimedia/muestra/relación estratigráfica.	
Descripción	El sistema deberá permitir a los actores Usuario Experto y Administrador crear nueva cremación/unidad estratigráfica/Matriz de Harris/ analítica de faunas/multimedia/muestra/relación estratigráfica.	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	El actor solicita crear un nuevo objeto del tipo seleccionado
	2	El Sistema muestra un formulario donde introducir los datos asociados al objeto.
	3	El actor introduce los datos necesarios para la creación del objeto.
	4	El Sistema guarda el objeto en el sistema.
Excepciones	Paso	Acción
	1a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	3a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	3b	Si hay algún error, el sistema lo notifica y continúa en el Paso 3
Postcondición	Se ha creado el objeto del tipo seleccionado	

Cuadro 3.17: Descripción del CU-17.

CU-18	Gestionar cremación/unidad estratigráfica/Matriz de Harris/ analítica de faunas/ multimedia/-muestra/relación estratigráfica.	
Descripción	El sistema deberá permitir a los actores Usuario Experto y Administrador gestionar nueva cremación/unidad estratigráfica/Matriz de Harris/ analítica de faunas/multimedia/muestra/relación estratigráfica.	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	2	El actor solicita crear un nuevo objeto del tipo seleccionado
	3	El Sistema muestra un formulario donde introducir los datos asociados al objeto.
	4	El actor introduce los datos necesarios para la creación del objeto.
	5	El Sistema guarda el objeto en el sistema.
Excepciones	Paso	Acción
	2a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	4a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	4b	Si hay algún error, el sistema lo notifica y continúa en el Paso 3
Postcondición	Se ha modificado el elemento del tipo seleccionado	

Cuadro 3.18: Descripción del CU-18.

CU-19	Añadir elemento a lista palabras clave/c.artificiales/c.geológicos/c.orgánicos/muestras/superficie/tipo de muestra/tipo de tumba/materias primas/fases	
Descripción	El sistema deberá permitir a los actores Usuario Experto y Administrador añadir un elementos a listas del tipo seleccionado.	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	El actor solicita añadir un elemento a listas.
	2	El Sistema muestra los tipos de lista disponibles.
	3	El actor selecciona un tipo de lista
	4	El Sistema muestra un formulario para introducir los datos.
	5	El actor introduce los datos asociados al elemento.
	6	El Sistema guarda los cambios y lo notifica.
Excepciones	Paso	Acción
	1a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	3a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	5a	Si hay algún error, el sistema lo notifica y continúa en el Paso 3
	5b	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
Postcondición	Se ha guardado un elemento en la lista seleccionada	

Cuadro 3.19: Descripción del CU-19.

CU-20	Gestionar elemento de una lista	
Descripción	El sistema deberá permitir a los actores Usuario Experto y Administrador la gestión de un elemento de una lista.	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	El actor solicita gestionar un elemento de una lista.
	2	El Sistema muestra los tipos de lista disponibles.
	3	El actor selecciona un tipo de lista
	4	El Sistema muestra los elementos asociados a la lista.
	5	El actor selecciona un elemento.
	6	El Sistema solicita los datos a gestionar.
	7	El actor introduce los datos solicitados.
	8	El Sistema guarda los cambios y lo notifica.
Excepciones	Paso	Acción
	1a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	3a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	4a	
	7a	Si hay algún error, el sistema lo notifica y continúa en el Paso 3
	7b	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
Postcondición	Se han guardado los cambios en el elemento	

Cuadro 3.20: Descripción del CU-20.

CU-21	Crear usuario	
Descripción	<p>El sistema deberá permitir a los actores Usuario Experto y Administrador crear un usuario</p> <ul style="list-style-type: none"> • actor Experto: puede crear usuarios novel y usuarios regular • actor Administrador: puede crear usuartios de cualquier tipo 	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	El actor solicita crear un nuevo usuario.
	2	El Sistema muestra un formulario con los campos a introducir.
	3	El actor selecciona un tipo de usuario e introduce los datos del mismo.
	4	El Sistema comprueba que no hay errores y guarda el usuario.
Excepciones	Paso	Acción
	1a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	3a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	4a	Si hay algún error, el sistema lo notifica y continúa en el Paso 3
Postcondición	Se ha creado un nuevo usuario	

Cuadro 3.21: Descripción del CU-21.

CU-22	Ver lista de usuarios	
Descripción	El sistema deberá permitir a los actores Usuario Experto y Administrador ver la lista de usuarios	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	El actor solicita ver la lista de usuarios.
	2	El Sistema muestra una lista de usuarios con su nivel, username, nombre y apellidos.
Excepciones	Paso	Acción
	1a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
Postcondición	Se muestra la lista de usuarios	

Cuadro 3.22: Descripción del CU-22.

CU-23	Gestionar usuario	
Descripción	El sistema deberá permitir al actor Administrador gestionar un usuario	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	Se realiza el caso de uso Identificarse(CU-01).
	2	Se realiza el caso de uso Ver lista de usuarios(CU-22)
	3	El actor selecciona un usuario.
	4	El Sistema muestra un formulario con los datos del usuario.
	5	El actor introduce los datos a modificar.
	6	El Sistema comprueba que no hay errores, y lo notifica.
Excepciones	Paso	Acción
	2a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
Postcondición	Se han guardado los cambios correctamente	

Cuadro 3.23: Descripción del CU-23.

CU-24	Validar registro	
Descripción	El sistema deberá permitir al actor Administrador validar un registro pendiente de un elemento	
Precondición	El actor tiene que estar previamente registrado en el Sistema	
Secuencia normal	Paso	Acción
	1	El actor solicita ver la lista de registros.
	2	El Sistema muestra la lista de registros.
	3	El actor selecciona un registro.
	4	El Sistema valida el registro y lo notifica.
Excepciones	Paso	Acción
	1a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
	3a	El actor cancela la operación, a continuación el caso de uso queda sin efecto.
Postcondición	El elemento queda validado	

Cuadro 3.24: Descripción del CU-24.

3.5. Modelo de dominio

El modelo de dominio se utiliza para representar los conceptos clave del dominio del problema. También sirve para identificar las relaciones entre las entidades del problema, cada una de ellas con sus atributos. A continuación se detalla un diagrama con las entidades identificadas en la fase de análisis y extraídas a partir de los casos de Uso y los requisitos:

4 Arquitectura y diseño

Este capítulo se centra en los aspectos del diseño del sistema. En él se encuentran los diagramas de secuencia correspondientes a los casos de uso descritos en el capítulo anterior, el modelo de dominio, el modelo relacional de la base de datos del sistema. Además se abordan los patrones utilizados y un esquema del funcionamiento del framework utilizado con dichos patrones.

4.1. Patrón MVC

El patrón MVC (Modelo-Vista-Controlador) [16] es un patrón de arquitectura software cuyo principal objetivo es desacoplar los objetos del dominio (modelo) de las vistas para mejorar la reusabilidad de los objetos del dominio y minimizar el impacto de los cambios de la interfaz en los objetos del dominio.

4.1.1. Componentes principales

Como su propio nombre indica los tres componentes principales son los Modelos, las Vistas y los Controladores, cada uno con unas responsabilidades bien diferenciadas:

Modelo : Encapsula el núcleo de datos de la aplicación, implementando también los accesos a dichos datos. Envía a la vista la información solicitada para ser mostrada. Las peticiones procedentes de la vista, las recibe el modelo a través del controlador.

Vista : es la parte 'visible' por el usuario, donde se le muestra la información solicitada en un formato adecuado para que el usuario pueda interactuar con el modelo a través de los controladores.

Controlador : Responde a eventos, que principalmente son las entradas del usuario e invoca al 'modelo' cuando se hace alguna solicitud sobre la información contenida en él. Si se pretende modificar los datos del modelo, es el controlador el que implementa el procedimiento de actualización en este caso.

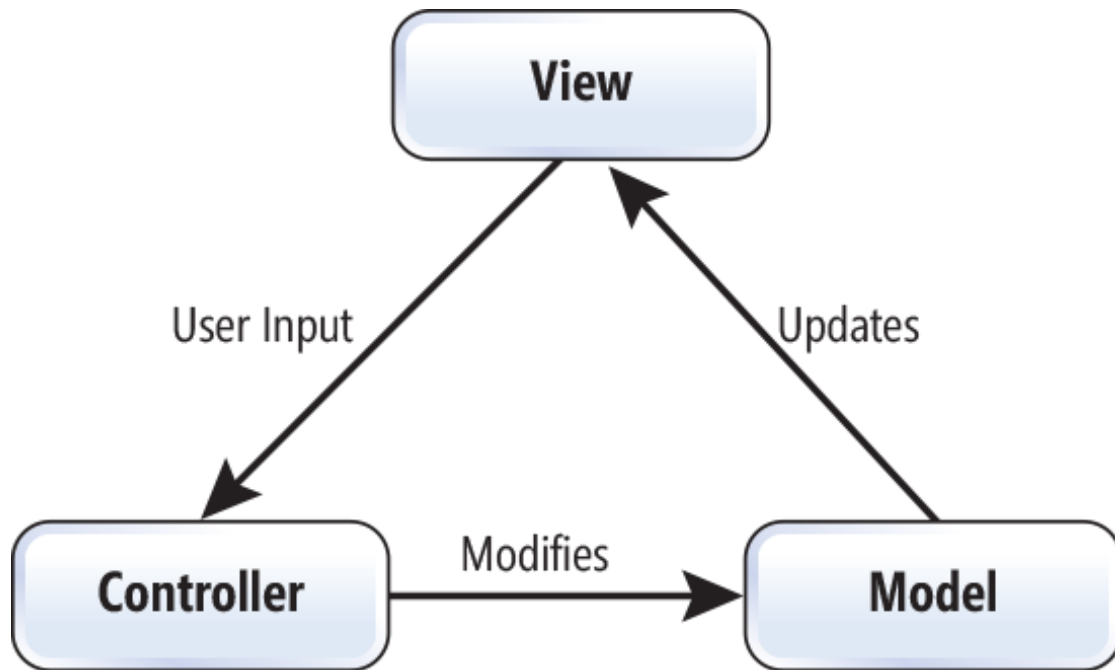


Figura 6: Modelo Vista Controlador

4.1.2. Flujo de control básico del patrón MVC

Aunque hay diferentes escenarios y diferentes implementaciones del patrón MVC, el flujo de control generalmente es el siguiente:

1. El usuario activa algún evento en su interacción con la interfaz de usuario.
2. El controlador recibe el evento del usuario con sus entradas, lo interpreta e invoca a un procedimiento del modelo.
3. El modelo realiza el servicio solicitado. Esto da como resultado un cambio en los datos internos de la aplicación y notifica a las vistas y controladores asociados, mediante el mecanismo de propagación de cambios, el cambio producido invocando sus procedimientos de actualización.
4. La vista solicita los datos cambiados al modelo y se vuelve a mostrar en pantalla.
5. El controlador queda a la espera de más eventos por parte del usuario.

4.2. Active Record

Los patrones de acceso a datos permiten de manera simplificada el acceso a los datos almacenados en un sistema de almacenamiento persistente, en este caso una base de datos relacional. Los modelos en Laravel implementan el patrón Active Record del cuál vamos a explicar su funcionamiento. En el patrón Active Record[17], una clase está

ligada a una tabla de la Base de Datos, y ésta clase implementa los métodos de consulta y modificación de la tabla a la que está ligada. De este modo aislamos a la aplicación de la lógica SQL ya que delegará en la capa ActiveRecord para realizar las operaciones.

