



Portal Web Ayuntamiento UOC - Trámites

Ana Isabel Armenteros Barroso
Grado de Ingeniería Informática

Gregorio Robles Martínez

09.06.2022



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc/3.0/es/)

B) GNU Free Documentation License (GNU FDL)

Copyright © 2022 Ana Isabel Armenteros Barroso

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

C) Copyright

© (Ana Isabel Armenteros Barroso)

Reservados todos los derechos. Está prohibido la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reprografía, el microfilme, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler y préstamo, sin la autorización escrita del autor o de los límites que autorice la Ley de Propiedad Intelectual.

FICHA DEL TRABAJO FINAL

Título del trabajo:	Portal Web Ayuntamiento UOC - Trámites
Nombre del autor:	Ana Isabel Armenteros Barroso
Nombre del consultor:	Gregorio Robles Martínez
Fecha de entrega (mm/aaaa):	03/2022
Área del Trabajo Final:	Desarrollo Web
Titulación:	<i>Grado Ingeniería Informática</i>

Resumen del Trabajo (máximo 250 palabras):

Este trabajo consiste en la creación de un portal web donde los ciudadanos puedan acceder fácilmente a los trámites de carácter burocrático que son indispensables realizar a través del Ayuntamiento, tales como el censo o inscripciones a eventos entre otros.

También será una guía o manual por parte de los trabajadores del Ayuntamiento encargados de tramitar dichos documentos, ya que, son trámites complejos en los que se necesita tiempo para formar al personal y, por el momento, no es posible debido a la gran cantidad de trámites y el poco personal del que se cuenta.

Por otro lado, al ser un generador de formularios, puede facilitar la tarea al personal de este Ayuntamiento que se encarga de generar dichos documentos.

Del mismo modo, se pueden encontrar funcionalidades extras para dar una mayor calidad al citado portal web, como, por ejemplo, la consulta de bandos, noticias y novedades, un área de deportes, área multimedia o datos de contacto entre otros. Con ello se pretende que la página web no solo sea una página de gestiones tanto para el ciudadano como para los trabajadores, si no que, se pretende que pueda proporcionar al ciudadano todo tipo de facilidades sobre actividades que gestiona o realiza el Ayuntamiento UOC.

Para conseguir el desarrollo de esta página web se han utilizado programas como XAMPP, Visual Studio Code, PHPMyAdmin, así como que ha sido programada en HTML, PHP y SQL.

Abstract (in English, 250 words or less):

This work consists of the creation of a web portal where citizens can easily access the bureaucratic procedures that are essential to carry out through the City Council, such as the census or registration for events among others.

It will also be a guide or manual for the City Council workers in charge of processing these documents, since they are complex procedures in which time is needed to train the staff and, for the moment, it is not possible due to the large number of procedures and the little staff that is counted.

On the other hand, being a generator of forms, it can facilitate the task of the staff of this City Council who is in charge of generating said documents.

In the same way, you can find extra functionalities to give a higher quality to the aforementioned web portal, such as, for example, the consultation of factions, news and novelties, a sports area, another multimedia área or contact data among others. This is intended to make the website not only a management page for both citizens and workers, but also to provide citizens with all kinds of facilities on activities managed or carried out by the UOC City Council.

To achieve the development of this web page, programs such as XAMPP, Visual Studio Code, PHPMyAdmin have been used, as well as programming in HTML, PHP and SQL.

Palabras clave (entre 4 y 8):

Portal web, ayuntamiento, XAMPP, HTML, SQL, desarrollo web, PHPMyAdmin, trámites.

Índice

1. Introducción.....	1
1.1 Contexto y justificación del Trabajo.....	1
1.2 Objetivos del Trabajo.....	3
1.4 Planificación del Trabajo	5
1.4.1 Recursos utilizados.....	5
1.4.2 Planificación.....	8
1.4.3 Entregas.....	11
1.5 Breve sumario de productos obtenidos	12
2. FASE DISEÑO	14
2.1. Fuentes portal web	14
2.2. Diagramas bases de datos.....	15
2.3. Captura de requisitos	16
2.4. Arquitectura del sistema portal web	17
2.5. Estructura portal principal web	18
3. FASE DESARROLLO	19
3.1. Fase desarrollo base de datos	19
3.2. Fase desarrollo web	23
3.2.1. Estructura básica portal Ayuntamiento UOC.....	24
3.2.2. Portal Ciudadano	28
3.2.3. Portal Área restringida	31
3.2.4. Portal administrador.....	34
3.2.5. Trámites online	35
3.2.6. Área noticias	36
3.2.7. Área multimedia	37
3.2.8. Área novedades.....	39
3.2.9. Contacta con nosotros	41
3.2.10 Sobre nosotros.....	42
3.2.11. Ofertas de empleo	43
3.2.12. Área de deportes	44
3.2.13. Área bandos.....	45
4. FASE TESTEO.....	47
5. Conclusiones.....	48
6. Glosario	49
7. Bibliografía	50

1. Introducción

1.1 Contexto y justificación del Trabajo

En la actualidad la demanda de trámites que tienen que se tiene que realizar desde un Ayuntamiento es de bastante importancia. Muchos de ellos requieren asistir presencialmente a las sedes de estos edificios públicos para gestionarlo, pero los horarios que ofrecen no son idóneos para poder realizarlo, pues, las personas que demandan dichos trámites suelen tener la misma jornada laboral que estos funcionarios y se hace complicado poder gestionar los citados trámites.

Cabe reseñar que muchos de estos trámites podrían solventarse si se ofrecieran de manera online, ya que, la mayoría de ellos consisten en formularios que deben ser rellenos por los ciudadanos y por ello, no es necesario asistir a la sede para obtener dicho formulario y rellenarlo.

Además, se observa que existen grandes colas para realizar estos trámites de manera presencial, y no se debe olvidar que, en la actualidad existe una pandemia mundial que no permite la masificación de gente, por lo que estas situaciones se deben evitar.

Ya han sido algunos Ayuntamientos conscientes de la problemática existente y han iniciado proyectos de páginas webs para realizar este tipo de trámites, pero las mismas resultan ser muy pobres o, no permiten realizar todos los trámites aconsejables, si no que, se dedican a realizar un muestrario de lo que ofrecen de manera presencial.

Es por ello, que se pretende desarrollar una página web diferente donde realmente el ciudadano pueda gestionar sus trámites de manera online sin tener que acudir en ningún momento de manera presencial al Ayuntamiento.

Además, se tiene como objetivo que no sirva sólo para tramitar dicha petición del ciudadano si no que pueda consultar en todo momento el estado de su trámite. También debe servir para que los trabajadores funcionen de manera grupal, atendiendo a las demandas que quedan pendiente en su portal y no realizar trabajos individuales como suele suceder de manera presencial, existiendo por tanto una gran descoordinación.

Ejemplos de páginas de Ayuntamientos que no cumplen con lo necesario:

En la Figura 1 se muestra el Ayuntamiento de Borox, donde se ofrece una sede electrónica y un catálogo de trámites, pero la amplia mayoría de dichos trámites que se ofrecen no se pueden realizar vía online [1].



Figura 1: Imágen de la portal web del Ayuntamiento de Borox, sección de trámites

El portal web del Ayuntamiento de Antequera se puede observar en la Figura 2 donde allí se pueden realizar tres trámites a pesar de la gran población que tienen [2]:



Figura 2: Imágen de la portal web del Ayuntamiento de Antequera, sección de trámites

1.2 Objetivos del Trabajo

El objetivo principal de este trabajo es la puesta en marcha de un portal web de trámites relacionados con el Ayuntamiento donde puedan operar, tanto los ciudadanos, como el personal laboral del citado edificio público. Para ello, se crean diferentes roles de usuarios para que, dependiendo de si es ciudadano o trabajador, puedan acceder a espacios donde puedan realizar las tareas que desean. El rol de ciudadano tendrá una visión básica de la web, mientras que, el rol de trabajador podrá acceder a casi la totalidad de los trámites existentes en el portal. Por último, se creará un rol de administrador, que será el encargado de poder generar nuevas acciones en la web.

Como subobjetivos se pretende mejorar la eficacia del Ayuntamiento, gracias a este portal web se reducen los tiempos de espera ya que los ciudadanos podrán hacer muchos de los trámites desde sus casas sin necesidad de tener que trasladarse al edificio público.

Por otro lado, se ha pretendido unificar los criterios entre los propios trabajadores del Ayuntamiento ya que, con ese portal se consigue que todos utilicen la misma metodología para trabajar y todos puedan coordinarse.

También se ha tratado de adaptar este Ayuntamiento a las tecnologías informáticas, reduciendo el papel y prosperando a la transformación digital del mismo apoyando de este modo el medio ambiente y, además reforzando una nueva imagen del citado Ayuntamiento.

Para finalizar se ha intentado que con la puesta en marcha de este portal web, se puedan ir incorporando nuevas acciones o trámites que puedan terminar de informatizar el citado Ayuntamiento. Esas mejoras han consistido en ser un portal de noticias, novedades, multimedia o portal de pagos entre otros.

1.3 Enfoque y método seguido

Observando las páginas web que se ofrecen en la actualidad referidas a este ámbito, se ha decidido realizar un ámbito más abierto, donde se ofrezcan muchas más posibilidades y pueda estar continuamente actualizándose para introducir mejoras.

Por otro lado, se ha valorado realizarlo mediante aplicación web o móvil, pero finalmente se ha optado por realizarla de manera web ya que así puede llegar al alcance de más usuarios. Aunque, hoy en día la gran mayoría de gente tiene teléfonos móviles y podría descargarse la app, se decide realizarla vía web, pues muchas veces es necesario un entorno más amplio y manejable a la hora de poder efectuar los trámites que se podrán realizar desde el portal web [3].

Además de lo anterior, se ha valorado realizarlo de una manera más cómoda, utilizando plantillas como Prestashop, pero se ha optado por personalizar totalmente la página web y programarla al 100%, tanto en el desarrollo de la misma, como en el diseño y conexión con la base de datos.

Por todo ello, se ha escogido un método de iniciar el desarrollo de este portal web con las bases de datos, a continuación, se ha realizado el esqueleto del portal web. Después de conseguir dicho objetivo se ha tratado de realizar las conexiones entre bases de datos y portal web. Y, para finalizar, se han desarrollado las subpáginas para lograr el fin indicado.

Como conclusión se indica que se ha optado por usar los lenguajes de programación de SQL, HTML y PHP, ya que son los más estandarizados para el desarrollo de páginas web y, cuando necesite actualizaciones la página en un futuro, podrá continuar programándose por cualquier desarrollador de páginas webs.

En la fase de pruebas, se optó por realizar nuevas mejoras en las subpáginas y trazar una nueva estrategia sobre el lado administrador para darle un toque más elevado de seguridad.

Por tanto, al finalizar el trabajo y gracias a la planificación final realizada se ha conseguido los objetivos iniciales de la propuesta además de aportar nuevas funcionalidades al portal web así como darle un toque más de seguridad. Esto da una garantía importante a todos los usuarios de la página, tanto los ciudadanos como los propios trabajadores y gestores de la web.

1.4 Planificación del Trabajo

1.4.1 Recursos utilizados

En el presente Trabajo de Fin de Grado ha sido necesaria la utilización de ciertos recursos informáticos que son caracterizados por ser básicos para poder realizar el desarrollo del portal web. Estos constan de recursos como programas o recursos como lenguajes de programación.

Además, ha sido necesario recurrir a otro tipo de utilidades que son de carácter complementario pero a la vez son necesarios para realizar el desarrollo básico del portal. Estos recursos que se mencionan son del tipo de diagramas, tales como los diagramas UML, captura de requisitos o estructura del portal entre otros, los cuales se verán expresados en la fase de diseño del presente proyecto ya que se trata de diagramas realizados sin ningún tipo de software.

En cuanto a los recursos informáticos básicos se ha tratado de expresar a continuación los utilizados para la creación del portal web. Además, se realiza una breve descripción de la utilidad de estos:



- XAMPP, tratándose de un software libre que gestiona el servidor web de Apache, las bases de datos MySQL y lenguajes de programación, tales como el que se va a utilizar para el desarrollo de la página PHP.