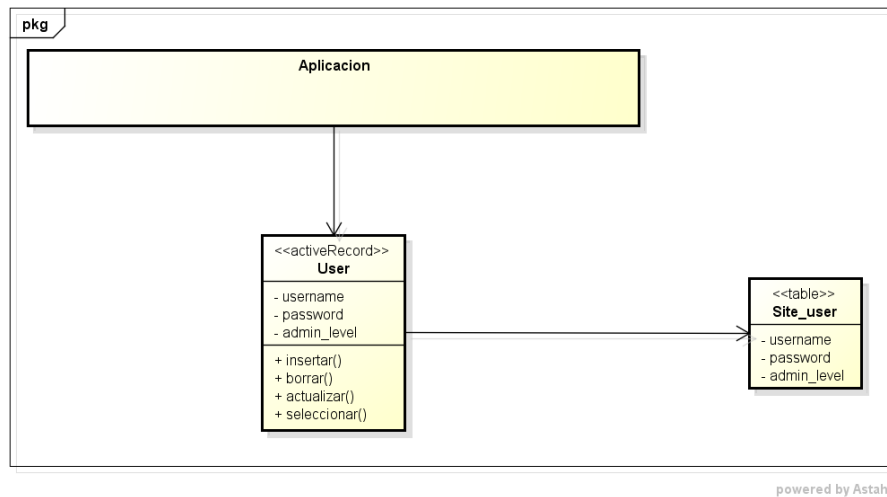


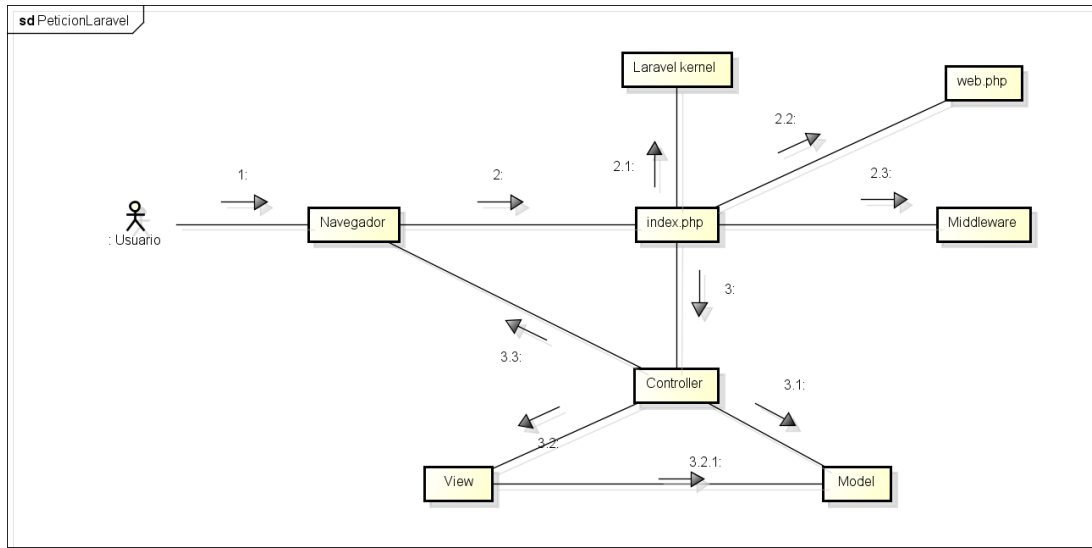
Figura 7: Esquema del patrón Active Record

La simplificación de acceso a estos datos, se consigue a través de un ORM, mapeo objeto relacional, que consiste en la transformación de las tablas de la base de datos en objetos, que simplifiquen este acceso. Este ORM[18], al proporcionar los métodos de acceso y las operaciones básicas de una base de datos, oculta parte de la complejidad con una capa de abstracción al desarrollador.

4.3. Funcionamiento de una petición web en Laravel

El esquema básico cuando Laravel recibe una petición es el siguiente:

1. Todas las peticiones tienen un punto de entrada que es el fichero index.php, donde se comprueba que la URL es válida.
2. Después la petición es enviada al kernel de Laravel que define los servicios y las tareas que se van a ejecutar, como detectar el entorno, manejar errores o comprobar middlewares
3. Después se llama al método del controlador que gestiona esa ruta.
4. El controlador procesará la petición para invocar al modelo y con ello actualizar los datos de la vista.



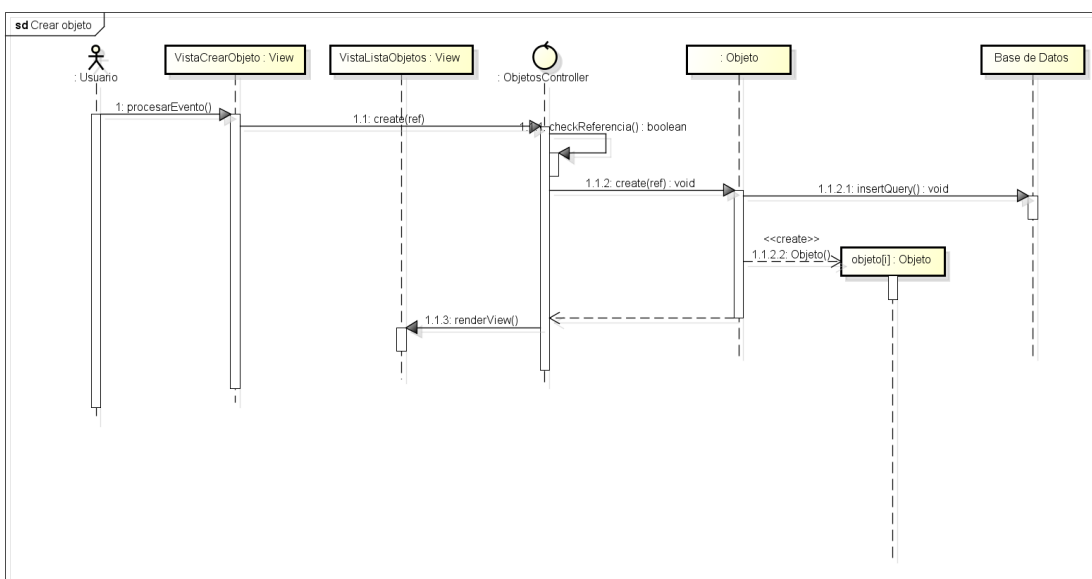
powered by Astah

Figura 8: Diagrama de comunicación de una petición web en Laravel

4.4. Diagramas de secuencia

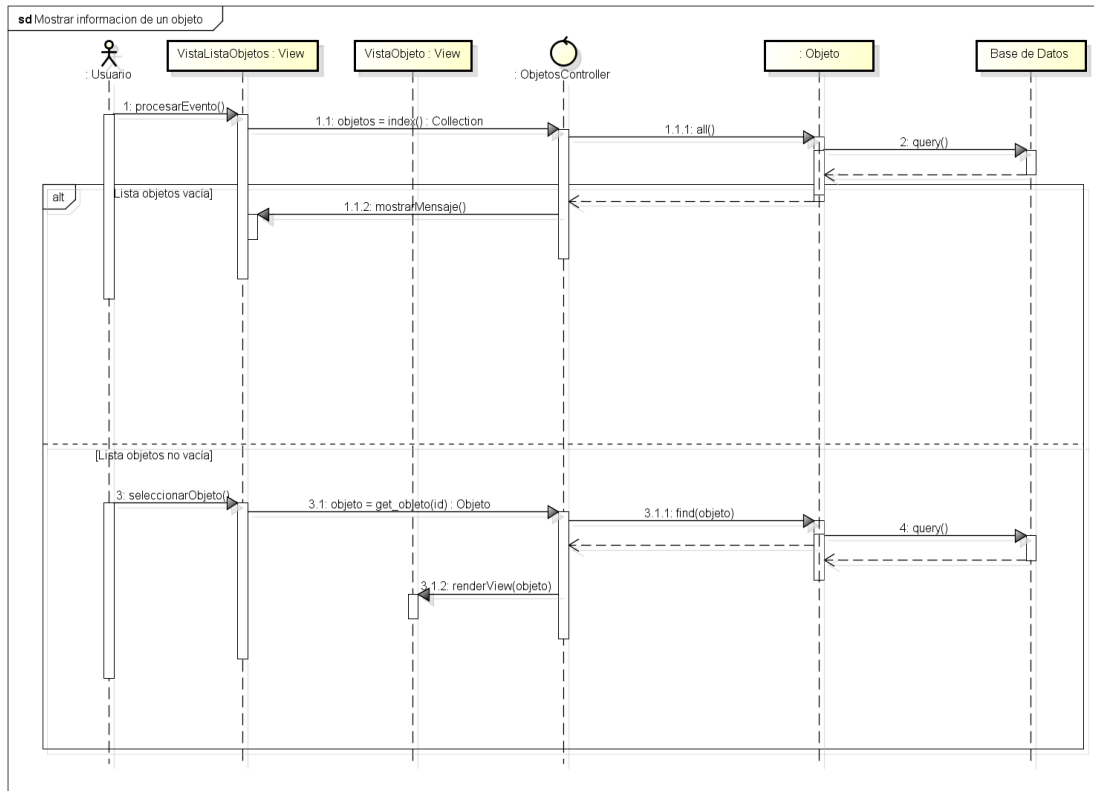
Los diagramas de secuencia muestran la interacción entre los diferentes objetos del sistema a través del tiempo y describen el comportamiento de un caso de uso.

Solo se han realizado los diagramas de secuencia para el caso de las operaciones Create Read Update Delete (CRUD) de las fichas de objeto, además de la búsqueda con filtros, con el fin de no sobrecargar este documento, ya que muchas operaciones son muy parecidas y no aportarían un valor añadido.



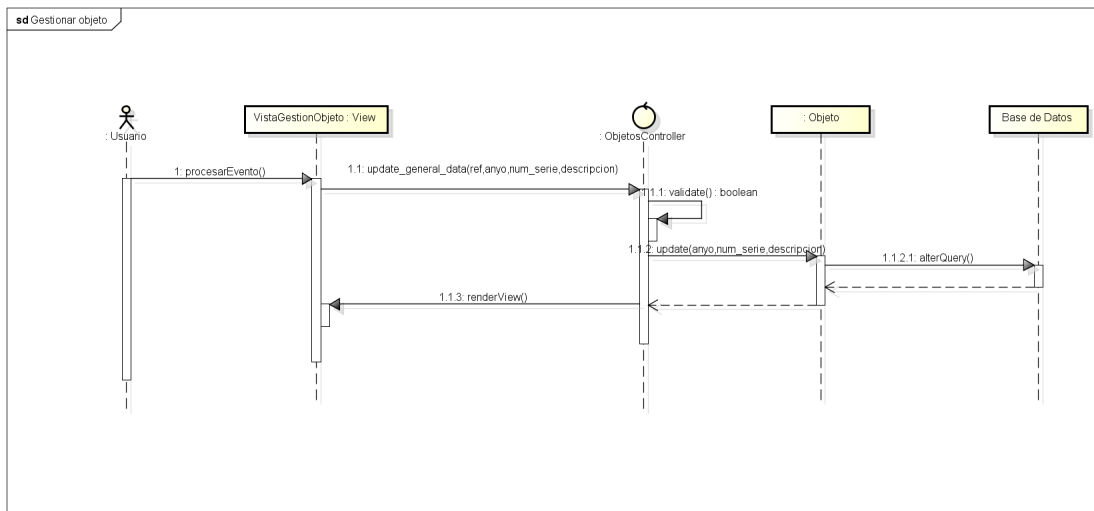
powered by Astah

Figura 9: Diagrama de secuencia del CU-08 Crear objeto



powered by Astah

Figura 10: Diagrama de secuencia del CU-04 Mostrar información de un objeto



powered by Astah

Figura 11: Diagrama de secuencia del CU-09 Gestionar un objeto

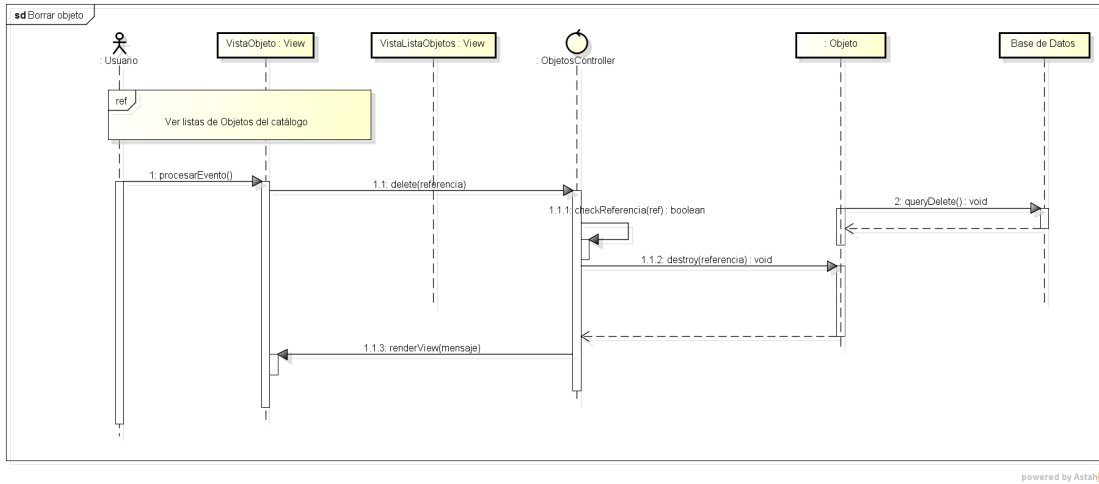


Figura 12: Diagrama de secuencia del CU-10 Borrar un objeto del catálogo

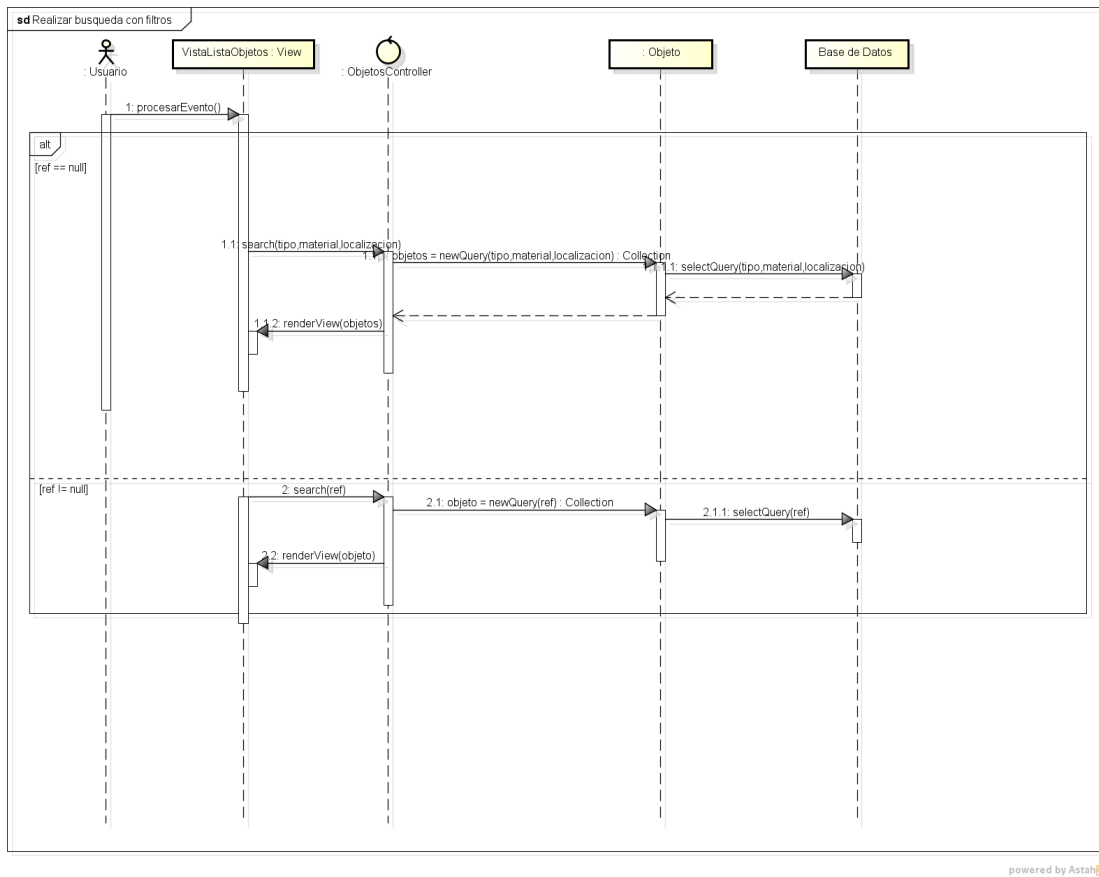


Figura 13: Diagrama de secuencia del CU-05 Realizar búsqueda con filtros

4.5. Diseño de la base de datos

4.5.1. Diagrama entidad relación de la base de datos

En el diagrama entidad relación quedan reflejadas las tablas que se compone la base de datos y las relaciones que hay entre ellas. Cada relación puede tener restricciones asociadas que son condiciones que restringen los datos que pueden almacenarse. Las más destacadas son:

Restricción de clave principal : subconjunto mínimo de campos de una relación que constituye un identificador único en cada tupla.

Restricción de clave foránea : conjunto de campos(clave foránea) en una relación que sirven para referenciar tuplas en otra tabla. El campo referenciado debe ser clave primaria, para asegurar que se hace referencia a una única tupla.

El diagrama entidad relación de la base de datos es el siguiente.

4.5.2. Descripción de las tablas

En el modelo entidad relación, tenemos cuatro tablas principales, que vienen complementadas con más información:

fichaobjeto : describe la información de un Objeto.

- **analisismetalografico** : analisis de la proporción de metales encontrada en el objeto. cada objeto puede tener un análisis asociado.
- **notasobjeto** : notas de texto asociadas a un objeto.
- **parteobjeto**: almacena las partes de las que se compone un objeto.
- **medidasobjeto y medidas** : medidas de cada parte del objeto.
- **camposobjeto y pendientesobjeto** : información sobre los campos que tiene pendientes de editar un objeto.
- **materialobjeto y materiaPrima** : materiales de los que se compone cada parte del objeto.
- **multimediaobjeto** : elementos multimedia asociados al objeto.
- **publicadoen y articulo** : artículos en los que se ha publicado un objeto.
- **autoria y autor**: guarda información sobre el autor o los autores que han publicado el artículo.
- **keywordsarticulo y palabraclave** : palabras clave asociadas a un artículo.
- **categoria y subcategoria** : categoría del objeto y la subcategoría de cada objeto, si la tuviera. Una categoría puede tener muchas subcategorías asociadas.
- **medidascategoria y medidassubcategoria** : medidas asociadas a cada categoría y subcategoría de los objetos.

tumba : describe la información sobre una tumba.

- **inhumacionestumba e inhumacion** : inhumaciones asociadas a una tumba con su información asociada.
- **ofrendasfauna y analiticafaunas** : analíticas de faunas asociadas a una tumba.
- **cremacionestumba y cremaciones** : cremaciones asociadas a una tumba con su información asociada.
- **campostumba y pendiente tumba** : campos pendientes de una tumba para editar.

- **multimediatumba** : multimedias asociados a una tumba. En **almacenmulti-media** se guarda la información asociada a los multimedias asociados a un objeto o tumba.

- **registro** : almacena los objetos,tumbas e inhumaciones pendientes de validar.

unidadestratigrafica : informacion asociada a una unidad estratigráfica.

- **notasue** : notas de texto asociadas a una unidad estratigráfica.
- **cgeologicosue y componentesgeologicos** : componentes geológicos asociados a una unidad estratigráfica.
- **cartificialesue y componentesartificiales** : componentes artificiales asociados a una unidad estratigráfica.
- **corganicosue y componentesorganicos** : componentes orgánicos asociados a una unidad estratigráfica.
- **muestrasue y muestras** : muestras asociadas a una unidad estratigráfica.
- **muestraesdetipo y tiposmuestra** : las muestras a su vez pueden estar asociados a diferentes tipos de muestra.
- **dietasfauna y analitica faunas** : analíticas de fauna asociadas a una unidad estratigráfica.
- **fosilesue y fosiles** : artefactos asociados a una unidad estratigráfica.
- **superficiesue y superficies** : superficies asociadas a una unidad estratigráfica.
- **fasesue y fase** : fases históricas asociadas a una unidad estratigráfica.
- **camposue y pendienteue** : campos pendientes de editar de una unidad estratigráfica.
- **relacionesestratigraficas y matrixharris** : relaciones entre las unidades estratigráficas mediante las matrices de Harris.
- **localización y lugar** : la localización donde se han encontrado un objeto, una tumba o una unidadestratigráfica, que pertenece a un lugar. Un lugar puede tener varias localizaciones asociadas.

usuarios : tablas donde se guarda información relativa a los usuarios.

- **site_user y site_user_info** : información asociada a los usuarios.
- **mensajesusuario** : mensajes enviados por los usuarios, ya sean privados o a las diferentes salas de mensajes.

4.6. Rediseño de las interfaces

Aunque la modificación del aspecto visual de la interfaz gráfica de la aplicación no era el principal cometido en este trabajo, se han modificado algunas vistas para mejorar su aspecto visual y fomentar el uso de la aplicación, siguiendo las directrices de CSS Bootstrap[19] como framework utilizado para el diseño de las interfaces.

4.6.1. Rediseño del catálogo de las fichas de objetos

Para la lista de objetos se ha modificado la presentación en forma de tabla a presentar los objetos en fila y usando los componentes 'card' proporcionado por Bootstrap, que están formado por una imagen de cabecera, en el centro la información que se desea visualizar (descripción y localización) y en el pie, los enlaces que dan acceso a la información de la ficha del objeto o al menú de gestión.

Además si no tiene ningún multimedia asociada se ha añadido una foto por defecto en blanco y negro.

Otro añadido importante es la inclusión de los materiales por los que está formado el objeto, en cada una de sus partes, utilizando el mismo código de colores de PintiaDatApp que es el siguiente:

Madera : marrón

Hierro : azul

Bronce : verde

Cerámica : naranja

Vidrio : azul celeste

Hueso : Blanco opaco

Otros : Gris

Finalmente, en los botones se ha eliminado el texto y se han mantenido los iconos y los mismos colores que se utilizan en la aplicación para que sea claro cuál es el cometido de cada uno de ellos.



Figura 15: Listado antiguo de las fichas de objetos

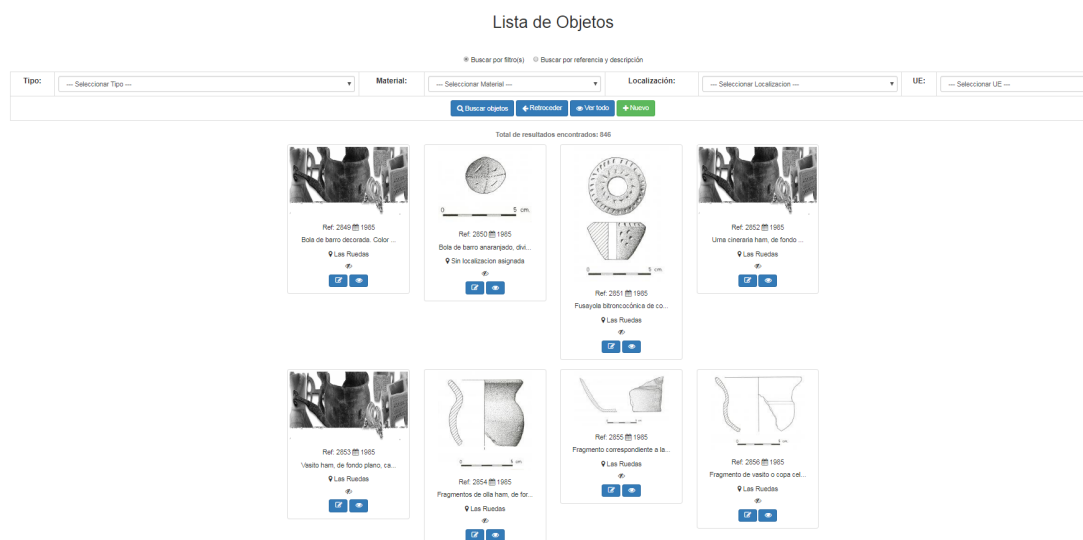


Figura 16: Listado nuevo de las fichas de objetos

4.6.2. Rediseño de la lista de elementos multimedia

Para la nueva interfaz gráfica de elementos multimedia se ha reutilizado el diseño de la lista nueva de objetos y homogeneizar los contenidos visuales de la aplicación

Se han utilizado los mismos botones que en la nueva vista de objetos. Así en cada ítem se pueden ver los siguiente datos: el nombre del elemento multimedia, su tipo (fotografía, planimetría, dibujo o documento). Si se pincha sobre la imagen del elemento se enlaza al archivo asociado al multimedia. Además si se trata de un documento, se da al usuario la posibilidad de descargarlo en su equipo.



Figura 17: Listado antiguo de los elementos multimedia



Figura 18: Listado nuevo de los elementos multimedia

5 Implementación

En este capítulo se enumeran y se describen las principales herramientas utilizadas durante su desarrollo, así como los errores corregidos y mejoras llevadas a cabo.

5.1. Corrección de errores

Durante la fase de implementación y dado que la principal fuente de pruebas y de consulta era el código fuente de la antigua versión del portal, se detectaron los siguientes errores.

1. **Medidas en las partes** : en la anterior versión, en un fallo con la sintaxis SQL de una consulta impedía el añadido de medidas a las partes de un objeto. Una vez corregido ese error se han implementado los métodos necesarios para hacerlo.
2. **Añadir restricciones de clave foránea**: se han añadido restricciones de clave foránea a las tablas de inhumación, fichaobjeto y tumba en la columna `user_id` que referencia a la tabla `site_user`. Por ello cuando se elimina un usuario, las inhumaciones, objetos, tumbas que hubiera creado, la columna `user_id` pasa a ser nula.
3. **Elementos duplicados en menús de selección**

A la hora de seleccionar un elemento para asociarlo con otro cuando se asociaba, aparecía en ambos menús, tanto en elementos a eliminar asociación como para añadir asociación tal y como queda reflejado en la figura 19



Figura 19: Elementos duplicados en los menús de selección

Se ha corregido este error, ya que claramente puede dar lugar a la confusión del usuario, al no saber si el elemento está asociado o no. Ahora al seleccionar un elemento para asociarlo, solo aparece en la lista de elementos a eliminar asociación.