Con XAMPP se pueden crear distribuciones que resultan de instalación sencilla. Además se pueden crear configuraciones por defecto y resulta ser de gran interés por ser un software gratuito. Para finalizar destacar que es un aplicativo multiplataforma soportando actualmente los sistemas operativos de Windows, Linux y OS X [4].



- phpMyAdmin. Se trata de una herramienta para administrar con mayor facilidad bases de datos como MySQL. Esto se permite gracias a que se maneja a través de páginas web. La misma se encuentra escrita en PHP.

Además phpMyAdmin por ser también un software libre disponible en multilenguaje, siendo capaz de soportar otro tipo de bases de datos como MariaDB y Drizzle. También entre sus ventajas se encuentra la posibilidad de exportar los datos en diversos formatos como CSV, SQL, XML entre otros [5].



- HTML, lenguaje de programación para el desarrollo de la página web del Ayuntamiento. Se trata del componente más elemental de una página web. Suele ser utilizado con otras tecnologías como CSS para desarrollar la apariencia o bien JavaScript para desarrollar funcionalidades.

Se pueden diferenciar cada uno de los elementos HTML gracias a sus etiquetas que van marcadas por los símbolos < >. Con ello se garantiza poder definir la estructura de la página web y el significado del contenido web [6].



- PHP, lenguaje de programación para el desarrollo web. Este lenguaje afectará también a la programación de la página web del Ayuntamiento.

Este lenguaje se ejecuta en el servidor hacia el cliente. PHP se caracteriza por ser bastante simple además de ofrecer características más complejas. Para poder utilizar este lenguaje se debe incrustar el código entre etiquetas del tipo <?php ?> [7].



- Se utiliza el programa Visual Studio Code para realizar el desarrollo de la página web del Ayuntamiento. Se trata de un editor para el desarrollo de programas o páginas web. Se caracteriza por ser multiplataforma y multilinguaje.

Entre las ventajas que encontramos es que es un editor de código bastante simple, lo que facilita a los desarrolladores mejor creación de contenido. Es capaz de realizar funciones tales como edición, compilación y depuración. Para finalizar con esta herramienta indicar que se trata de un software completamente gratuito [8].



- Denominado CSS. Este tipo de lenguaje se usa para poder aportar un formato y estilo a la página web. Entre las características que se pueden estilizar a través de este lenguaje se encuentra el color, el tipo de fuente, el tamaño, la orientación, etc.

Gracias a poder contar con dos tipos de lenguaje a la hora de la creación de páginas web (CSS y HTML) es posible modificar el estilo y formato sin que se vea afectado el contenido de la programación [9].



- Lenguaje utilizado para las bases de datos, su creación, modificación y consultas de los datos almacenados en ellas. Con SQL se puede realizar la descripción de un conjunto de datos para así poder responder preguntas.

Es muy importante escribir la sintaxis de la pregunta de manera correcta debido a que no podrá arrojar resultados si está mal ejecutada. La sintaxis debe escribirse en inglés y cumplir las reglas que establece este tipo de lenguaje de computación [10].

1.4.2 Planificación

Inicialmente se creó el siguiente Diagrama de Gantt que se ha tratado de cumplir en su totalidad el cual se muestra en la Figura 3:



Figura 3: Diagrama de Gantt realizado en la propuesta de tema de TFG.

Estas tareas a su vez se han agrupado en tres fases para la cumplimentación de los objetivos:

FASE DISEÑO:

Se trata de crear el diseño de la interfaz del portal y de la base de datos para ello se realizan las siguientes tareas:

- Búsqueda de fuentes que se utilizarán para el portal web.
- Diseño de diagramas para crear la estructura de la base de datos.
- Diseño de diagramas para crear la estructura del portal web.

FASE DE DESARROLLO:

En esta fase se crean diversos puntos para empezar a tener un esqueleto del portal web, que son los siguientes:

- Creación de base de datos de Ayuntamiento:

Se tratará de crear una base con los roles que se encontrarán en la página web. A través de MySQL – PhpMyAdmin.

Los roles que se han creado son los de administrador, ciudadano y trabajador. Cada uno de ellos tendrá opciones de acceder a diferentes acciones.

- Estructura básica de la web del Ayuntamiento:

Consiste en indicar en la página principal una visión global de las funcionalidades que ofrece el portal web.

- Desarrollo del portal desde el lado ciudadano:

Donde se ubica la pantalla de logueo del portal del lado ciudadano para que pueda observar sus trámites en curso. Además, se ofrece la posibilidad de que si no están registrados puedan realizar la creación de una cuenta.

- Desarrollo del portal desde el lado Ayuntamiento:

Al acceder a esta área restringida se tendrá que acceder a través de un logueo, sin posibilidad de crear cuenta si no estás registrado ya que esta función será misión de los administradores. Una vez que se accede con usuario y contraseña se podrá acceder a las solicitudes de trámites que estén pendientes.

- Inserción de trámites / formularios:

Aparecerán diferentes opciones de cara al portal del ciudadano mostrando en tablas diferentes formularios donde tendrán una opción para poder ejecutar el citado trámite.

- Desarrollo del portal desde el lado Administrador:

Se tratará de programar una página donde puedan realizar labores de mantenimiento de la página los administradores. En este caso, se inician esas labores con la creación de usuarios desde dicha área.

- Extra: apartado noticias / bandos:

Se trata de poner sin necesidad de loguearse donde aparezcan novedades del ayuntamiento que sean de interés para los ciudadanos residentes en la localidad.

- Extra: apartado multimedia:

En esta área se muestra una galería de fotografías de eventos que se han realizado en la localidad. Para acceder a este apartado no existe la necesidad de loguearse.

FASE DE TESTEO:

En esta fase se realizan diversas pruebas para observar que el portal web funcione de manera correcta.

No obstante, se plasma en la tabla de la Tabla 1 para una mayor apreciación de fechas:

Objetivo	Inicio	Fin	Días
Diseño base de datos de Ayuntamiento	01.03.2022	10.03.2022	9
Estructura básica de la web del Ayuntamiento	11.03.2022	31.03.2022	20
Entrega de PEC 2			
Diseño portal desde el lado ciudadano	01.04.2022	15.04.2022	14
Diseño portal desde el lado Ayuntamiento	16.04.2022	30.04.2022	14
Inserción de trámites / formularios	01.05.2022	15.05.2022	14
Entrega de PEC 3			
Diseño portal lado Administrador	16.05.2022	31.05.2022	15
Extra: apartado noticias / bandos	01.06.2022	09.06.2022	8
Extra: apartado pagos / tpv	01.06.2022	09.06.2022	8
Extra: apartado multimedia	01.06.2022	09.06.2022	8
Entrega de PEC 4			

Tabla 1: Planificación inicial entregada en la propuesta del TFG

Durante el desarrollo del Proyecto y debido a circunstancias varias entre las que se encuentran las cuestiones laborales, la inclusión de las correcciones propuestas por el tutor y la nueva toma de decisiones para aportar calidad al trabajo, se optó por realizar una serie de cambios, resaltados en color azul que quedan plasmados en la Tabla 2. Cabe resaltar que los objetivos se han logrado de igual manera e incluso se ha tratado de darle un toque mayor calidad.

Objetivo	Inicio	Fin	Días
Diseño base de datos de Ayuntamiento	01.03.2022	10.03.2022	9
Estructura básica de la web del Ayuntamiento	11.03.2022	31.03.2022	20
Entrega de PEC 2			
Corrección de errores comentados por el tutor	01.04.2022	04.04.2022	3
Diseño portal desde el lado ciudadano	05.04.2022	10.04.2022	5
Diseño portal desde el lado Ayuntamiento	11.04.2022	16.04.2022	5
Inserción de trámites / formularios	17.04.2022	22.04.2022	5
Extra: apartado noticias / bandos	22.04.2022	27.04.2022	5

Extra: apartado multimedia	28.04.2022	03.05.2022	5
Extra: empleo	04.05.2022	07.05.2022	3
Extra: deportes	08.2022	10.05.2022	3
Extra: contacta con nosotros	10.05.2022	13.05.2022	3
Ajuste a modelo TFG proporcionado por la UOC	14.05.2022	15.05.2022	1
Entrega de PEC 3			
Corrección de errores comentados por el tutor	16.05.2022	31.05.2022	15
Diseño portal lado Administrador	31.05.2022	04.06.2022	4
Finalización de memoria	05.06.2022	06.06.2022	1
Realización de presentación	07.06.2022	08.06.2022	1
Realización de grabación	08.06.2022	09.06.2022	1
Formulario autoevaluación	08.06.2022	09.06.2022	1
Entrega de PEC 4			

Tabla 2: Planificación final para el TFG

1.4.3 Entregas

PEC1. Se realiza una propuesta de trabajo, en la que se habla de los objetivos a alcanzar en el desarrollo de un portal web dedicado a la realización de trámites de un Ayuntamiento. También se habla de la metodología a utilizar, como es el lenguaje de programación y los programas que se van a utilizar.

PEC2. Una vez aceptada la propuesta de trabajo. Se adapta la citada propuesta a la plantilla de memoria que se entregará como entrega final. Para ello se irá alimentando entrega a entrega con los hitos conseguidos gracias al desarrollo web que se ha planteado.

En este caso, se ha realizado entrega del desarrollo de la base de datos, su conexión con la página web, el desarrollo de la página web principal, y como funciones adicionales, se ha entregado la subpágina de contacto y sobre nosotros.

PEC3. Se realiza entrega del desarrollo de los portales lado ciudadano y área restringida. En el interior de ellos se procede a la creación del login y logout así como una sección para registrarse. Además de lo anterior y con lo que se cumplimentaría los objetivos de esta PEC, se desarrolla el espacio de trámites online, donde se realiza una tabla con los distintos formularios existentes por el momento en la página web. También se subsanan los errores que comenta el tutor para la corrección de la PEC2, insertando, por ejemplo, diversos diagramas.

Adicionalmente y por existir tiempo para ello, se crean las secciones restantes indicadas en la página principal.

Como se observa en este entregable, se ha continuado con el seguimiento del plan con exactitud. No obstante, se han añadido secciones

adicionales no contempladas en el plan pero que, a vista de la autora del Trabajo de Fin de Grado, eran necesarios para realizar una página web con mayores características de las demandas y que resultase un portal más completo. Aunque bien es sabido que siempre debe estar en constante desarrollo y con la introducción de errores y actualizaciones cuando la misma se encuentre activa para el ciudadano.

PEC4. Tras el aporte de valoraciones de la última entrega del profesor – tutor se realizan las mejoras que el mismo indica así como la corrección de errores que detecta.

Además de lo anterior, en esta PEC se realiza el cuestionario de autoevaluación, una presentación donde se efectúa una explicación del proyecto realizado así como la creación de un vídeo donde se aporta comentarios por parte de la presente alumna.