4. **Fallos de seguridad/permisos** : Se encontraron errores relativos a los permisos, un usuario novel podía acceder a la sala de mensajes de expertos y en los objetos podía acceder a objetos que no eran suyos. Estos fallos se han corregido protegiendo las distintas URLs con implementando un middleware de seguridad para que solo accedan los usuarios autorizados a los distintos contenidos.

5.2. Mejoras realizadas

1. **Rediseño de la barra de navegación** : Se ha suprimido la sección Nuevo/modificar, ya que tenía enlaces repetidos tanto en la sección consultar como en la sección eliminada.
2. **Relación estratigráfica inversa** : A la hora de introducir una nueva relación estratigráfica se guarda en 'ambos sentidos' en la base de datos. Por ejemplo: supongamos que hay dos unidades estratigráficas A y B, y queremos crear la relación A 'cubre a' B. Además de guardar esta relación, también se guardará la inversa, B 'cubierta por' A. Si se elimina dicha relación también se eliminará la inversa.
3. **Nuevos filtros de búsqueda** : se han añadido los siguientes filtros de búsqueda:
 - **Objetos** : se ha añadido que se puedan buscar por UE asociada y su descripción

- **Tumba** : se ha añadido que se pueda buscar por UE asociada
 - **Unidad Estratigráfica** : ahora se puede buscar por la fase a la que puede pertenecer. Para esta búsqueda se han añadido las tablas *FasesUE* y *Fases* a la Base de Datos
 - **Multimedia** : se pueden realizar búsquedas por el título del elemento multimedia
4. **Filtros de búsqueda implementados en el lado del cliente** : se ha implementado un script que recogiendo la entrada del usuario muestra inmediatamente los resultados que contienen la cadena introducida por el usuario, sin necesidad de una consulta contra la base de datos, lo cual mejora enormemente la eficiencia del sistema.
 5. **Peticiones AJAX** : mencionamos esta tecnología en el apartado 2.7.3, que mejoraba la eficiencia y la usabilidad del sistema. Se ha introducido en varios puntos del sistema que detallaremos a continuación.
 - a) **Zona de mensajes** : Si se desea cambiar de una sala a otra, al ser una petición sincrónica el cambio es más rápido. Dependiendo del tipo de sala se realizan las operaciones necesarias en el cliente para que el aspecto de los mensajes esté de acuerdo con las especificaciones de cada sala.
 - b) **Medidas** : ya no es necesario pulsar el botón para seleccionar una medida, con seleccionar una opción recupera los datos de la medida y se hacen los cambios oportunos, mejorando la gestión de las medidas.
 - c) **Lugares** : en el menú para editar lugares, ahora se puede editar el lugar directamente seleccionándolo en un cuadro desplegable de opciones, sin necesidad de pulsar un botón como estaba anteriormente.
 6. **Más ayuda y mensajes al usuario**: en la anterior versión del portal, aunque había ayuda abundante, había secciones donde la ayuda era insuficiente o inexistente.

A su vez, se han añadido cuadros de texto de error o de éxito cada vez que se hace una operación sobre la base de datos. En el caso de las inserciones, si ha habido algún problema con los datos introducidos se notifica el problema con un cuadro rojo, y si ha habido éxito con un cuadro verde.

En el caso de los borrados o las actualizaciones de datos, es el mismo caso. Un cuadro verde, si la operación se ha llevado a cabo correctamente y un cuadro rojo cuando ha habido algún problema, con la correspondiente notificación.

7. **Mejoras comprobaciones de las entradas del usuario** : con el uso de la clase Validator de Laravel se han añadido nuevas comprobaciones para que no se introduzcan datos erróneos en la base de datos. En la anterior versión del sistema

las comprobaciones iban dirigidas únicamente a comprobar si un campo no estaba vacío o comprobaciones de tipo. Son las siguientes:

- a) **Multimedias** : se comprueba que sea un fichero del formato adecuado dependiendo del tipo de multimedia.
 - b) **Comprobaciones de clave foránea** : cuando introducimos un objeto que necesita de una clave foránea de otra tabla, comprobamos que el valor de dicha clave exista y haya coherencia con la Base de Datos.
 - c) **Comprobaciones de clave primaria** : con esta comprobación evitamos que haya claves primarias duplicadas en una determinada tabla.
 - d) **Comprobaciones para campos requeridos y tipo de datos** : con estas comprobaciones obligamos a que los campos requeridos para un determinado objeto sean rellenados. Asimismo se comprueba que sean del tipo que hemos definido en el esquema de la base de datos.
 - e) **Comprobaciones tipos de datos enumerados** : exigimos que el valor introducido para este tipo de datos se encuentre dentro de la lista de valores que puede tomar.
8. **Retroceder en las búsquedas** : debido al volumen de elementos en el sistema, se ha habilitado la opción de poder retroceder en el historial de búsquedas realizadas.

5.3. Estructura de un proyecto Laravel

Las aplicaciones en Laravel se dividen en varios directorios, donde en cada uno de ellos se almacenan distintos ficheros con diferentes propósitos. A continuación vamos a mencionar alguno de los más importantes:

Directorio App: en el se encuentran los controladores y los modelos de la aplicación, en los subdirectorios Controllers y Models respectivamente. Además, también se encuentran las clases que forman los middleware de seguridad del proyecto.

Directorio Config: directorio en el que se guardan los ficheros php con la configuración general del proyecto, como los sistemas de ficheros a gestionar, servidor de correo, el tipo de conexión a la base de datos...

Directorio Public: los ficheros css, scripts AJAX y JavaScript implementados y las imágenes que necesita la web.

Directorio Resources: se encuentran las vistas de la aplicación con la extensión .blade.php al usar Blade, como motor de plantillas.

Directorio Routes: se encuentra uno de los ficheros más importantes, donde se registran las rutas de la aplicación para acceder a los recursos web. La estructura de una ruta es la siguiente:

```
Route::peticionHTTP(uri, callback)
```

peticionHTTP: petición HTTP al servidor.

uri: uri que representa el recurso que se solicita al servidor

callback: invocación al método que va a responder a la petición. Puede ser un método de un controlador, una función ...

Fichero .env: fichero que define la configuración del entorno de desarrollo. Los valores que tomen pueden ser distintos cuando la aplicación está aún en desarrollo o ya en un entorno de producción.

5.4. Control de versiones

Se denomina control de versiones a los cambios realizados sobre algún producto o la configuración del mismo. La versión del producto es el estado en el que se encuentra el mismo en un momento dado del desarrollo.

Los sistemas de control de versiones son herramientas que facilitan la gestión de estos cambios. Estos sistemas deben proporcionar:

- Un mecanismo de almacenamiento de los elementos a gestionar
- Posibilidad de realizar cambios sobre los elementos almacenados
- Un registro histórico de las acciones llevadas a cabo sobre cada elemento o conjunto de ellos

Aunque cada sistema de control de versiones pueda tener sus particularidades hay algunos términos comunes a la gran mayoría de ellos:

Repositorio : es el lugar donde se almacenan los elementos a gestionar y los históricos de cambios

Módulo : conjunto de archivos dentro del repositorio que pertenecen al mismo proyecto

Revisión : es una versión determinada de la información gestionada

Línea base : es una revisión aprobada, a partir del cual se realizan los siguientes cambios

Rama : secuencia de versiones de un módulo en una línea temporal.

Publicar : se dice que se publica cuando una copia de los cambios hecho a una copia local es integrada en el repositorio

5.4.1. Git

Para gestionar los cambios en el código fuente a lo largo del desarrollo del proyecto, se ha utilizado Git[21] como programa de control de versiones y GitHub[22] como plataforma donde se aloja el repositorio remoto del proyecto.

Las principales ventajas de este software son su velocidad, un diseño sencillo y su capacidad para controlar proyectos de gran envergadura. Además tiene un fuerte apoyo al desarrollo no lineal, por lo que la gestión de ramas y el acceso al historial de desarrollo es sencillo.

Por último es completamente distribuido por lo que el desarrollador almacena en su equipo una copia local del proyecto y los cambios se propagan al repositorio remoto. Además en el caso de un fallo catastrófico en la plataforma remota, siempre se tiene una copia local para poder restaurarla más adelante.

Algunos de los comandos más importantes son:

git init : crea un repositorio para ser gestionado por git

git clone : Crea una copia local del repositorio

git status : muestra el estado actual de la rama, principalmente los cambios que se han realizado.

git add <archivo> : añade <archivo> a la lista de elementos para seguir sus cambios

git pull : trae los cambios más nuevos del repositorio remoto al repositorio local

git checkout -b <rama> : crea una nueva rama para trabajar sobre ella

git commit -m <mensaje> : confirma los cambios realizados, añadiendo <mensaje> opcionalmente que describe los cambios.

git push origin <rama> : sube los cambios registrados en el commit de la rama local a la rama del repositorio remoto <rama>

git push origin <rama> : sube los cambios registrados en el commit de la rama local a la rama del repositorio remoto <rama>

git revert <commit> : Revierte los cambios introducidos por el <commit>, indicando el hash por el que viene identificado.

6 Pruebas

El siguiente capítulo se centra en describir las pruebas de caja negra realizadas sobre el sistema para detectar errores, tanto nuevos que hayan podido aparecer como los que se encontraban presentes en la anterior versión y verificar que el sistema funciona correctamente

6.1. Introducción

Las pruebas son el conjunto de actividades del desarrollo de software cuyo objetivo es presentar información sobre la calidad del producto a las personas responsables del mismo. Tienen como misión: encontrar 'bugs' en la aplicación, aumentar la confianza en la calidad y evitar la aparición de defectos. Existen dos tipos de pruebas bien diferenciadas:

Pruebas estáticas : son las que se realizan sin la necesidad de ejecutar el código de la aplicación

Pruebas dinámicas : requieren para su ejecución, el código de la aplicación. Dentro de éstas se aplican distintas técnicas de caja negra y caja blanca para medir con mayor precisión el comportamiento de la aplicación desarrollada.

Pruebas de caja negra : son pruebas centradas en las especificaciones del programa. Proporcionando unas entradas, se estudian las salidas resultantes para ver si concuerdan con las salidas esperadas.

Pruebas de caja blanca : son pruebas ligadas al código fuente de la aplicación. Se prueban distintos valores de entrada para comprobar que el flujo de ejecución del programa y las salidas son las esperadas.

Para este trabajo se han realizado pruebas de caja negra basadas en los casos de uso extraídos durante la fase de análisis. Se han empleado datos de entradas correctos e incorrectos para comprobar si las salidas son las esperadas.

Se ha aprovechado la relación de especialización, explicada en el capítulo 2, que hay entre los actores del sistema para elaborar una lista de pruebas en función de cada tipo de usuario.

6.2. Pruebas de caja negra

Pruebas asociadas a un usuario web I			
Prueba	Entrada	Salida	Resultado
Ver objetos	Pulsar botón objetos	Lista de objetos	Correcto
Ver tumbas	Pulsar botón tumbas	Lista de tumbas	Correcto
Ver inhumaciones	Pulsar botón inhumaciones	Lista de inhumaciones	Correcto
Ver cremaciones	Pulsar botón cremaciones	Lista de cremaciones	Correcto
Ver unidades estratigráficas	Pulsar botón unidades estratigráficas	Lista de unidades estratigráficas	Correcto
Ver matrices de Harris	Pulsar botón matrices de Harris	Pulsar botón Matrices de Harris	Correcto
Ver analíticas de faunas	Pulsar botón analíticas de faunas	Lista de analíticas de faunas	Correcto
Ver artículos	Pulsar botón artículos	Lista de artículos	Correcto
Ver autores	Pulsar botón autores	Lista de autores	Correcto
Ver multimedias	Pulsar el botón multimedias	Lista de fichas de multimedias	Correcto
Ver información de objeto	Pulsar botón ver	Información del objeto	Correcto
Ver información de tumba	Pulsar botón ver	Información de la tumba	Correcto
Ver información de inhumación	Pulsar el botón ver	Información asociada a la inhumacion	Correcto
Ver información de cremación	Pulsar el botón ver	Información asociada a la cremacion	Correcto
Ver información de unidad estratigráfica	Pulsar el botón ver	Información asociada a la UE	Correcto
Ver información de artículo	Pulsar el botón ver	Información asociada al artículo	Correcto
Ver información de autor	Pulsar el botón ver	Información asociada al autor	Correcto

Cuadro 6.1: Pruebas de caja negra asociadas a un usuario web I.