1.5 Breve resumen de productos obtenidos

El portal web se forma gracias a las siguientes páginas:

- Secciones que se muestran en la Figura 4:

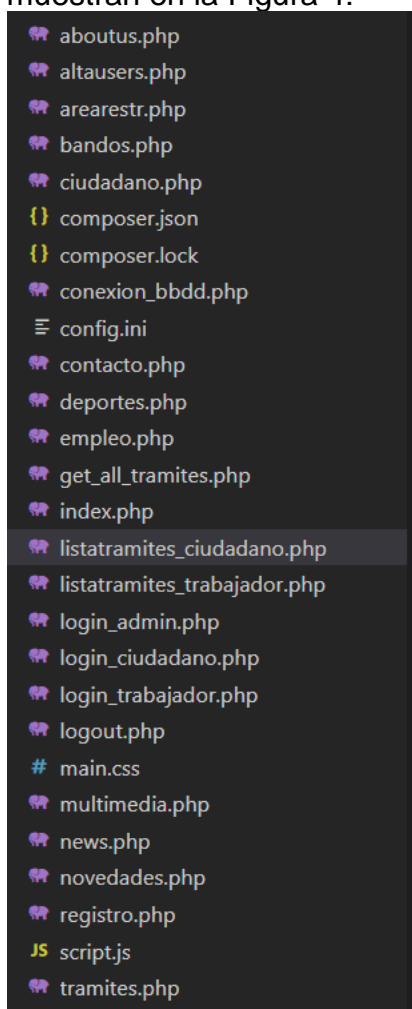


Figura 4: Captura de pantalla donde se muestran cada uno de los archivos PHP creados para la estructura del portal web

- Imágenes que se pueden observar en el portal web, se muestran a continuación en la Figura 5:

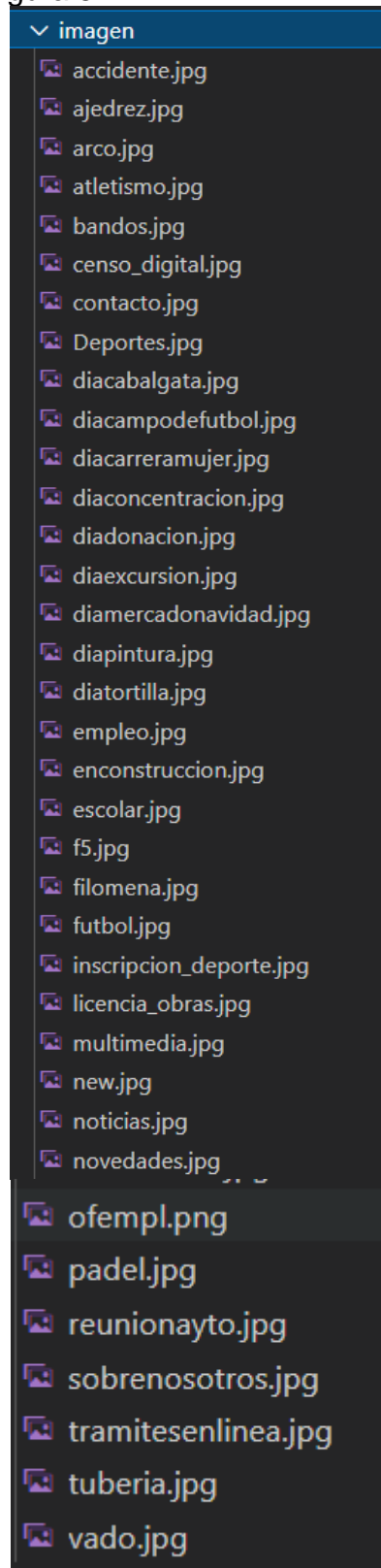


Figura 5: Captura de pantalla donde se muestran cada una de las imágenes insertados en el portal web

2. FASE DISEÑO

2.1. Fuentes portal web

Para estructurar la página web, se ha decidido escoger fuentes de Google fonts, las cuales se han nombrado en cada una de las páginas creadas para que aparecieran con tal tipología.

Las fuentes escogidas son la de Montserrat, observando en la columna de la derecha los subtipos escogidos más concretamente como se muestra a continuación en la Figura 6 [11].

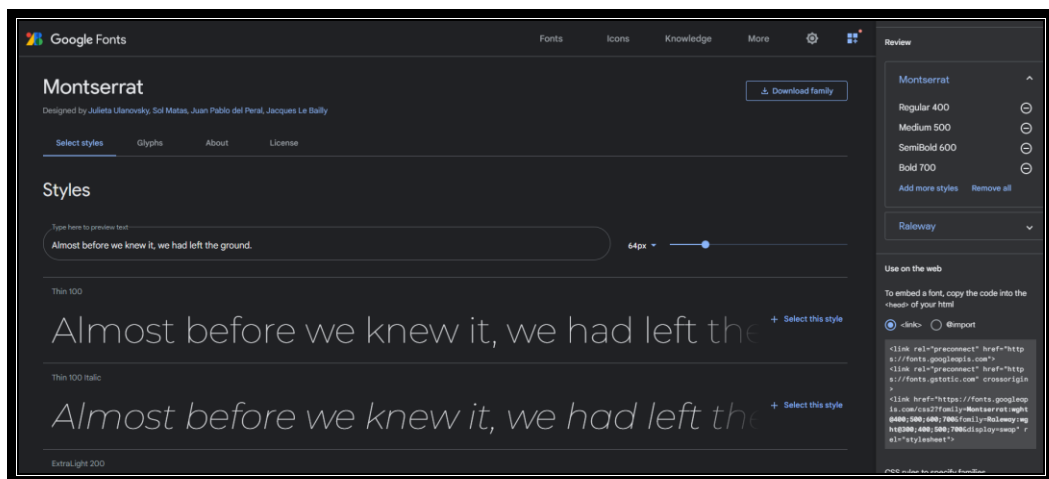


Figura 6: Captura de pantalla de la página web Google Fonts de donde se extraen los tipos de letra a utilizar

Por otro lado se ha escogido la fuente de letra Raleway, y como en el caso anterior, en la columna de la derecha se puede observar los subtipos escogidos más concretamente como se muestra en la Figura 7 [12].

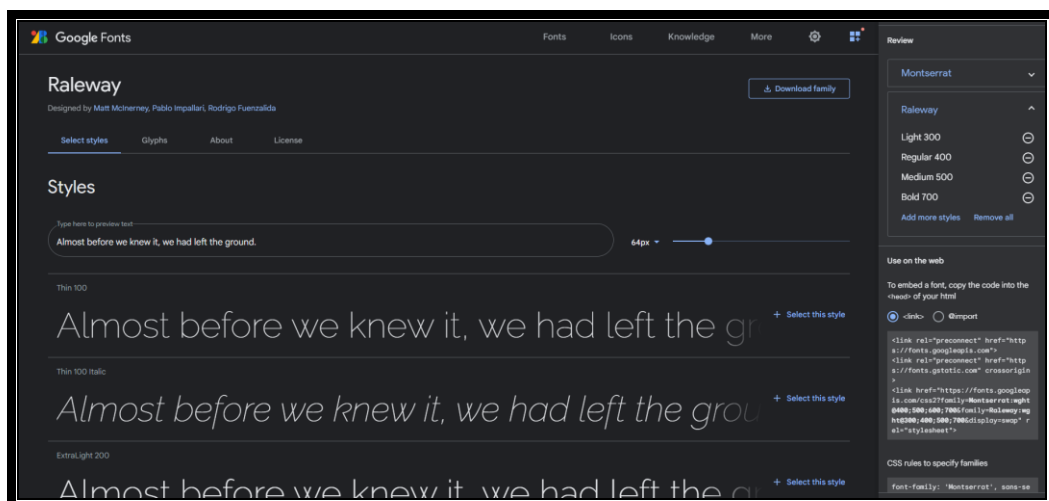


Figura 7: Captura de pantalla de la página web Google Fonts de donde se extraen los tipos de letra a utilizar

Para hacer referencia en ellos, es necesario nombrarlos en el encabezado de cada una de las páginas de las que se compone el portal web del Ayuntamiento UOC. Se programan de la manera que se aprecia en la Figura 8 donde se crea un espacio de fuentes de Google y se insertan cada uno de los enlaces:

```

<head>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<title></title>
<meta name="description" content="">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="main.css">
<!-- fuentes de google -->
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Montserrat:wght@400;500;600;700&family=Raleway:wght@300;400;500;700&display=swap" rel="stylesheet">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1Bne4KwBq781YhF1dvKuhFTAU6au08T194MhFtJ0brCF"
<!-- icons -->
<script src="https://kit.fontawesome.com/5bb9e37ec7.js" crossorigin="anonymous"></script>
</head>

```

Figura 8: Captura de pantalla del código HTML creado para enlazar las fuentes obtenidas de Google Fonts.

2.2. Diagramas bases de datos

Se inserta en la Figura 9 el diagrama de clases – UML que se ha generado para la creación de la base de datos:

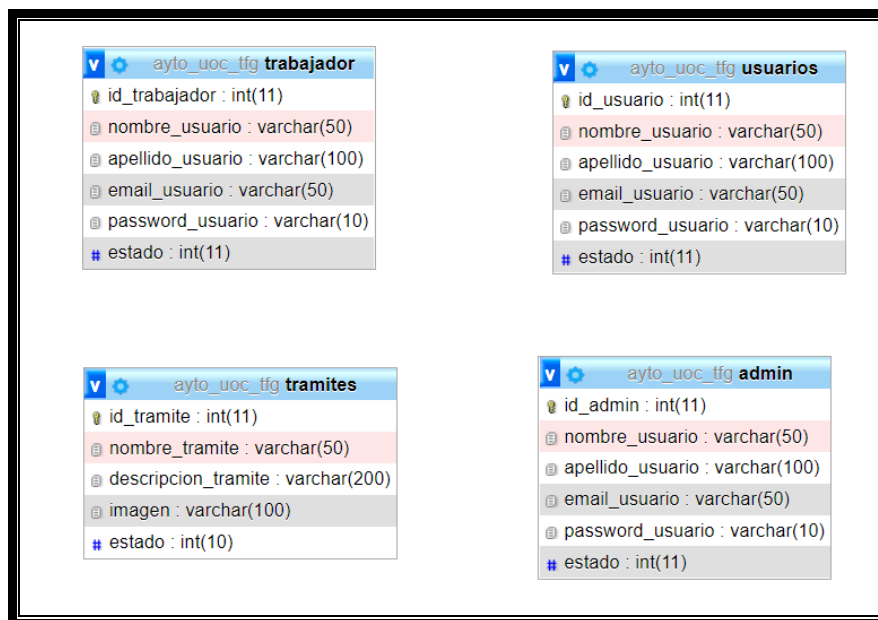


Figura 9: Diagrama de Clases – UML referente a la base de datos del Ayuntamiento UOC.

En cuanto a las relaciones de la base de datos, se indica que se ha tratado de realizar tablas independientes en cuanto a la relación de registro de usuarios con diferentes roles debido a que se quiere añadir un indicador adicional de seguridad y, en caso de intrusión en alguna de las tablas, tratar de poner un poco más complicado la detección de datos del resto de roles.

En un futuro, cuando se creen los formularios, debería relacionarse la tabla de trámites con la de los usuarios y trabajadores para poder observar el estado de los trámites en caso de que acceda un

trabajador y por otro lado para poder consultar trámites realizados por parte del ciudadano.

Visto de otra manera, según las actividades de cada uno de los actores, se plasma a continuación en la Figura 10 un diagrama de entidad – relación con cada una de las entidades que figuran en la base de datos. Se subraya las claves primarias de cada una de las tablas.

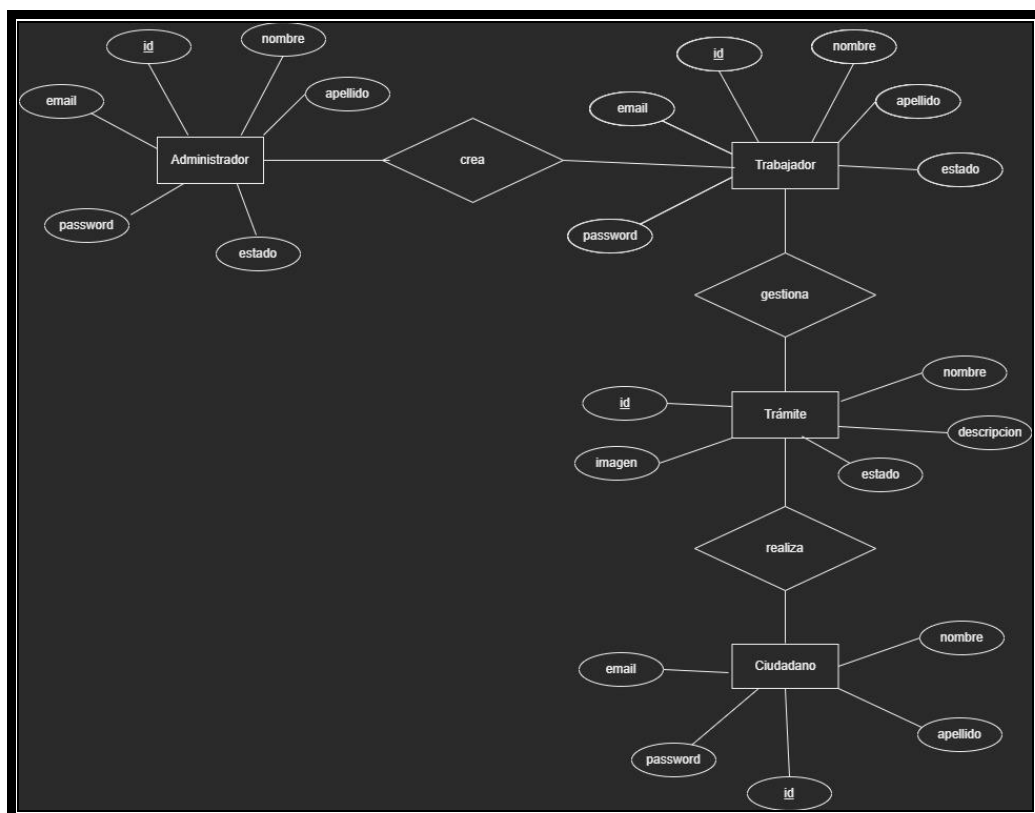


Figura 10: Diagrama Entidad – Relación referenciado a la base de datos del Ayuntamiento UOC.

2.3. Captura de requisitos

Para este apartado se han tomado como actores tres personas que corresponderían al lado ciudadano, al lado trabajador y al lado administrador.

En el lado ciudadano, el usuario podrá acceder a la página web y navegar en ella con total libertad, si bien, existe una área privada en la que es necesario registrarse para poder ofrecer al cliente datos personalizados. Para ello, es necesario registrarse y, si el mismo no se encuentra registrado, se ofrece la posibilidad de crear una cuenta. En ellas se almacena un correo electrónico y una contraseña. Gracias a ello, una vez que accedan a los trámites que desean solicitar, podrán realizar un seguimiento personalizado de los mismos.

Por otro lado, en el lado trabajador se puede navegar también con total libertad pero para poder acceder a la gestión de los trámites pendientes es necesario loguearse. Para ello es necesario tener una

cuenta de correo electrónico y una contraseña. En este caso no se ofrece la posibilidad de crear una cuenta nueva si no está registrado, si no que esa opción deberá realizarla el administrador para obtener así una seguridad óptima.

Los administradores tendrán la opción de administrar a los trabajadores a través de PhpMyAdmin para mayor garantía de seguridad.

Todo estos requisitos se tratan de expresar en el gráfico representado en la Figura 11 donde se añade todas las funcionalidades que puede hacer cada uno de los actores indicados al principio de este apartado:

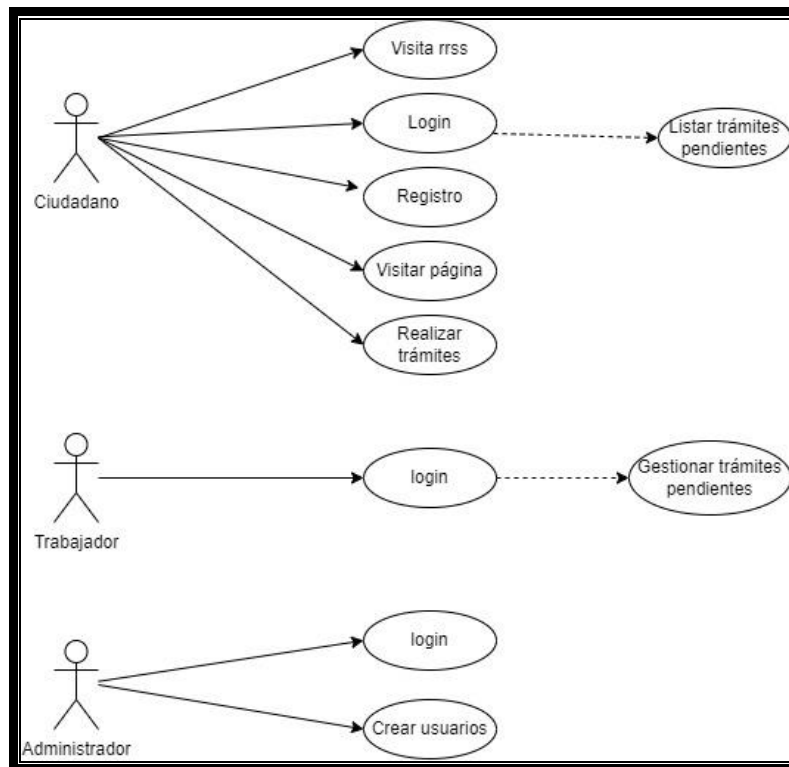


Figura 11: Captura de requisitos que afectan al portal web del Ayuntamiento UOC.

2.4. Arquitectura del sistema portal web

Para la arquitectura del portal web se ha considerado el portal web como elemento principal y las bases de datos y los usuarios, ya sean del rol que sean, como actores principales. Por tanto la arquitectura del sistema queda expresado en la Figura 12 de la siguiente manera:

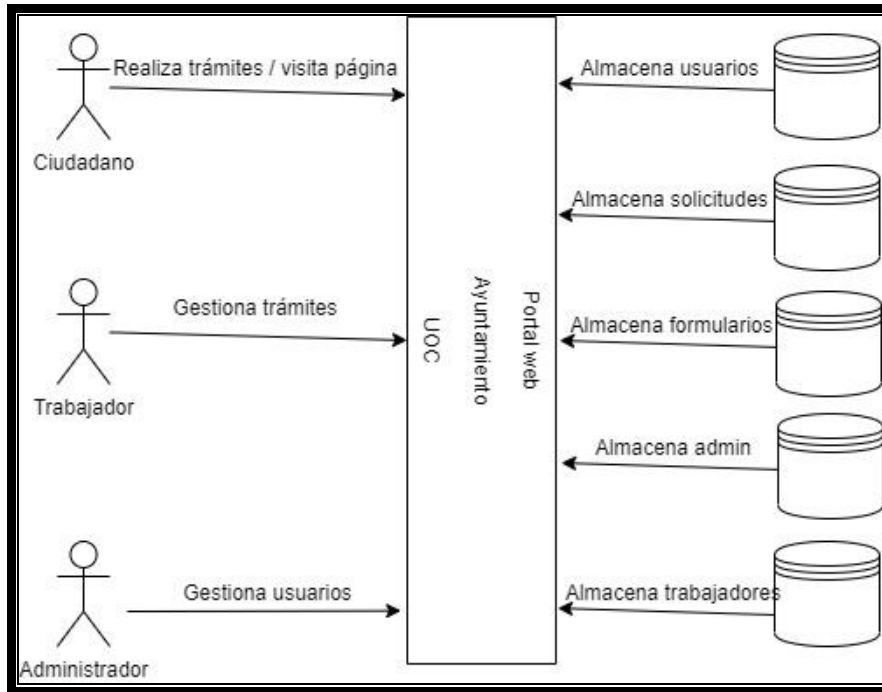
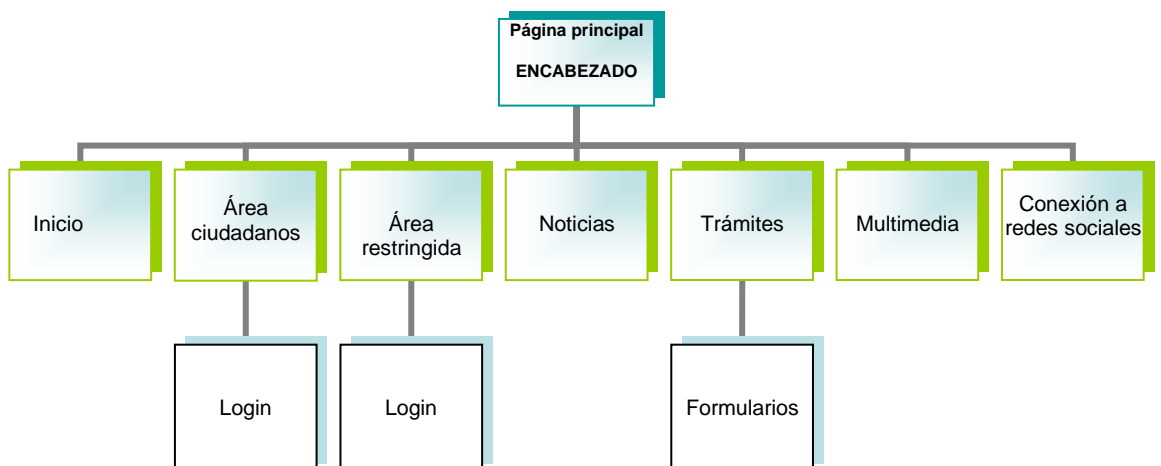
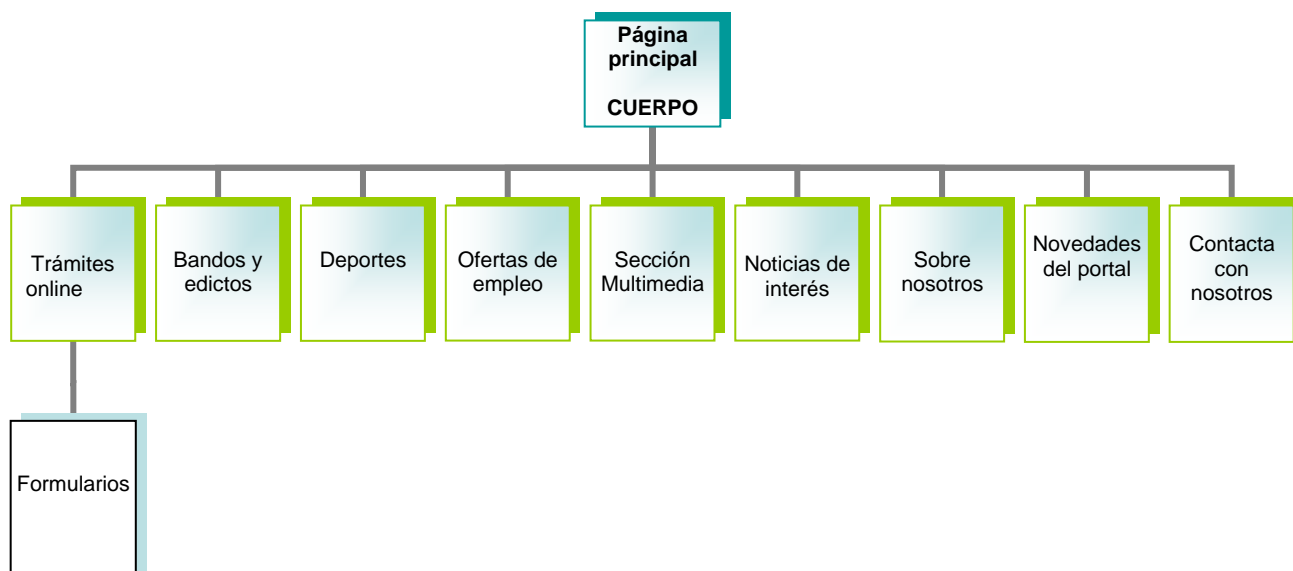


Figura 12: Arquitectura del sistema portal web del Ayuntamiento UOC.

2.5. Estructura portal principal web

Se ha decidido seguir esta estructura para el sitio web del Ayuntamiento donde se realizando dos divisiones. La primera de ellas se hace referencia a todas las secciones que se van a ofrecer de manera permanente en el encabezado y, la segunda de ellas se refiere al cuerpo de la página web donde existen secciones dinámicas como se puede ver a continuación:





3. FASE DESARROLLO

3.1. Fase desarrollo base de datos

Primeramente, ha sido necesario la instalación del programa Xampp. Dicho software es compatible con los distintos sistemas operativos existentes hoy en día, en mi caso, se ha utilizado la versión de Windows, concretamente la versión de 64 bits, 7.3.30, ya que según asignaturas anteriores se recomendaba esta versión por la estabilidad que ofrecía para el desarrollo de páginas webs.

Una vez instalada, se abre el puerto 82 para el alojamiento de phpMyAdmin, donde se crearán las bases de datos necesarias para la página web del Ayuntamiento como se muestra en la Figura 13.