Pruebas asociadas a un usuario web II			
Prueba	Entrada	Salida	Resultado
Busqueda mediante filtros I	Filtros aplicados por el usuario	Mensaje donde se indica que no hay ningún resultado	Correcto
Busqueda mediante filtros II	Filtros aplicados por el usuario	Un resultado	Correcto
Busqueda mediante filtros III	Filtros aplicado por el usuario	Más de un resultado	Correcto
Consultar ayuda	Pulsar botón ayuda	Mostrar la ayuda necesaria	Correcto
Enviar correo	Pulsar botón enviar	Correo enviado con éxito	Correcto

Cuadro 6.2: Pruebas de caja negra asociadas a un usuario web II.

Pruebas asociadas a un usuario regular			
Prueba	Entrada	Salida	Resultado
Iniciar sesión I	Datos de usuarios correctos	Página inicial de acuerdo con el rol del usuario	Correcto
Iniciar Sesión II	Datos de usuario incorrectos	Página inicial con un mensaje de error	Correcto
Cerrar sesión	Pulsar botón salir	Página inicial sin iniciar sesión	Correcto
Modificar datos cuenta usuario	Datos actualizados de la cuenta de usuario	Se informa de que los cambios se han guardado con éxito	Correcto

Cuadro 6.3: Pruebas de caja negra asociadas a un usuario regular.

Pruebas asociadas a un usuario arqueólogo novel I			
Prueba	Entrada	Salida	Resultado
Crear objeto I	Referencia válida	Objeto creado correctamente	Correcto
Crear objeto II	Referencia no válida	Mensaje de error	Correcto
Crear tumba I	Identificador válido	Tumba creada con éxito	Correcto
Crear tumba II	Identificador no válido	Mensaje de error	Correcto
Crear inhumación I	Datos que incluyen la descripción y la fecha	Inhumación creada con éxito	Correcto
Crear inhumación II	Datos que no incluyen la descripción ni la fecha	Mensaje de error	Correcto
Editar datos de inhumación I	Datos actualizados que incluyen la descripción y la fecha	Datos actualizados con éxito	Correcto
Editar datos de inhumación II	Datos que no incluyen la descripción ni la fecha	Mensaje de error	Correcto
Editar datos de objeto	Datos actualizados del objeto	Datos del objeto actualizados con éxito	Correcto
Crear nota	Datos de la nota	Nota creada con éxito	Correcto
Crear análisis metalográfico	Datos válidos para el análisis metalográfico	Análisis metalográfico creado con éxito	Correcto
Crear análisis metalográfico II	Datos válidos para el análisis metalográfico	Análisis metalográfico creado con éxito	Correcto
Editar análisis metalográfico	Datos actualizados del análisis metalográfico	Análisis metalográfico actualizado con éxito	Correcto

Cuadro 6.4: Pruebas de caja negra asociadas a un usuario arqueólogo novel I.

Pruebas asociadas a un usuario arqueólogo novel II			
Prueba	Entrada	Salida	Resultado
Editar análisis meta-lográfico II	Datos inválidos o falta el nombre del análisis	Mensaje de error	Correcto
Añadir/eliminar asociación de material a objeto	Materiales disponibles	Se muestra la lista actualizada de materiales asociados	Correcto
Añadir/eliminar asociación de unidad estratigráfica a objeto	Lista de unidades estratigráficas para asociar	Objeto asociado a unidad estratigráfica con éxito	Correcto
Añadir/eliminar asociación de tumba a objeto	Lista de tumbas disponibles para asociar	Objeto asociado a tumba con éxito	Correcto
Añadir/eliminar asociación de multimedia a objeto	Lista de multimedias disponibles para asociar o eliminar asociación	Multimedia asociado o asociación eliminada con éxito	Correcto
Añadir/eliminar asociación de artículo a objeto	Lista de artículos disponibles para asociar o eliminar asociación	Artículo asociado o eliminado de asociados al objeto con éxito	Correcto
Añadir/eliminar asociación de localización a objeto	Lista de localizaciones disponibles	Localización asociada al objeto con éxito	Correcto
Añadir/eliminar asociación de campo pendiente a objeto	Lista de campos pendientes o eliminar de pendientes	Campo añadido o eliminado de la lista de pendientes con éxito	Correcto
Crear parte de un objeto	Nombre de la parte válido	Parte creada con éxito	Correcto
Crear parte de un objeto II	Nombre de la parte no válido o falta	Mensaje de error	Correcto
Eliminar parte de un objeto	Pulsar botón eliminar	Parte del objeto eliminada con éxito	Correcto

Cuadro 6.5: Pruebas de caja negra asociadas a un usuario arqueólogo novel II.

Pruebas asociadas a un usuario arqueólogo novel III			
Prueba	Entrada	Salida	Resultado
Añadir/eliminar medida a la parte de un objeto	Lista de medidas disponibles o asociadas	Medida asociada o eliminada de asociadas correctamente	Correcto
Añadir/eliminar medida a la parte de un objeto II	Datos inválidos o falta algún dato	Mensaje de error	Correcto
Eliminar objeto	Pulsar botón eliminar	Objeto eliminado con éxito	Correcto
Editar datos de tumba	Datos actualizados de la tumba	Tumba actualizada con éxito	Correcto
Añadir/eliminar asociación de tipo de tumba a tumba	Lista de tipos de tumba asociados y sin asociar	Tipo de tumba asociado o asociación eliminada con éxito	Correcto
Añadir/eliminar asociación de cremación a tumba	Lista de cremaciones asociadas y no asociadas	Cremación asociada o asociación eliminada con éxito	Correcto
Añadir/eliminar asociación de inhumación a tumba	Lista de inhumaciones asociadas y sin asociar	Inhumación asociada o asociación eliminada con éxito	Correcto
Añadir/eliminar asociación de unidad estratigráfica a tumba	Lista de unidades estratigráficas	Unidad estratigráfica asociada con éxito	Correcto
Añadir/eliminar asociación de ofrenda de fauna a tumba	Lista de ofrendas de fauna asociadas y no asociadas	Ofrenda de fauna asociada o asociación eliminada con éxito	Correcto
Añadir/eliminar asociación de localización a tumba	Lista de localización o localización asociada	Localización asociada o asociación con localización eliminada con éxito	Correcto

Cuadro 6.6: Pruebas de caja negra asociadas a un usuario arqueólogo novel III.

Pruebas asociadas a un usuario arqueólogo novel III			
Prueba	Entrada	Salida	Resultado
Añadir/eliminar asociación de multimedia a tumba	Lista de multimedias asociados y sin asociar	Multimedia asociado o asociación eliminada con éxito	Correcto
Añadir/eliminar asociación de campos pendientes a tumba	Lista de campos pendientes y completados	Campo añadido a completados o a pendientes con éxito	Correcto
Eliminar tumba	Pulsar botón eliminar	Tumba eliminada con éxito	Correcto
Enviar mensaje privado	Contenido del mensaje y usuario de destino seleccionado	Mensaje enviado con éxito	Correcto
Ver mensajes sala de noveles	Pulsar botón sala de noveles	Muestra lista de mensaje publicados en sala de noveles	Correcto
Enviar mensaje a sala de noveles	Contenido del mensaje y pulsar botón Enviar	Mensaje publicado con éxito en sala de Noveles	Correcto
Borrar mensaje	Pulsar botón eliminar	Mensaje eliminado con éxito	

Cuadro 6.7: Pruebas de caja negra asociadas a un usuario arqueólogo novel III.

Pruebas asociadas a un usuario arqueólogo experto I			
Prueba	Entrada	Salida	Resultado
Crear cremación	Datos válidos	Cremación creada correctamente	Correcto
Crear cremación II	Datos sin el código propio	Mensaje de error	Correcto
Crear unidad estratigráfica	Identificador de UE válido	UE creada con éxito	Correcto
Crear unidad estratigráfica II	Identificador de UE no válido	Mensaje de error	Correcto
Crear matriz de Harris	Datos correctos	Matriz de Harris creada con éxito	Correcto
Crear matriz de Harris II	Falta algún dato o datos incorrectos	Mensaje de error	Correcto
Crear analítica de faunas	Datos correctos	Analítica de fauna creada con éxito	Correcto
Crear analítica de faunas II	Datos sin descripción ni partes óseas	Mensaje de error	Correcto

Cuadro 6.8: Pruebas de caja negra asociadas a un usuario arqueólogo experto I.

Pruebas asociadas a un usuario arqueólogo experto II			
Prueba	Entrada	Salida	Resultado
Crear artículo	Datos correctos	Artículo creado con éxito	Correcto
Crear artículo II	Datos incorrectos o faltan datos	Mensaje de error	Correcto
Crear multimedia	Datos válidos	Multimedia creado con éxito	Correcto
Crear multimedia II	Datos incorrectos	Mensaje de error	Correcto
Crear muestra	Datos válidos	Muestra creada con éxito	Correcto
Crear muestra II	Datos incorrectos o número de registro inválido	Mensaje de error	Correcto
Crear relación estratigráfica	Datos válidos	Relación estratigráfica creada con éxito	Correcto
Crear relación estratigráfica II	Datos inválidos o vacíos	Mensaje de error	Correcto
Crear elemento de una lista	Denominación válida	Elemento creado correctamente	Correcto
Crear elemento en una lista II.	Denominación inválida o vacía	Mensaje de error	Correcto
Editar elemento de una lista	Denominación actualizada	Elemento actualizado con éxito	Correcto
Eliminar elemento de una lista	Pulsar botón eliminar	Elemento eliminado con éxito	Correcto
Crear medida	Datos válidos	Medida creada con éxito	Correcto
Crear medida II	Datos inválidos o falta algún dato	Mensaje de error	Correcto
Crear categoría	Nombre válido de categoría	Categoría creada con éxito	Correcto
Crear categoría II	Nombre inválido o vacío	Mensaje de error	correcto

Cuadro 6.9: Pruebas de caja negra asociadas a un usuario arqueólogo experto II.

Pruebas asociadas a un usuario arqueólogo experto III			
Prueba	Entrada	Salida	Resultado
Crear subcategoría	Nombre válido	Subcategoría creada y asignada a la categoría	Correcto
Crear subcategoría II	Nombre inválido o vacío	Mensaje de error	Correcto
Crear lugar	Datos válidos	Lugar creado con éxito	Correcto
Crear lugar II	Datos inválidos	Mensaje de error	Correcto
Crear localización	Datos válidos	Localización creada correctamente	Correcto
Crear localización II	Datos inválidos o falta algún dato	Mensaje de error	Correcto
Editar cremación	Datos actualizados válidos	Cremación actualizada con éxito	Correcto
Editar cremación II	Datos actualizados inválidos o falta código propio	Mensaje de error	Correcto
Eliminar cremación	Pulsar botón eliminar	Mensaje de borrado de la cremación	Correcto
Editar datos generales de unidad estratigráfica	Datos actualizados de la UE	Unidad estratigráfica actualizada con éxito	Correcto
Asociar o eliminar asociación de componente a unidad estratigráfica	Componente a asociar o a eliminar de asociados	Mensaje con asociación añadida o eliminada con éxito	Correcto
Asociar o eliminar asociación de superficie a unidad estratigráfica	Superficie a asociar o a eliminar de asociadas	Mensaje con asociación añadida o eliminada con éxito	Correcto
Asociar o eliminar asociación de artefacto(fósil) a unidad estratigráfica	Artefacto a asociar o a eliminar de asociados	Mensaje con asociación añadida o eliminada con éxito	Correcto

Cuadro 6.10: Pruebas de caja negra asociadas a un usuario arqueólogo experto III.

Pruebas asociadas a un usuario arqueólogo experto IV			
Prueba	Entrada	Salida	Resultado
Asociar o eliminar asociación de dieta de fauna a unidad estratigráfica	Artefacto a asociar o a eliminar de asociados	Mensaje con asociación añadida o eliminada con éxito	Correcto
Asociar o eliminar asociación de matriz de Harris a unidad estratigráfica	Añadir matriz de Harris con UE o eliminar una de la listas con las UE relacionadas	Mensaje con asociación añadida o eliminada con éxito	Correcto
Asociar o eliminar asociación de campo pendiente a unidad estratigráfica	Campo de la lista de disponibles o pendientes.	Mensaje con campo añadido o eliminado de pendientes con éxito	Correcto
Editar localización	Datos actualizados de la localización	Localización actualizada con éxito	Correcto
Eliminar localización	Pulsar botón eliminar	Localización borrada con éxito	Correcto
Asociar localización a lugar	Seleccionar lugar para asociar la localización	Localización asociada con éxito	Correcto
Eliminar unidad estratigráfica	Pulsar botón eliminar	Unidad estratigráfica eliminada con éxito	Correcto
Editar matriz de Harris	Datos actualizados de la matriz de Harris	Matriz de Harris actualizada con éxito	Correcto
Editar matriz de Harris II	Dato incorrecto o falta algún dato	Mensaje de error	Correcto
Eliminar matriz de Harris	Pulsar botón eliminar	Matriz de Harris eliminada correctamente	Correcto

Cuadro 6.11: Pruebas de caja negra asociadas a un usuario arqueólogo experto IV.