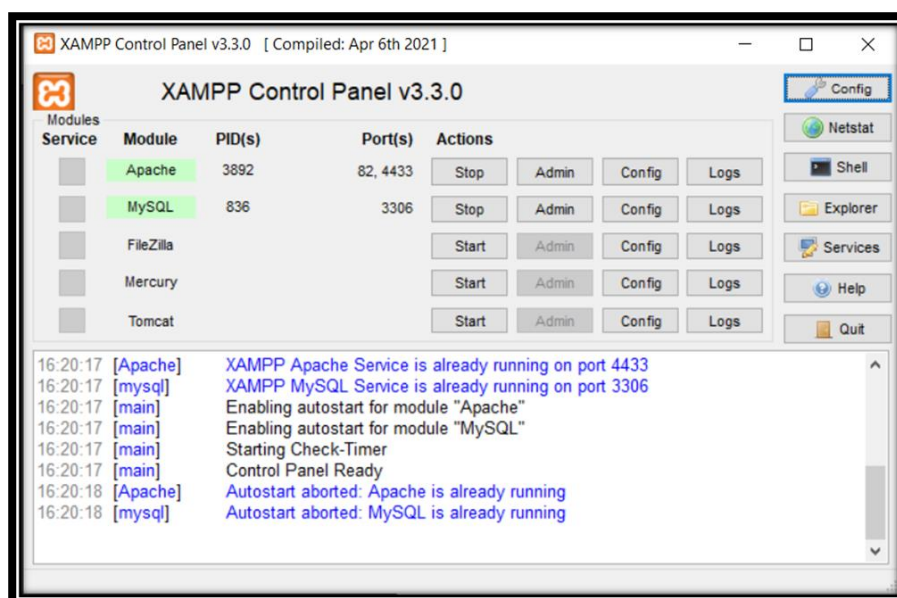


Figura 13: Interfaz de Xampp donde se puede observar los módulos instalador y funcionando además de los puertos configurados para acceder a dichos módulos.

Para el desarrollo de la base de datos en phpMyAdmin, se crea una con el nombre de “ayto_uoc_tfg”, donde, en principio, se van a crear tres tablas para usuarios y otra para formularios. Las de usuarios se clasificarán como ciudadanos, trabajadores o desarrolladores. Dependiendo del rol de acceso que tengan, podrán realizar unas funciones u otras.

Por otro lado, la cuarta tabla donde se puedan almacenar los formularios o peticiones que se generen dentro de la página. Estos formularios se rellenarán por parte del ciudadano y se deberán observar por parte del trabajador.

Para la creación de las tablas de registro de usuarios se ha desarrollado la siguiente consulta SQL en phpMyAdmin como se muestra en las Figuras 14, 15 y 16:

Tabla usuarios:

```
CREATE TABLE USUARIOS( id_usuario int not null AUTO_INCREMENT,
                        nombre_usuario varchar(50), apellido_usuario varchar(100),
                        email_usuario varchar(50) not null,
                        password_usuario varchar(10) not null,
                        estado int not null,
                        CONSTRAINT pk_usuario PRIMARY KEY (id_usuario)
);|
```

Figura 14: Consulta SQL para crear la tabla USUARIOS.

Tabla administradores

```
CREATE TABLE ADMIN( id_admin int not null AUTO_INCREMENT,
                     nombre_usuario varchar(50),
                     apellido_usuario varchar(100),
                     email_usuario varchar(50) not null,
                     password_usuario varchar(10) not null,
                     estado int not null,
                     CONSTRAINT pk_admin PRIMARY KEY (id_admin));|
```

Figura 15: Consulta SQL para crear la tabla ADMINISTRADOR.

Tabla trabajadores:

```
CREATE TABLE TRABAJADOR( id_trabajador int not null AUTO_INCREMENT,
                           nombre_usuario varchar(50),
                           apellido_usuario varchar(100),
                           email_usuario varchar(50) not null,
                           password_usuario varchar(10) not null,
                           estado int not null,
                           CONSTRAINT pk_trabajador PRIMARY KEY (id_trabajador));|
```

Figura 16: Consulta SQL para crear la tabla TRABAJADOR.

OBSERVACIONES: Según ha avanzado el proyecto, se ha detectado un fallo a la hora de crear la tabla de usuarios administradores ya que al programar en php está detectando que es un comando predeterminado por lo que se ha optado por modificar el nombre de la tabla de ADMIN a ADMINISTRADOR.

Por otro lado, se realiza el desarrollo de la inserción de usuarios a través de un código SQL. Se crearán tres roles para la presente práctica, un rol ciudadano, un rol trabajador y un rol administrador. Las consultas que hay que realizar para crear estos roles se muestran en las Figuras 17, 18 y 19.

Creación de rol ciudadano:

```
1 INSERT INTO USUARIOS (nombre_usuario,apellido_usuario,email_usuario,password_usuario,estado) VALUES ('Ana','Armenteros','aarmenteros@uoc.edu','123456',1);
```

Figura 17: Consulta SQL para crear el usuario en la tabla USUARIOS.

Creación de rol trabajador:

```
1 INSERT INTO TRABAJADOR (nombre_usuario,apellido_usuario,email_usuario,password_usuario,estado) VALUES ('Carlos','Trabajador','trabajador@uoc.edu','123456',1);
```

Figura 18: Consulta SQL para crear el usuario en la tabla TRABAJADOR.

Creación del rol administrador:

```
1 INSERT INTO ADMIN (nombre_usuario,apellido_usuario,email_usuario,password_usuario,estado) VALUES ('Jose','Administrador','admin@uoc.edu','123456',1);
```

Figura 19: Consulta SQL para crear el usuario en la tabla ADMINISTRADOR.

Una vez que se haya creado la base de datos, es necesario conectarla a la página web que vamos a desarrollar. Para ello, codificaremos en PHP la citada conexión como se muestra en la Figura 20:

```
≡ config.ini
1  dns='mysql:host=localhost;dbname=ayto_uoc_tfg'
2  usuario=root
3  clave=''
```

Figura 20: Programación HTML para acceder conectar la base de datos con la página web.

Ahora ocurre un problema, y es que, si se accede a la url del ayuntamiento, concretamente a la parte de "config.ini" se puede observar lo que se ha escrito para acceder, abriendo una brecha de seguridad importante, ya que salen las credenciales a la vista como se puede apreciar en la Figura 21 [13]:

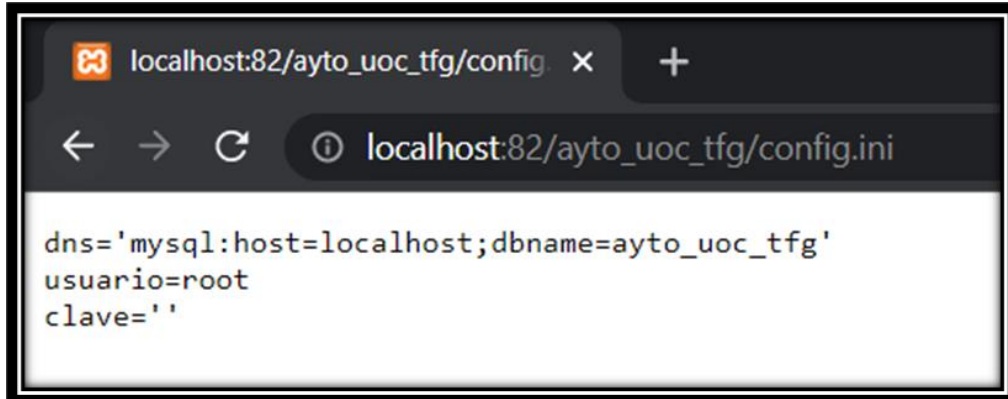


Figura 21: Captura de pantalla donde se puede observar la brecha de seguridad al insertar la url con la terminación "config.ini"

Para agregar un poco de seguridad, se restringirá el acceso a los datos de acceso de la base de datos, para ello se creará un archivo denominado .htaccess [14] donde se desarrollará el código que se muestra en la Figura 22:



Figura 22: Código HTML para evitar la brecha de seguridad mostrada en la Figura 20.

El resultado que arroja la página es el que se puede observar en la Figura 23 una vez programado el código anterior:

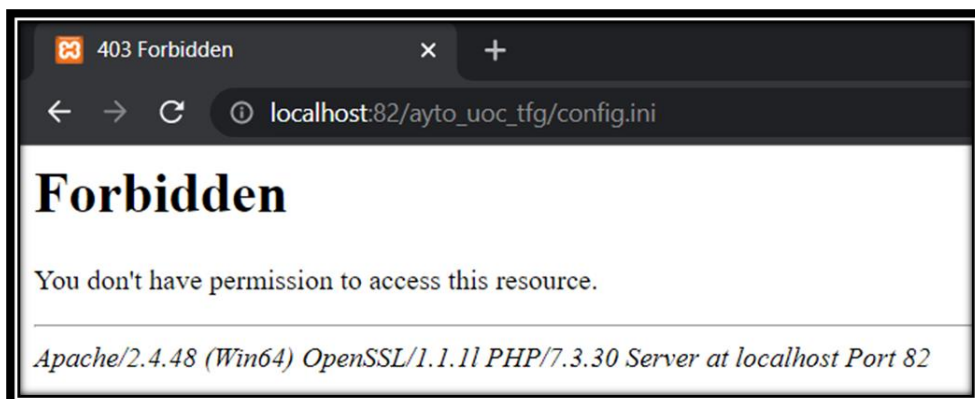


Figura 23: Resultado al aplicar el código que se ha visualizado en la Figura 21.

3.2. Fase desarrollo web

Aunque se ha hecho una breve mención en el apéndice del contenido de la página web, se quiere mostrar a continuación el árbol de ficheros creados para una situación más clara para el lector. En dichas páginas se indicará también una breve descripción de la función que ejercen dentro del portal web:

- `aboutus.php`: indica los datos de contacto del Ayuntamiento UOC así como su localización.
- `altausers.php`: lugar para dar de alta a usuarios trabajadores.
- `arearest.php`: lugar donde se dirige al trabajador registrado.
- `bandos.php`: lugar donde se publican los bandos del Ayuntamiento.
- `ciudadano.php`: espacio dedicado la información del ciudadano una vez logueado.
- **`conexión_bbdd.php`**: configuración para conectar el portal con la base de datos.
- **`config.ini`**: archivo donde se indican la información de conexión de la base de datos.
- `contacto.php`: formulario para contactar con el Ayuntamiento UOC.
- `deportes.php`: espacio dedicado a la información deportiva
- `empleo.php`: espacio dedicado a las ofertas de trabajo del Ayuntamiento.
- `get_all_tramites.php`: página donde se pueden conseguir todos los trámites pendientes.
- **`index.php`**: página principal del portal del Ayuntamiento UOC.
- `listatramites_ciudadano.php`: página donde se pueden encontrar el listado de trámites de un ciudadano una vez logueado.
- `listratramites_trabajador.php`: página donde se pueden encontrar el listado de trámites de un trabajador una vez logueado.
- `login_admin.php`: lugar de logueo para el administrador.
- `login_ciudadano.php`: lugar de logueo para el ciudadano.
- `login_trabajador.php`: lugar de logueo para el trabajador.

- logout.php: desconexión de usuario.
- **main.css**: hoja de estilos, donde se da formato a la página web.
- multimedia.php: espacio donde se publica contenido multimedia.
- news.php: página donde se publican las noticias de interés.
- novedades.php: lugar donde se publican las novedades de la página web.
- registro.php: configuración para registrarse como usuario.
- tramites.php: espacio donde se ofrecen los trámites que se pueden tramitar de manera online.

3.2.1. Estructura básica portal Ayuntamiento UOC

La estructura básica sobre el portal principal de la página web principal, tiene los elementos que se muestran en la Figura 24:

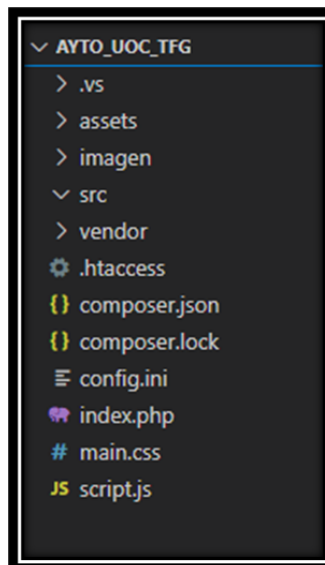


Figura 24: Estructura básica de la portada principal de página web principal del Ayuntamiento UOC.

Donde en index.php se ha desarrollado la página principal. Desde esta ubicación se redirigirá a cada una de las opciones o funciones que se podrán realizar en la citada página del ayuntamiento.

La página principal, ha quedado desarrollada de la manera que se ve en la Figura 25:



Figura 25: Interfaz principal de la web Ayuntamiento UOC.

Como funciones extras, se han creado varias opciones, que se programarán para una página web más completa. Estas funciones adicionales son las de ofertas de empleo, contacta con nosotros, sobre nosotros y las ya mencionadas en el diagrama de Gantt.

Para la utilización de las fuentes, se ha recurrido a las disponibles en Google Fonts, y se ha programado el código HTML del modo que se observa en la Figura 26 para que puedan ser seleccionadas en dicho portal web:

```
index.php
1 <!DOCTYPE html>
2 <html class="no-js">
3 <head>
4 <meta charset="utf-8">
5 <meta http-equiv="X-UA-Compatible" content="IE=edge">
6 <title></title>
7 <meta name="description" content="">
8 <meta name="viewport" content="width=device-width, initial-scale=1">
9 <link rel="stylesheet" href="main.css">
10 <!-- fuentes de google -->
11 <link rel="preconnect" href="https://fonts.googleapis.com">
12 <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
13 <link href="https://fonts.googleapis.com/css2?family=Montserrat:wght@400;500;600;700&family=Raleway:wght@300;400;500;700&display=swap" rel="stylesheet">
14
```

Figura 26: Fragmento de código HTML donde se recurre a las fuentes seleccionadas de Google Fonts.

Por otro lado, la página principal se ha dividido en dos estructuras, en el encabezado y en el cuerpo. El encabezado no cambiará en ninguna de las subpáginas, por que siempre aparecerá las diferentes opciones que podrán elegir los visitantes de la página. Mientras que, el cuerpo, será la parte donde se ofrecerán las acciones principales de la función elegida.

Cabe recordar que la página web se divide en tres secciones como normal general. La primera parte es la cabecera, denominada también *header*, donde se suele incorporar el título, los logos, imágenes y un encabezado que se desea que permanezca inalterable, como

puede ser el menú. La segunda sección se trata del cuerpo o también llamado body donde se publica el contenido principal que se quiere muestra al usuario. Y para finalizar, la tercera sección es el pie o también denominado footer donde se suele poner un menú rápido, datos de contacto o datos sobre los derechos de la página.

En este caso, para el desarrollo de la cabecera de la página web, se ha realizado con la utilización de un menú, de pequeños accesos directos a redes sociales y del título de la página Ayuntamiento UOC, quedando de la manera que se aprecia en la Figura 27:

```
<header>
  <!-- título -->
  <div class="navigation-container">
    <div class = "top-head">
      <div class = "web-name">
        <h1>AYUNTAMIENTO UOC</h1>
      </div>
      <!-- opción menú -->
      <div class = "ham-btn">
        <span>
          <i class = "fas fa-bars"></i>
        </span>
      </div>
      <!-- opción cerrar -->
      <div class = "times-btn">
        <span>
          <i class = "fas fa-times"></i>
        </span>
      </div>
    </div>
    <!-- barra de navegación -->
    <div class = "nav-bar" id="nav-bar">
      <nav>
        <ul>
          <li><a href="#">Inicio</a></li>
          <li><a href="#">Área Ciudadanos</a></li>
          <li><a href="#">Área restringida</a></li>
          <li><a href="#">Noticias</a></li>
          <li><a href="#">Trámites</a></li>
          <li><a href="#">Multimedia</a></li>
        </ul>
      </nav>
    </div>
    <!-- iconos redes sociales -->
    <div class = "social-icons">
      <ul>
        <li><a href="#"><i class = "fab fa-facebook"></i></a></li>
        <li><a href="#"><i class = "fab fa-instagram"></i></a></li>
        <li><a href="#"><i class = "fab fa-youtube"></i></a></li>
        <li><a href="#"><i class = "fab fa-twitter"></i></a></li>
      </ul>
    </div>
  </div>
</header>
```

Figura 27: Fragmento de código HTML donde se desarrolla la cabecera de la página web.

Para finalizar, la estructura básica del página principal se ha expuesto un imágenes y opciones que se podrán escoger en el portal web. Para ello, se ha codificado de la forma que se plasma a continuación y que se puede ver gráficamente en la Figura 28:

- Sección – *container box*: Creación de un contenedor donde se publicarán cada una de las secciones.
- *div*: creación de cada una de las divisiones donde en su interior contarán las características de las secciones.
 - *href*: se indica en su interior la página vinculada a la que debe dirigirse.
 - *img src*: vincula a la imagen que debe aparecer.
 - *h4*: título que aparecerá en el inferior de la sección a la que nos deseamos dirigir.


```

<section class="banner">
  <div class="banner-main-content">
    <h2>BIENVENIDO A LA PÁGINA DEL AYUNTAMIENTO DE UOC</h2>
    <h3>Aquí podrás realizar los trámites de manera rápida y sencilla</h3>
  </div>
</section>

<section class="container-box">
  <div class="imagen-container"><a href="tramites.php"></a>
  <h4>Trámites Online</h4></div>
  <div class="imagen-container"> <a href="bandos.php"></a>
  <h4>Bandos y edictos</h4></div>
  <div class="imagen-container"> <a href="deportes.php"></a>
  <h4>Deportes</h4></div>
  <div class="imagen-container"> <a href="empleo.php"></a>
  <h4>Ofertas de Empleo</h4></div>
  <div class="imagen-container"> <a href="multimedia.php"></a>
  <h4>Sección multimedia</h4></div>
  <div class="imagen-container"> <a href="news.php"></a>
  <h4>Noticias de interés</h4></div>
  <div class="imagen-container"> <a href="novedades.php"></a>
  <h4>Novedades del portal</h4></div>
  <div class="imagen-container"> <a href="aboutus.php"></a>
  <h4>Sobre nosotros</h4></div>
  <div class="imagen-container"> <a href="contacto.php"></a>
  <h4>Contacta con nosotros</h4></div>
</section>

```

Figura 28: Fragmento de código HTML donde se configura la cabecera de la página web.

También, aunque no es objetivo fundamental de este Trabajo, se ha tratado de darle un diseño a la página de tal manera que quede lo mejor posible, para ello, se ha creado un archivo “main.css” [15] donde se ha dado estilo a cada una de las opciones que se implementan en la página web. Se muestra un ejemplo de cómo se está diseñando en la Figura 29:

```

/*Estilos standar*/
img{
  width:100%;
}
ul{
  list-style: none;
}
a{
  text-decoration: none;
  color:#0000;
}
hr{
  width: 95%;
  margin: 0 auto;
}
/*header*/
.top-head{
  display:flex;
  justify-content: space-between;
  align-items: center;
  margin: 0 auto;
  background: #f2f2f2;
  color: #fff;
  padding:20px 30px;
}
.top-head .fas{
  font-size: 2rem;
  cursor: pointer;
}
/*Estilos js relates*/
.times-btn{
  display: none;
}
.nav-bar{
  background: #f333;
  height: 0;
  overflow: hidden;
}

```

Figura 29: Fragmento de código CSS referente al diseño de la página web.

3.2.2. Portal Ciudadano

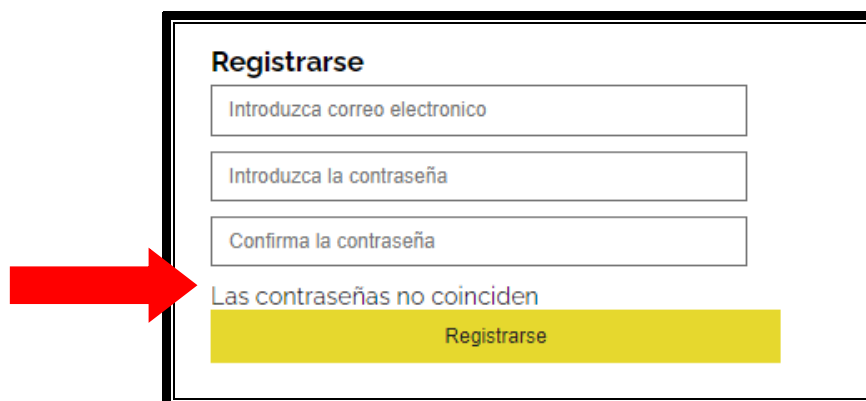
La interfaz resultante de esta sección es la que se muestra en la Figura 30 donde se ha creado la posibilidad de que un usuario pueda crearlo si no está registrado o bien, directamente se registre.



The screenshot shows the 'AYUNTAMIENTO UOC' header with navigation links for 'INICIO', 'AREA CIUDADANOS', and 'A'. Below the header, the page title is 'PÁGINA DEL AYUNTAMIENTO DE UOC: Área del ciudadano' with a subtitle 'Regístrate para acceder al área privada y comprobar el estado de tus trámites'. There are two main sections: 'Iniciar sesion' and 'Registrarse'. The 'Iniciar sesion' section has two input fields: 'Introduzca correo electronico' and 'Introduzca la contraseña', followed by a yellow 'Ingresar' button. The 'Registrarse' section has three input fields: 'Introduzca correo electronico', 'Introduzca la contraseña', and 'Confirma la contraseña', followed by a yellow 'Registrarse' button.

Figura 30: Interfaz de la sección Portal Ciudadano.

Se ha insertado un formulario, donde el ciudadano tiene la opción de iniciar sesión o registrarse. Se ha programado de tal modo, que advierte de errores al ciudadano si por ejemplo, las contraseñas no coinciden (Figura 31), el correo electrónico no coincide (Figura 32) o bien, existen fallos de conexión con la base de datos donde se encuentran almacenados los usuarios.



The screenshot shows the 'Registrarse' section of the form. It has three input fields: 'Introduzca correo electronico', 'Introduzca la contraseña', and 'Confirma la contraseña'. Below these fields, a red arrow points to a yellow error message that reads 'Las contraseñas no coinciden'. At the bottom of the form is a yellow 'Registrarse' button.

Figura 31: Mensaje de aviso al introducir contraseñas diferentes a la hora de crear el usuario.

Figura 32: Mensaje de aviso al introducir un usuario ya registrado previamente.

Para ello, en el proceso de registro se ha creado el siguiente código PHP que se ve en la Figura 33 donde destaca el hecho de que se abre una consulta switch para comprobar que no surja ningún tipo de error a la hora de crear el usuario. Es importante realizar este paso para no que luego a la hora de registrarse no existan usuarios duplicados y por tanto existan problemas de seguridad:

- ```

switch($_GET['ereg']){
 case '1':
 echo '....';
 break;
 case '2':
 echo '....';
 break;
 case '3':
 echo '....';
 break;
 default:
 break;

```

```

<form action="registro.php" method="POST">
<h3>Registrarse</h3>
<input type="text" name="email_usuario_registro" placeholder="Introduzca correo electronico">
<input type="password" name="password_usuario_registro" placeholder="Introduzca la contraseña">
<input type="password" name="password_usuario_registro_confirmacion" placeholder="Confirma la contraseña">

<?php
 if(isset($_GET['ereg'])){
 switch($_GET['ereg']){
 case '1':
 echo '<p>Error de conexion a bbdd</p>';
 break;
 case '2':
 echo '<p>Error ya está registrado</p>';
 break;
 case '3':
 echo '<p>Las contraseñas no coinciden</p>';
 break;
 default:
 break;
 }
 }
?>
<button>Registrarse</button>
</form>

```

Figura 33: Código PHP en el que se crea un SWITCH para comprobar los posibles errores a la hora de registrarse.

Por otro lado, cuando se loguea un ciudadano previamente registrado, se ha configurado de la manera que se ve en la Figura 34:

```
<form action="login_ciudadano.php" method="POST">
<h3>Iniciar sesion</h3>
<input type="text" name="email_usuario" placeholder="Introduzca correo electronico">
<input type="password" name="password_usuario" placeholder="Introduzca la contraseña">
<?php
 if(isset($_GET['e'])){
 switch($_GET['e']){
 case '1':
 echo '<p>Error de conexion a bbdd</p>';
 break;
 case '2':
 echo '<p>Error email</p>';
 break;
 case '3':
 echo '<p>contraseña incorrecta</p>';
 break;
 default:
 break;
 }
 }
?>
<button>Ingresar</button>
```

Figura 34: Código PHP donde se muestra las opciones de login.