Pruebas asociadas a un usuario arqueólogo experto V			
Prueba	Entrada	Salida	Resultado
Editar analítica de faunas	Datos actualizados de la analítica de faunas	Analítica de faunas actualizada con éxito	Correcto
Editar analítica de faunas II	Datos no válidos o falta de algún dato	Mensaje de error	Correcto
Eliminar analítica de faunas	Pulsar botón eliminar	Analítica de faunas borrada con éxito	Correcto
Editar artículo	Datos actualizados del artículo	Artículo actualizado con éxito	Correcto
Editar artículo II	Datos no válidos o falta de algún dato	Mensaje de error	Correcto
Eliminar artículo	Pulsar botón eliminar	Artículo borrado con éxito	Correcto
Editar autor	Datos actualizados del autor	Autor actualizado con éxito	Correcto
Editar autor II	Datos no válidos o falta de algún dato	Mensaje de error	Correcto
Eliminar autor	Pulsar botón eliminar	Autor eliminado con éxito	Correcto
Editar multimedia	Datos actualizados del multimedia	Multimedia editado con éxito	Correcto
Editar multimedia II	Datos no válidos para el multimedia	Mensaje de error	Correcto
Eliminar multimedia	Pulsar botón eliminar	Multimedia eliminado con éxito	Correcto
Editar muestra	Datos actualizados de la muestra	Muestra actualizada con éxito	Correcto
Editar muestra II	Datos no válidos o falta el número de registro	Mensaje de error	Correcto
Eliminar muestra	Pulsar botón eliminar	Muestra eliminada con éxito	Correcto
Eliminar análisis metalográfico	Pulsar botón eliminar	Análisis metalográfico eliminado correctamente	Correcto

Cuadro 6.12: Pruebas de caja negra asociadas a un usuario arqueólogo experto V.

Pruebas asociadas a un usuario arqueólogo experto VI			
Prueba	Entrada	Salida	Resultado
Editar categoría	Datos actualizados de la categoría	Categoría actualizada con éxito	Correcto
Editar categoría II	Datos no válidos o falta la denominación	Mensaje de error	Correcto
Eliminar categoría	Pulsar botón eliminar	Categoría y subcategorías asociadas eliminadas con éxito	Correcto
Editar subcategoría	Nuevo nombre de la subcategoría	Nombre actualizado con éxito	Correcto
Editar subcategoría II	Nombre inválido o falta nombre	Mensaje de error	Correcto
Eliminar subcategoría	Pulsar botón eliminar	Subcategoría borrada correctamente	Correcto
Editar lugar	Datos actualizados del lugar	Lugar actualizado con éxito	Correcto
Editar lugar II	Datos incorrectos o falta el municipio	Mensaje de error	Correcto
Eliminar lugar	Pulsar botón eliminar	Lugar eliminado con éxito	Correcto
Ver mensajes en sala de expertos	Pulsar botón Sala de Expertos	Lista de mensajes en la sala de expertos	Correcto

Cuadro 6.13: Pruebas de caja negra asociadas a un usuario arqueólogo experto VI.

Pruebas asociadas a un usuario administrador			
Prueba	Entrada	Salida	Resultado
Crear usuario	Datos de usuario válidos	Usuario creado con éxito	Correcto
Crear usuario II	Datos de usuario no válidos o falta de algún dato	Mensaje de error	Correcto
Editar usuario	Datos de usuario actualizados	Usuario actualizado con éxito	Correcto
Editar usuario II	Datos no válidos	Mensaje de error	Correcto
Eliminar usuario	Pulsar botón eliminar	Usuario eliminado con éxito	Correcto
Ver lista de registros	Pulsar botón Registros	Muestra Lista de registros pendientes de validar	Correcto
Validar registro	Pulsar botón validar	Registro validado con éxito	Correcto

Cuadro 6.14: Pruebas de caja negra asociadas a un usuario administrador.

7 Planificación del proyecto

En este apartado se describe el plan de proyecto llevado a cabo con las respectivas actividades y fases de las que se conforma. Además se ha elaborado un plan de riesgos para atajarlos, si es que pudieran aparecer.

7.1. Gestión de los riesgos

En todo proyecto es necesario gestionar los riesgos que pueden presentarse para identificar los problemas potenciales antes de que ocurran. Este apartado es fundamental en cualquier proyecto, ya que la gestión de riesgos a veces es una inversión de futuro, debido a que muchas veces resulta más barato evitar un problema potencial que corregir uno que se ha producido[23]

7.1.1. Caracterización de los riesgos

Los riesgos que pudieran aparecer se han caracterizado de la siguiente manera

Tipos de riesgos

- **Riesgos de proyecto** : amenazan la planificación del proyecto. Si estos se cumplen, el proyecto requerirá mayor esfuerzo y coste.
- **Riesgos técnicos** : están relacionados con los aspectos técnicos debido a la implementación o el diseño de la aplicación. Si estos se cumplen, el proyecto es más complejo de lo estimado.
- **Riesgos de negocio** : Ponen en peligro todo el proyecto. Si estos se cumplen, el proyecto se cancelará.

Probabilidad de los riesgos

- **Muy alta** : >75%
- **Alta** : 50-75%
- **Media** : 30-50%
- **Baja** : 10-30%

Consecuencias de los riesgos : ordenados de mayor a menor impacto

- **Catastrófico** : si se produce, los objetivos del proyecto fracasarían
- **Crítico** : de producirse, el sistema final no tendría la calidad esperada.
- **Marginal** : se trataría de riesgos que afectarían a objetivos no principales
- **Despreciable** : se trata de riesgos que causarían problemas menores

7.1.2. Identificación de los riesgos y plan de acción

A continuación se detallan los riesgos identificados a partir de la caracterización definida en el apartado anterior

R01 - Incumplimiento de la planificación	
Descripción	se produce cuando ocurre algún imprevisto que causa retrasos y situaciones inesperadas a lo previsto
Tipo	Proyecto
Probabilidad	Media
Impacto	Crítico
Fase	Todas las fases
Plan de acción	Revisión de la planificación existente y reajustarla a la nueva situación

Cuadro 7.1: Descripción del riesgo R01

R02 - Requisitos poco definidos	
Descripción	los requisitos del sistema son muy vagos y generales, por lo que están poco definidos
Tipo	Proyecto
Probabilidad	Baja
Impacto	Crítico
Fase	Análisis
Plan de acción	Realizar una reunión con el cliente para analizar los requisitos y establecerlos con una mayor precisión

Cuadro 7.2: Descripción del riesgo R02

R03 - Pruebas insuficientes sobre el sistema	
Descripción	Las pruebas realizadas sobre el sistema son pocas como para detectar posibles errores y problemas que podrían aparecer después
Tipo	Técnico
Probabilidad	Media
Impacto	Crítico
Fase	Pruebas
Plan de acción	Realizar un plan de pruebas en la fase de implementación, para corregir errores que en fases posteriores son más difíciles de subsanar.

Cuadro 7.3: Descripción del riesgo R03

R04 - Fallo en la infraestructura de código del repositorio remoto (Github)	
Descripción	La infraestructura donde se aloja el código tiene un fallo y queda inutilizada temporalmente
Tipo	Proyecto
Probabilidad	Baja
Impacto	Crítico
Fase	Todas las fases
Plan de acción	Realizar las copias necesarias en el equipo local y migración del código a otra plataforma o esperar a que la plataforma actual vuelva a estar activa

Cuadro 7.4: Descripción del riesgo R04

R05 - Desconocimiento de alguna tecnología software utilizada	
Descripción	Alguna de las tecnologías que se va a utilizar no se conoce o bien se tiene poco conocimiento de ella
Tipo	Proyecto
Probabilidad	Baja
Impacto	Marginal
Fase	Análisis/Implementación/Pruebas/Despliegue
Plan de acción	Realizar actividades complementarias de aprendizaje de las herramientas a utilizar

Cuadro 7.5: Descripción del riesgo R05

R06 - Plan de trabajo poco definido	
Descripción	Alguna de las tecnologías que se va a utilizar no se conoce o bien se tiene poco conocimiento de ella
Tipo	Proyecto
Probabilidad	Media
Impacto	Crítico
Fase	Todas las fases
Plan de acción	Reajustar la planificación de forma que sea posible realizarla dentro del nuevo plan de trabajo

Cuadro 7.6: Descripción del riesgo R06

7.2. Plan de trabajo

El plan de trabajo de un proyecto se utiliza para organizar la estructuración y las fases del proyecto, utilizando las herramientas necesarias para ello. Para ello, implica la identificación y documentación de las actividades que lo van a conformar y su duración e interacción entre ellas para llegar a una planificación temporal realista y alcanzable

7.2.1. Diagrama de Gantt

Los diagramas de Gantt son una herramienta gráfica que se utilizan para visualizar el tiempo dedicado a las diferentes tareas de las que se compone un proyecto.

A continuación se muestra el listado de actividades que se han identificado durante la planificación y el diagrama de Gantt correspondiente

	Modo de	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
0		Ampliación y revisión de PintiaData	135 días?	lun 19/06/17	vie 22/12/17	
1		Inicio	6 días	lun 19/06/17	lun 26/06/17	
2		Definición de objetivos	2 días	lun 19/06/17	mar 20/06/17	
3		Realización del plan de proyecto	2 días	mié 21/06/17	jue 22/06/17	2
4		Puesta en marcha del sistema de control de versiones	2 días	vie 23/06/17	lun 26/06/17	3
5		Análisis	14 días	lun 26/06/17	jue 13/07/17	
6		Revisión de requisitos ya existentes	4 días	mar 27/06/17	vie 30/06/17	4
7		Identificación de nuevos requisitos	2 días	lun 03/07/17	mar 04/07/17	6
8		Elaboración del modelo de CU	2 días	mié 05/07/17	jue 06/07/17	7
9		Redacción y descripción de los CU	5 días	vie 07/07/17	jue 13/07/17	8
10		Diseño	18 días	jue 13/07/17	lun 07/08/17	
11		Modelo de dominio	2 días	jue 13/07/17	vie 14/07/17	
12		Diseño de la Base de Datos	2 días	lun 17/07/17	mar 18/07/17	11
13		Diagramas de secuencia	3 días	mié 19/07/17	vie 21/07/17	12
14		Rediseño de las interfaces a modificar	5 días	lun 24/07/17	vie 28/07/17	13
15		Diseño de la arquitectura del sistema	6 días	lun 31/07/17	lun 07/08/17	14
16		Implementación	59 días	mar 08/08/17	vie 27/10/17	
17		Implementación de las vistas	14 días	mar 08/08/17	vie 25/08/17	
18		Implementación de controladores y modelos	34 días	lun 28/08/17	jue 12/10/17	17
19		Desarrollo de nueva funcionalidad	8 días	vie 13/10/17	mar 24/10/17	18
20		Pruebas	18 días	mié 25/10/17	vie 17/11/17	19
21		Elaboración de pruebas de caja negra	6 días	mar 31/10/17	mar 07/11/17	
22		Corrección de errores	8 días	mié 08/11/17	vie 17/11/17	21
23		Despliegue	4 días	lun 20/11/17	jue 23/11/17	
24		Configuración de la máquina virtual del servidor de producción	2 días	lun 20/11/17	mar 21/11/17	
25		Lanzamiento a entorno de producción	2 días	mié 22/11/17	jue 23/11/17	24
26		Documentación	21 días	vie 24/11/17	vie 22/12/17	
27		Elaboración del documento de la memoria	21 días	vie 24/11/17	vie 22/12/17	25

Figura 20: Lista de actividades del proyecto

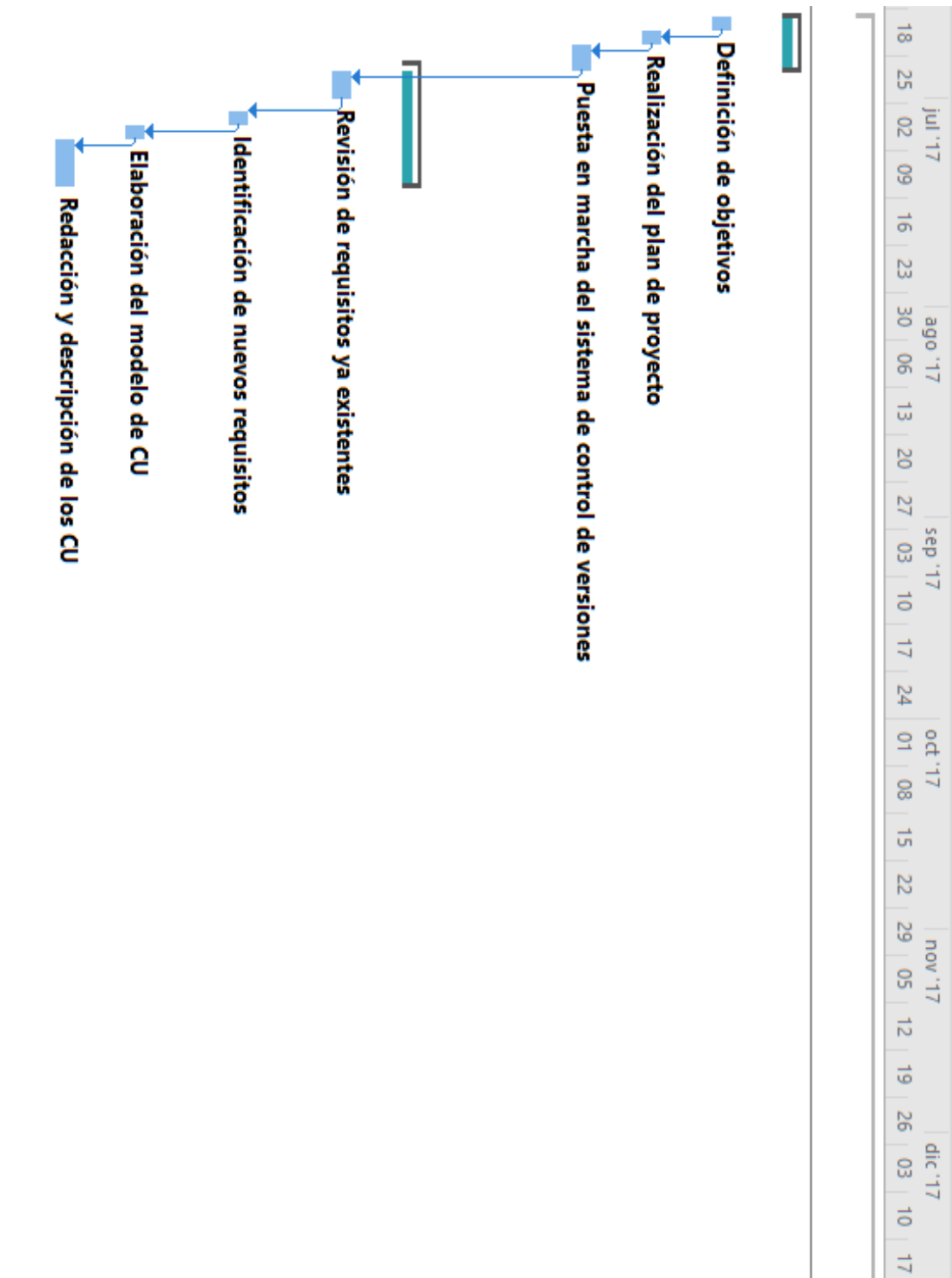


Figura 21: Diagrama de Gantt con la fase de Inicio y análisis

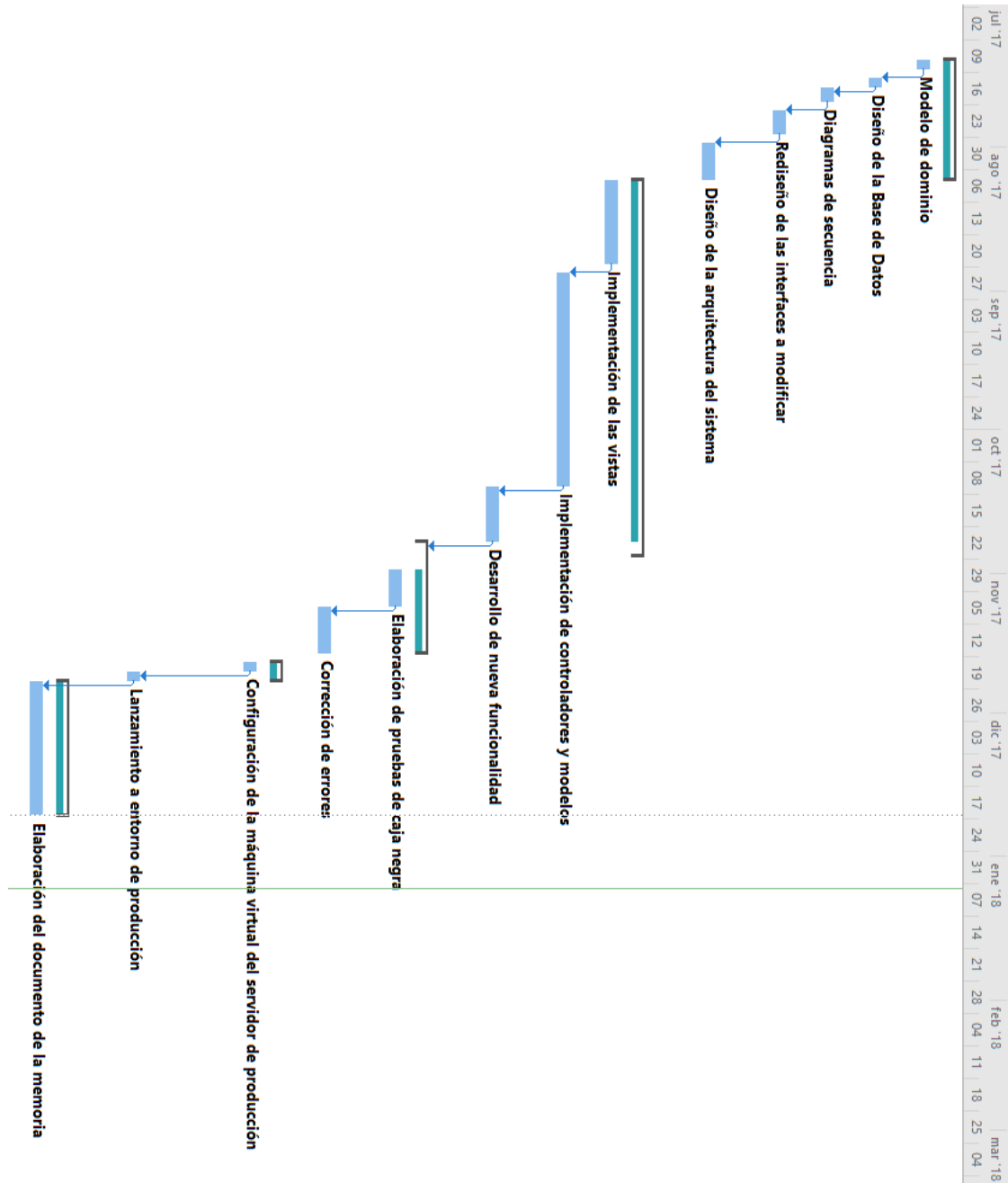


Figura 22: Diagrama de Gantt con la fase de diseño, implementación, despliegue y documentación

Como ya se ha dicho en el apartado de la gestión de riesgos, durante el proyecto siempre existe la posibilidad de que surge alguno y en este caso surgió el R01: Incumplimiento de la planificación, debido a unas prácticas en empresa extracurriculares durante la fase de Diseño e Implementación por lo que se tuvo que reajustar la planificación inicial a la nueva situación. Al cambiar la planificación solamente de la fase de Implementación y de Diseño vamos poner solo el diagrama de Gantt con la replanificación de estas fases:

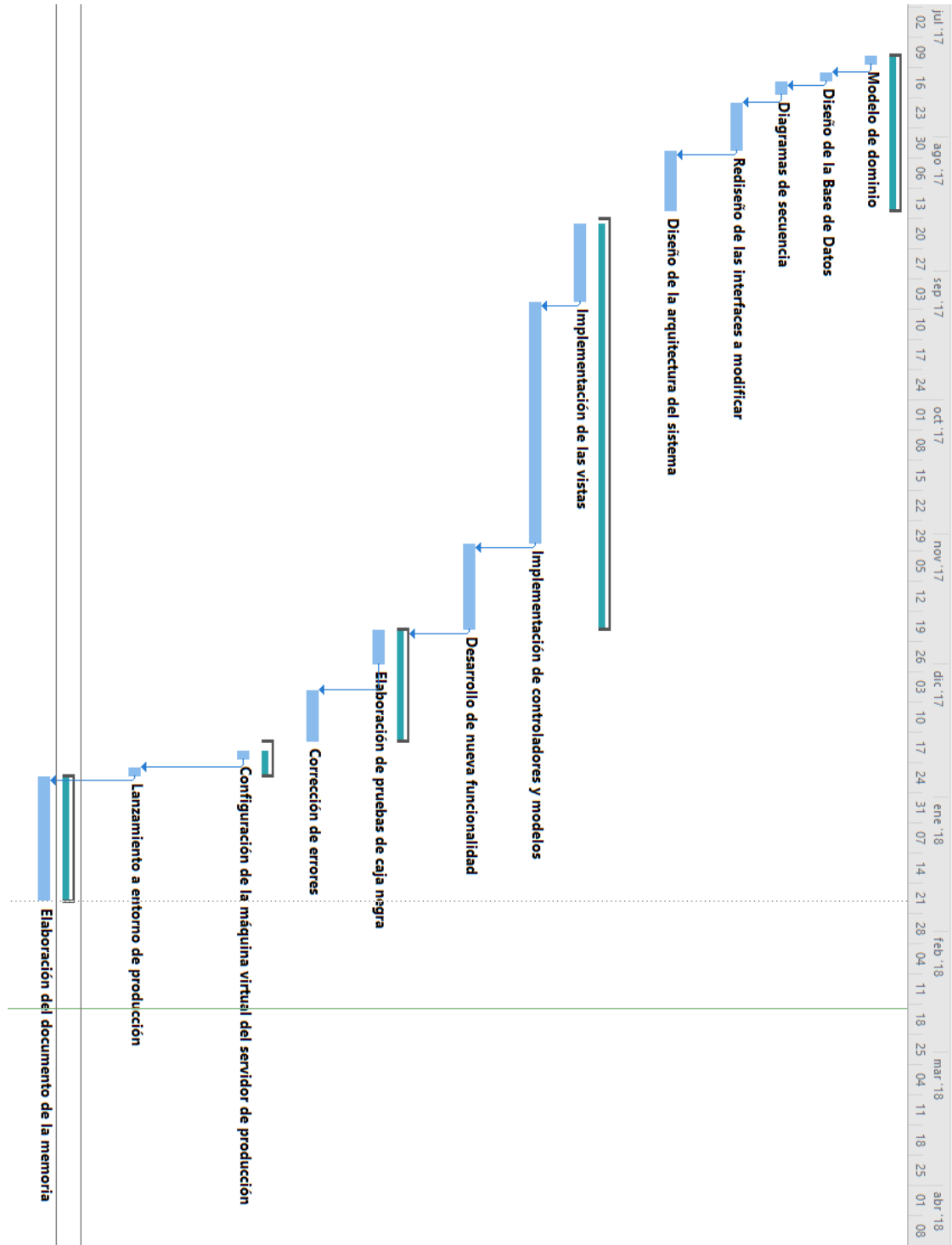


Figura 23: Diagrama de Gantt con la replanificación de la fase de Diseño y la fase de Implementación

8 Conclusiones

En este capítulo se extraen las conclusiones personales finales tras la realización del trabajo y el posible trabajo futuro que pueda surgir.

8.1. Conclusiones finales del trabajo

Tras la finalización del trabajo los objetivos planteados al inicio se han cumplido con éxito. Con la aplicación de un framework web y utilizando el patrón MVC, el código fuente ahora es más legible y mantenible. A partir de las especificaciones y necesidades planteadas se ha desarrollado nueva funcionalidad en la aplicación de acuerdo con las necesidades del cliente, además de corregir errores presentes en la anterior versión. Algunas de ellas son la introducción de nuevos filtros de búsqueda, el uso de AJAX para mejorar la eficiencia de la aplicación o el rediseño de las interfaces gráficas de multimedias y lista de objetos, donde ahora es mucho más visual y fomenta el uso de la aplicación.

Por otro lado se han hecho mejoras en aspectos de seguridad para evitar que los usuarios no autorizados editen información del sistema, así como permisos para acceder a ciertas partes de la aplicación.

A modo personal, este trabajo me ha servido para poner en práctica los conocimientos adquiridos durante el grado, siendo *Servicios y Sistemas Web*, *Diseño de software*, *Planificación y gestión de proyectos* y *Diseño de Bases de datos* las asignaturas con más relación. Por otro lado, he aprendido el funcionamiento de un framework como es Laravel y mejorar mis conocimientos con los lenguajes PHP y JavaScript.

8.2. Trabajo futuro

El desarrollo de esta aplicación es iterativo e incremental, en cuanto se desea añadir nuevas funciones y características nuevas en el sistema para cubrir nuevas necesidades.