Como punto a destacar, se ha optado por separar los logueos de ciudadanos, trabajadores y administradores. Los registros de cada uno se han introducido en la base de datos en tablas diferentes, pero a la vez, le añade un punto adicional de seguridad, puesto que, si existe una brecha de seguridad, no podrían acceder a todos y cada uno de los usuarios que existieran en las bases.

Además de lo anterior, se ha optado que, para mayor seguridad, en la zona de área restringida no exista la posibilidad de registrarse, pues se trata de usuarios de que deben estar muy controlados y deben ser gestionados por el administrador únicamente.

Para finalizar, una vez que se accede a la página aparece la siguiente portada de la Figura 35:

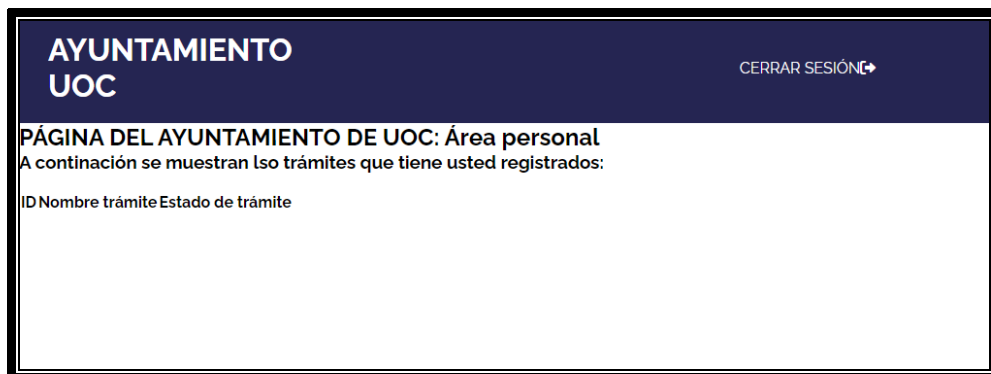


Figura 35: Interfaz que se muestra después del logueo.

Al no desarrollar los formularios y la solicitud de los mismos, no puede haber conexión a la base de datos y mostrar el cuadro con los trámites pendientes. No obstante, se desarrolla el código y se deja como

comentario que muestra las siguientes características que se ven en la Figura 36. Esta parte de código lo que hace es realizar una consulta sql que se dirige a la base de datos mediante la página “conexion\_bbdd.php” y posteriormente se arroja el resultado mediante un array:

- Añadimos la página que conecta la base de datos.

```
include('conexion_bbdd.php');
```

Incluimos la consulta deseada

```
$sql="SELECT * FROM nombre_tramite";
```

Mandamos la consulta a la base de datos

```
$result=mysqli_queri($con,$sql);
```

Arrojamos el resultado en array

```
While($row=mysqli_fetch_array($result)){
```

```
 echo
```

```
 '<tr>
```

```
 <th>ID</th>
```

```
 <th>Nombre trámite</th>
```

```
 <th>Estado de trámite</th>
```

```
 </tr>
```

Con esto se consigue ir poniendo el resultado en

columnas

```
 <td>'.row['id']. '</td>
```

```
<table class="table table-striped">
 <thead>
 <tr>
 <th>ID</th>
 <th>Nombre trámite</th>
 <th>Estado de trámite</th>
 </tr>
 </thead>
 <tbody>
 <?php
 //esta funcion estaria incluida en el momento que se pudieran gestionar trámites
 /* include('conexion_bbdd.php');
 $sql="SELECT * FROM nombre_tramite";
 $result=mysqli_queri($con,$sql);
 while($row=mysqli_fetch_array($result)){
 echo
 '<tr>
 <td>'.row['id']. '</td>
 <td>'.row['nombretramite']. '</td>
 <td>'.row["estado"]. '</td>
 </tr>';
 }
 <?php
 </tbody>
</table>
```

Figura 36: Código PHP comentado para consultar en la base de datos los trámites pendientes.

### 3.2.3. Portal Área restringida

La interfaz del Área restringida queda similar al de ciudadanos salvo que no es posible que se puedan crear usuarios. El aspecto se puede ver en la Figura 37:



Figura 37: Interfaz del Área Restringida.

El aspecto de la página ha quedado como se puede apreciar arriba. Como se ha indicado en el apartado anterior, en el área restringida no existe la posibilidad de registrarse, únicamente acceder a la zona indicada con los usuarios previamente autorizados por un administrador.

La codificación en HTML ha quedado de igual manera que la expuesta para el área ciudadano. En este caso, se va a proporcionar el código escrito para el login, ya que realiza una consulta a la base de datos como se puede apreciar a continuación en la Figura 38. Lo más destacado es lo siguiente para realizar el login:

- Conectamos con la base de datos.  
*include ('conexión\_bbdd.php')*  
 Extraemos el correo electrónico para realizar la consulta en la base de datos.  
*\$email\_usuario=\$\_POST['email\_usuario'];*  
 Generamos la consulta sql que se mandará a la tabla trabajador.  
*\$sql="SELECT \* FROM TRABAJADOR WHERE email\_usuario='\$email\_usuario';*  
 ....  
 Si es correcto, se iniciará la sesión.  
*session\_start();*

```

login_trabajador.php
1 <?php
2 include('conexion_bbdd.php');
3 $email_usuario=$_POST['email_usuario'];
4 $sql="SELECT * FROM TRABAJADOR WHERE email_usuario='$email_usuario'";
5 $result=mysqli_query($con,$sql);
6 if($result){
7 $row=mysqli_fetch_array($result);
8 $count=mysqli_num_rows($result);
9 if($count!=0){
10 $password_usuario=$_POST['password_usuario'];
11 if($row['password_usuario']!=$password_usuario){
12 //contraseña incorrecta
13 header('Location: ../login_trabajador.php?e=3');
14 }else{
15 session_start();
16 $_SESSION['id_usuario']=$row['id_usuario'];
17 $_SESSION['email_usuario']=$row['email_usuario'];
18 $_SESSION['password_usuario']=$row['password_usuario'];
19 header('Location: ../');
20 }
21 }else{
22 //Error email
23 header('Location: ../login_trabajador.php?e=2');
24 }
25 }else{
26 //Error de conexion a bbdd
27 header('Location: ../login_trabajador.php?e=1');
28 }

```

Figura 38: Código PHP para consultar en la base de datos si el usuario pertenece a los autorizados en el área restringida.

Una vez que se accede a la página queda con el siguiente aspecto. Hay que puntualizar que al no desarrollar la parte de formularios, no puede conectarse a la base de datos para saber los trámites pendientes (Ver Figura 39):

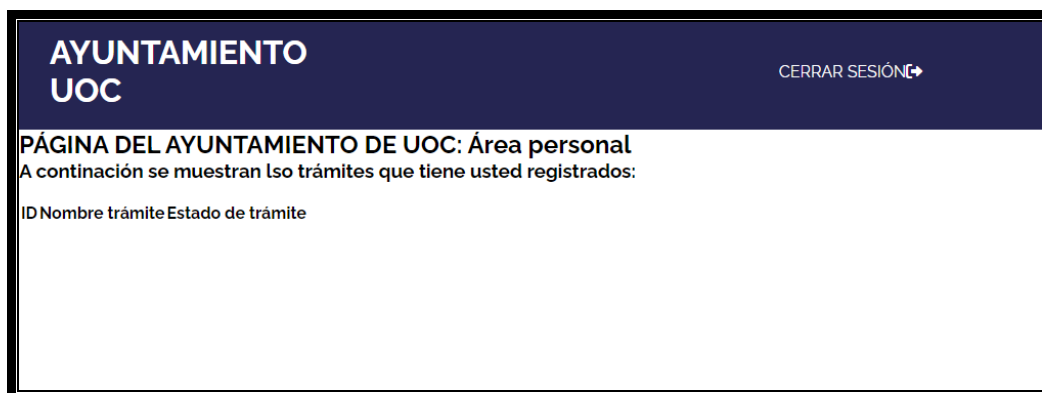


Figura 39: Aspecto que muestra la interfaz una vez logueado.

No obstante, se desarrolla el código para que se produzca la conexión aunque se deja comentada para que no aparezca error en la página web como se muestra en la Figura 40. A continuación se muestra lo más destacado de este trozo de código:

- Añadimos la página que conecta la base de datos.

*include('conexion\_bbdd.php');*

Incluimos la consulta deseada

*\$sql="SELECT \* FROM nombre\_tramite";*

Mandamos la consulta a la base de datos

*\$result=mysqli\_queri(\$con,\$sql);*

Arrojamos el resultado en array

*While(\$row=mysqli\_fetch\_array(\$result)){*

*echo*

*'<tr>*

Con esto se consigue ir poniendo el resultado en

columnas

*<td>'.row['id'].</td>*

```

<table class="table table-striped">
 <thead>
 <tr>
 <th>ID</th>
 <th>Nombre trámite</th>
 <th>Estado de trámite</th>
 </tr>
 </thead>
 <tbody>
 <tr>
 <td>'.row['id'].</td>
 <td>'.row['nombretramite'].</td>
 <td>'.row["estado"].</td>
 </tr>
 </tbody>
</table>

```

Figura 40: Código PHP comentado para consultar en la base de datos los trámites pendientes.

### 3.2.4. Portal administrador

Como se ha mencionado en la fase de planificación del proyecto esta área ha sufrido una modificación pues, se pretendía que desde la misma página web se pudieran crear usuarios por parte el administrador web pero, se ha valorado darle un toque de seguridad [16] y por este motivo se ha decidido que no se haga desde el portal web si no que se hará desde la base de datos directamente, es decir, desde PHPMyAdmin.



### 3.2.5. Trámites online

En esta página, de cara al futuro, se ha dejado la parte de inserción de formularios mediante creación de consultas SQL programándolo desde la página HTML como se puede apreciar a continuación en la Figura 41. Esta parte del código trata de generar una array en el que una vez realizada la consulta en la base de datos, mostrará todos los formularios que se encuentren alojados en dichas bases. La parte del código que se ha explicado es la siguiente:

- Se genera el array  
`$datos=[];`  
Contador a 0.  
`$i=0;`  
Consulta sql que se enviará a la base de datos.  
`$sql="select * from tramites where estado = 1";`  
Se envía la consulta a la base de datos.  
`$result=mysqli_query($con,$sql);`  
Bucle para mostrar el resultado.  
`while($row=mysqli_fetch_array($result)){`  
Se van plasmando en la página cada uno de los trámites.  
`$obj=new stdClass();`  
`$obj->id_tramite=$row['id_tramite'];`  
`$obj->nombre_tramite=$row['nombre_tramite'];`  
`$obj->descripcion_tramite=$row['descripcion_tramite'];`  
`$obj->imagen=$row['imagen'];`  
`$datos[$i]=$obj;`

```
get_all_tramites.php
1 <?php
2 include('../conexion_bbdd.php');
3 $response=new stdClass();
4
5 //$datos=array();
6 $datos=[];
7 $i=0;
8 $sql="select * from tramites where estado=1";
9 $result=mysqli_query($con,$sql);
10 while($row=mysqli_fetch_array($result)){
11 $obj=new stdClass();
12 $obj->id_producto=$row['id_tramite'];
13 $obj->nombre_producto=$row['nombre_tramite'];
14 $obj->descripcion_producto=$row['descripcion_tramite'];
15 $obj->imagen=$row['imagen'];
16 $datos[$i]=$obj;
17 $i++;
18 }
19 $response->datos=$datos;
20
21 mysqli_close($con);
22 header('Content-Type: application/json');
23 echo json_encode($response);
```

Figura 41: Código PHP donde se realiza una consulta SQL para observar los trámites que existen en la base de datos.