Debido al carácter informativo y divulgativo de la web, se podrían añadir mejoras en estos aspectos

- Añadir nuevos filtros de búsqueda que no se hayan contemplado anteriormente
- Descarga o exportación a PDF de las fichas de objetos,tumbas...
- Implementación de una API Rest como servicio web para la comunicación con otras aplicaciones.

Todas estas sugerencias se pueden implementar de este trabajo y añadirlas en un futuro debido a que el framework permite ampliar fácilmente nueva funcionalidad que se requiera, aunque la API requeriría de un diseño y análisis inicial.

Bibliografía

- [1] Laura Rodríguez Martín. En: *PintiaData: Herramienta para la catalogación y consulta de piezas del yacimiento arqueológico de Pintia* (2015). URL: <https://uvadoc.uva.es/bitstream/10324/17259/1/TFG-G%201741.pdf> (último acceso 5 de diciembre de 2017).
- [2] Catálogo del Museo Británico. *Sitio web del Museo Británico*. URL: https://www.britishmuseum.org/research/collection_online/search.aspx (último acceso 2 de diciembre de 2017).
- [3] Museo Arqueológico de Atenas. *Sitio web del Museo Arqueológico de Atenas*. URL: <http://www.namuseum.gr/collections/index-en.html> (último acceso 2 de diciembre de 2017).
- [4] Museo Arqueológico Nacional. *Sitio web del Museo Arqueológico Nacional*. URL: <http://www.man.es/man/coleccion/catalogo-cronologico.html> (último acceso 2 de diciembre de 2017).
- [5] Colaboradores de Wikipedia. *Servidor Web*. URL: https://es.wikipedia.org/wiki/Servidor_web (último acceso 3 de diciembre de 2017).
- [6] Colaboradores de Wikipedia. *Sistema de gestión de Base de Datos*. URL: https://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_bases_de_datos (último acceso 3 de diciembre de 2017).
- [7] Colaboradores de Wikipedia. *HTML*. URL: <https://es.wikipedia.org/wiki/HTML> (último acceso 3 de diciembre de 2017).
- [8] Colaboradores de Wikipedia. *CSS*. URL: https://es.wikipedia.org/wiki/Hoja_de_estilos_en_cascada (último acceso 3 de diciembre de 2017).
- [9] Colaboradores de Wikipedia. *JavaScript*. URL: https://es.wikipedia.org/wiki/JavaScript#Uso_en_p.C3.A1ginas_web (último acceso 4 de diciembre de 2017).
- [10] W3 Schools. *AJAX*. URL: https://www.w3schools.com/xml/ajax_intro.asp (último acceso 4 de diciembre de 2017).

- [11] Colaboradores de Wikipedia. *SQL*. URL: <https://es.wikipedia.org/wiki/SQL> (último acceso 4 de diciembre de 2017).
- [12] José María Galindo Haro. En: *Diseño e implementación de un marco de trabajo (framework) de presentación para aplicaciones JEE* (2008), págs. 38-40. URL: <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/876/1/00765tfc.pdf> (último acceso 4 de diciembre de 2017).
- [13] Taylor Otwell. *Laravel*. URL: <https://laravel.com/> (último acceso 4 de diciembre de 2017).
- [14] Nils Adermann y Jordi Boggiano. *Composer - Dependency Manager for PHP*. URL: <https://getcomposer.org/> (último acceso 5 de diciembre de 2017).
- [15] HotFrameworks. *PHP - Web framework rankings | HotFrameworks*. URL: <https://hotframeworks.com/languages/php> (último acceso 5 de diciembre de 2017).
- [16] Yania Crespo. *Apuntes de la asignatura de diseño de software 2015-2016, Capítulo 4 - MVC*. Universidad de Valladolid. (Último acceso 7 de diciembre de 2017).
- [17] Colaboradores de Wikipedia. *Active Record*. URL: https://en.wikipedia.org/wiki/Active_record_pattern (último acceso 7 de diciembre de 2017).
- [18] Taylor Otwell. *Laravel - Eloquent ORM*. URL: <https://laravel.com/docs/5.6/eloquent> (último acceso 7 de diciembre de 2017).
- [19] Colaboradores de Bootstrap. *Bootstrap - The most popular HTML, CSS, and JS library in the world*. URL: <https://getbootstrap.com/> (último acceso 5 de diciembre de 2017).
- [20] Alberto Macho de la Riva. En: *PintiaDatApp: catálogo móvil del sitio arqueológico de Pintia* (2016). URL: <http://uvadoc.uva.es/bitstream/10324/20935/1/TFG-G%202246.pdf> (último acceso 2 de septiembre de 2017).
- [21] Linus Torvalds. *Git*. URL: <https://git-scm.com/> (último acceso 10 de diciembre de 2017).
- [22] Github Team. *The world's leading software development platform · GitHub*. URL: <https://github.com/> (último acceso 10 de diciembre de 2017).
- [23] Pablo de la Fuente. *Apuntes de la asignatura de planificación y gestión de proyectos 2016-2017, Gestión de riesgos*. Universidad de Valladolid. (Último acceso 10 de diciembre de 2017).

Anexos

Apartado dedicado a los anexos correspondientes a esta memoria, los cuales son la lista de acrónimos y un breve manual para el programador.

Anexo 1. Lista de Acrónimos

- AJAX** Asynchronous JavaScript And XML
- CRUD** Create Read Update Delete
- CSS** Cascading Style Sheet
- DDL** Data Definition Language
- DML** Data Manipulation Language
- DOM** Document Object Model
- HTML** HyperText Markup Language
- HTTP** Hypertext Transfer Protocol
- LAMP** Linux, Apache, MySQL/MariaDB, PHP/Perl/Python
- MAMP** Mac OS, Apache, MySQL, PHP/Perl/Python
- MVC** Modelo Vista Controlador
- ORM** Object Relational Mapping
- PHP** Hypertext Preprocessor
- SGBD** Sistema Gestor de Bases de Datos
- SQL** Structured Query Language
- URI** Uniform Resource Identifier
- URL** Uniform Resource Locator
- WAMP** Windows, Apache, MySQL, PHP/Perl/Python

Anexo 2. Manual del programador

En este anexo se describen los pasos a seguir para instalar el proyecto en un entorno local, bajo un sistema operativo Windows 10. Los pasos descritos a continuación.

- En primer lugar debemos descargar XAMPP que instalará en nuestro sistema, el servidor web Apache, el sistema gestor de Base de Datos MySQL y el lenguaje PHP. Hay que tener en cuenta que hay que descargar una versión con PHP 5.6 o superior para que el proyecto pueda funcionar. El link es el siguiente:

```
https://www.apachefriends.org/es/index.html
```

- Después, debemos instalar Composer que gestionará las dependencias necesarias para nuestro proyecto:

```
https://getcomposer.org/download/
```

- A continuación, debemos descargar e instalar Git que nos permitirá traer el código fuente de la aplicación :

```
https://git-scm.com/
```

- Acto seguido, nos traemos a nuestro equipo el código fuente que reside en el repositorio remoto, mediante el siguiente comando de git:

```
git clone https://github.com/rodnune/Pintia.git
```

- En la carpeta htdocs dentro de la carpeta donde hemos instalado XAMPP, abrimos el Símbolo del Sistema. Ahora Composer se encargará de resolver las dependencias necesarias para nuestro proyecto. Para ello ejecutar el siguiente comando:

```
composer install
```

- A continuación debemos crear el fichero `.env` dentro de la carpeta raíz del proyecto. Este fichero define el entorno y la configuración del proyecto. Con el paso anterior las únicas líneas que tendremos que modificar serán `DB_NAME`, `DB_USERNAME` y `DB_PASSWORD`, con los datos de la base de datos que creemos.

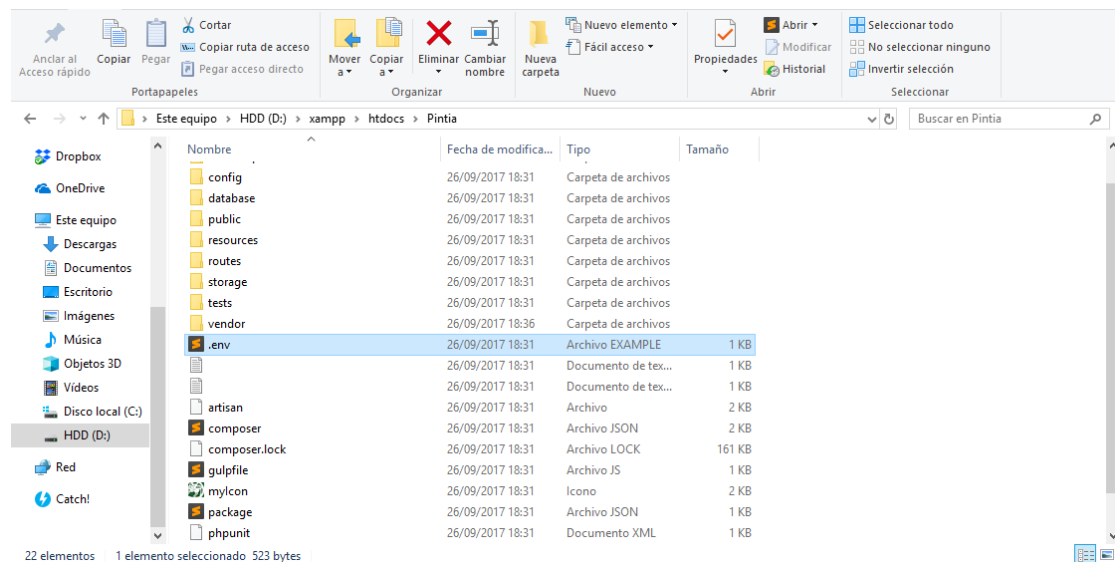


Figura 24: Aspecto de las estructura del proyecto remarcando el fichero `.env`

- Por último, arrancamos el servidor web Apache y MySQL que dejará listo el proyecto para empezar a recibir peticiones. Para ello, si todo ha ido correctamente, en un navegador Web introducimos como URL `http://localhost:8080` y aparecerá la página inicial de la Web Pintiadata dentro de un entorno local

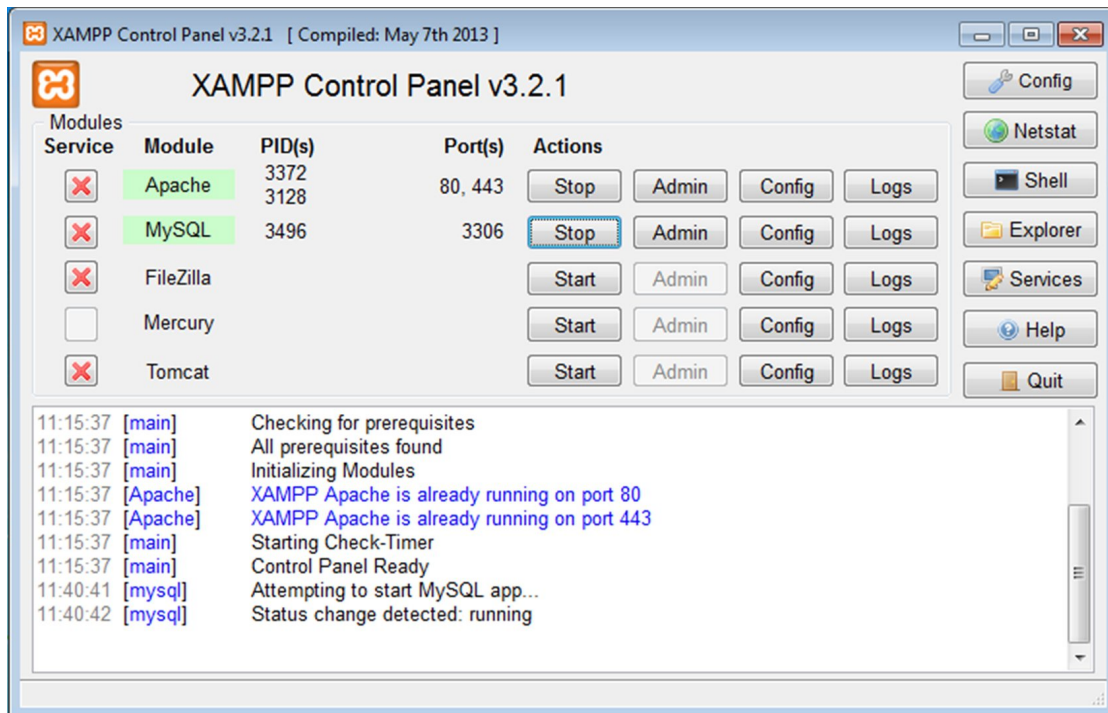


Figura 25: Aspecto de XAMPP con servidor Apache y MySQL arrancados



Figura 26: Aspecto final con la aplicación instalada en un entorno local