Se crean algunos trámites de empleo mediante consultas SQL por demostrar otro tipo de método para alimentar la base de datos se muestra en la Figura 42. Como destacado se indica lo siguiente:

- **INSERT INTO:** en sql indica que queremos crear un dato en la tabla que indiquemos seguidamente.
- **(datos):** a continuación debemos indicar los nombres de las columnas donde se quieren crear esos datos. Siempre deben seguir el orden de las columnas de las tablas para no generar errores.
- **VALUES:** donde se deberá indicar entre comillas los datos literales que deseamos que aparezcan en la insercción de datos de la tabla.

```
1 INSERT INTO tramites(nombre_tramite,descripcion_tramite,imagen)
2 VALUES('Solicitud de vado', 'Este trámite requiere estar registrado. Dirigido a las titulares de vehículo que quieran ejercer su derecho de acceder a sus
inmuebles.', 'vado.jpg'),
3 ('Solicitud de censo en el municipio', 'Dirigido a todos los ciudadanos que deseen censarse en el municipio', 'censo_digital.jpg'),
4 ('Solicitud de licencia de obras', 'Dirigido a aquellos que deseen realizar una pequeña o gran reforma en sus viviendas o locales', 'licencia_obras.jpg'),
5 ('Solicitud de inscripción en actividades deportivas', 'Abierta la temporada 2021-2022 para la inscripción en la oferta deportiva que ofrece el
municipio', 'inscripcion_deporte.jpg'),
6 ('Solicitud de inscripción en guardería', 'Abierto el plazo para la inscripción en las guarderías públicas del municipio', 'escolar.jpg');
```

Figura 42: Código SQL para realizar la alimentación de la base de datos en la tabla Trámites.

### 3.2.6. Área noticias

El aspecto de la página de noticias queda de la manera que se observa en la Figura 43:

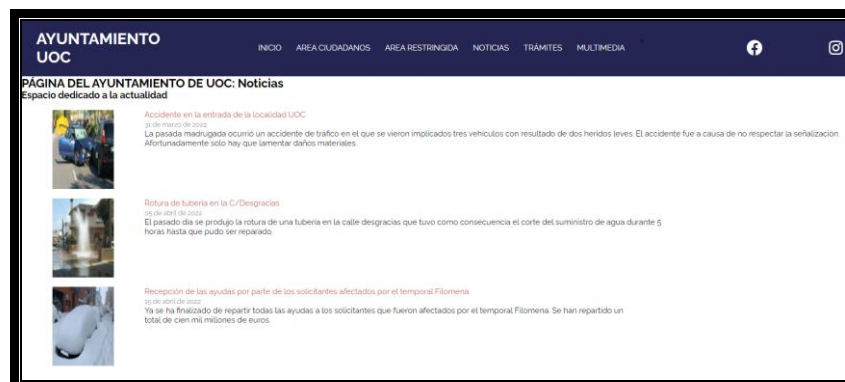


Figura 43: Interfaz de la sección de noticias.

Para que tenga el presente aspecto, se ha desarrollado un código en HTML donde se ha creado un contenedor con varios cajones (cada una de las noticias). Este código ha quedado del modo que aparece en la Figura 44 y en el que se destacan las circunstancias siguientes:

- **Sección – container box:** Creación de un contenedor donde se publicarán cada una de las noticias.

- *div*: creación de cada una de las divisiones donde en su interior contarán las características de las noticias.
  - *img src*: vincula a la imagen que debe aparecer sobre la noticia.
  - *div class*: título de la noticia.
  - *small*: fecha de la noticia
  - *descripcion*: donde se detalla la descripción completa de la noticia.

```

<section class="container-box">
 <div class="cajon-noticia">
 <div class="foto">

 </div>
 <div class="titulo">Accidente en la entrada de la localidad UOC</div>
 <small>31 de marzo de 2022</small>
 <p class="descripcion">La pasada madrugada ocurrió un accidente de tráfico en el que se vieron implicado

</p>
 </div>
 <div class="cajon-noticia">
 <div class="foto">

 </div>
 <div class="titulo">Rotura de tubería en la C/Desgracias</div>
 <small>05 de abril de 2022</small>
 <p class="descripcion">El pasado día se produjo la rotura de una tubería en la calle desgracias que tuvo

</p>
 </div>
 <div class="cajon-noticia">
 <div class="foto">

 </div>
 <div class="titulo">Recepción de las ayudas por parte de los solicitantes afectados por el temporal Filo
</div>
 <small>15 de abril de 2022</small>
 <p class="descripcion">Ya se ha finalizado de repartir todas las ayudas a los solicitantes que fueron af

</p>
 </div>
</section>

```

Figura 44: Código HTML donde se observa cómo se programan cada una de las noticias que aparecen en la sección noticias de la web Ayuntamiento UOC.

### 3.2.7. Área multimedia

Se ha tratado de toque adicional a esta sección. Pues, además de ver las fotos, se pueden seleccionar las mismas para agrandarlas y verlas con más detalle. El aspecto de la página con la visión general de las imágenes es la que se ve en la Figura 45:

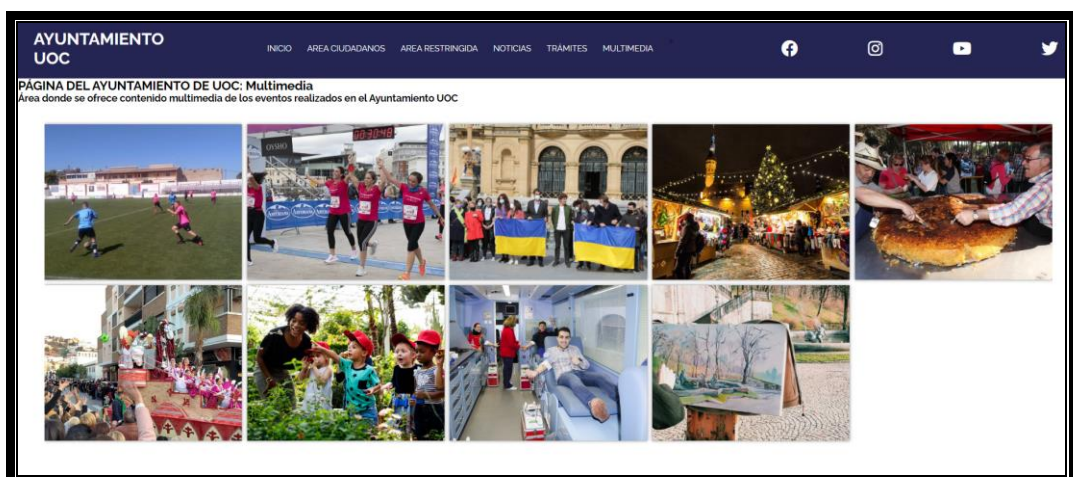


Figura 45: Interfaz de la sección multimedia.

Al pasar el ratón por cada una de las fotos, se ha tratado de darle un poco de efecto, y la misma gira y se ve borrosa. Para ello se ha programado en CSS el siguiente código que se muestra en la Figura 46 y que se explica a continuación:

- Se indica que cuando se pase el ratón haga el efecto (a:hover)  
*.container-box a:hover img{*
  - Se crea un filtro borroso.  
*filter:blur(2px);*
  - Se crea un pequeño giro de 10 grados y un escalado para generar efectos.  
*transform:rotate(10deg) scale (1.3);*

```
.container-box a:hover img{
 filter:blur(2px);
 transform:rotate(10deg) scale(1.3);
}
```

Figura 46: Código CSS para genera efecto al pasar el ratón sobre la imagen multimedia.

El código en HTML para exponer las imágenes es el siguiente de la Figura 47:

```
<section class="container-box">

</section>
```

Figura 47: Código HTML con el muestrario de fotos que aparece en la sección multimedia de la web del Ayuntamiento UOC.

Por otro lado, como se ha comentado al principio de esta área, al pulsar cada una de las imágenes, se ha desarrollado un código en HTML que permite verlas en grande con más detalle. Así mismo, permite pasar a la siguiente imagen, retroceder o cerrar la ampliación.

Una captura de pantalla de cómo queda el aspecto que se ve en la Figura 48:

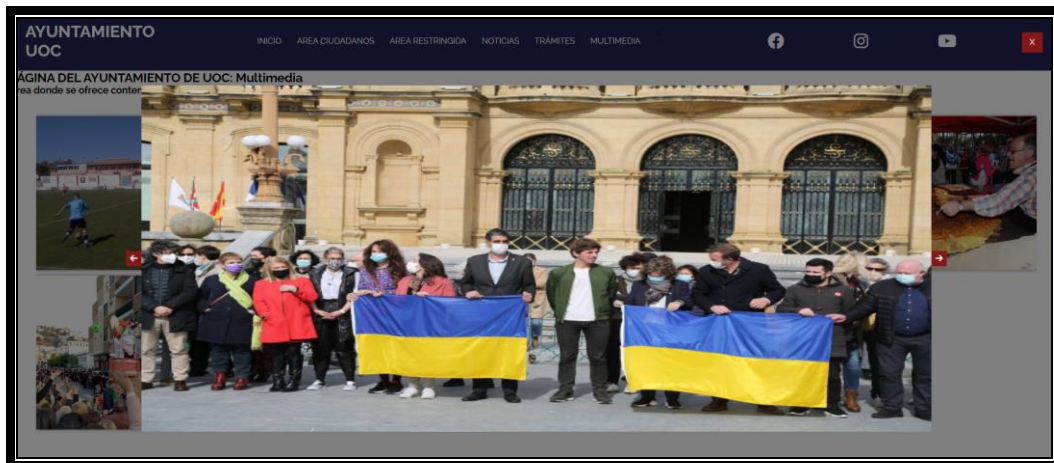


Figura 48: Interfaz al pulsar una imagen y que se agrande, permitiendo poder pasar a la siguientes o anteriores imágenes así como cerrar este modo de visualización.

Y, el contenedor que lleva a cada una de las imágenes ampliado queda programado del siguiente modo (se muestra sólo un contenedor en la Figura 49, cada una de las imágenes que haya en la vista principal debe llevar el mismo contenedor). La explicación detallada del contenedor es el siguiente:

- *Class*: se indica que tiene que ser la imagen en grande
- *Id*: indica la imagen que tiene que mostrar
- *Class siguiente imagen 1*: indica que tiene que ir a la imagen anterior si se pulsa el botón izquierda.
- *Class siguiente imagen 3*: indica que tiene que ir a la imagen posterior si se pulsa el botón derecha.

```
<article class="grande-box" id="imagen2">
 <i class="fas fa-arrow-left"></i>

 <i class="fas fa-arrow-right"></i>
 X
</article>
```

Figura 49: Código HTML del contenedor donde se programa la imagen ampliada.

### 3.2.8. Área novedades

El apartado novedades se ha desarrollado de igual modo a los anteriores. No se ha adentrado demasiado en el diseño de la sección, puesto que el presente trabajo se trata de desarrollar la página web en general y no diseñar todo en detalle. Por todo ello, se muestra una captura de pantalla (Figura 50) de la visualización de esta área:



Figura 50: Interfaz de la sección Novedades de la web del Ayuntamiento UOC.

Al igual que los códigos de las áreas anteriores, se ha programado en HTML un cajón en cuyo interior se van exponiendo cada una de las novedades que tiene la página web del Ayuntamiento UOC (Figura 51). La explicación de estos contenedores es el siguiente:

- Sección – *container box*: Creación de un contenedor donde se publicarán cada una de las novedades.
- *div*: creación de cada una de las divisiones donde en su interior contarán las características de las novedades.
  - *img src*: vincula a la imagen que debe aparecer sobre la novedad.
  - *div class*: título de la novedad.
  - *small*: fecha de la novedad

```

<section class="container-box">
 <div class="cajon-noticia">
 <div class="foto">

 </div>
 <div class="titulo">Actualización de la página: se incorpora la sección Sobre Nosotros</div>
 <small>10 de marzo de 2022</small>

 </div>
 <div class="cajon-noticia">
 <div class="foto">

 </div>
 <div class="titulo">Actualización de la página: se incorpora la sección Contacta con nosotros</div>
 <small>22 de marzo de 2022</small>

 </div>
 <div class="cajon-noticia">
 <div class="foto">

 </div>
 <div class="titulo">Actualización de la página: se incorpora la portal de Empleo</div>
 <small>15 de abril de 2022</small>

 </div>

```

Figura 51: Código HTML donde programa cada una de las novedades que luego se visualizan en la sección novedades de la web del Ayuntamiento UOC.

### 3.2.9. Contacta con nosotros

Se ha desarrollado el aspecto de la subpágina “Contacta con nosotros” donde se ha insertado un formulario que permite enviarlo a la base de datos y así poder realizar diferentes consultas. Los trabajadores podrán visualizar estos formularios rellenos para poder tramitar las dudas que se plantean desde esta sección como se ve en la Figura 52.

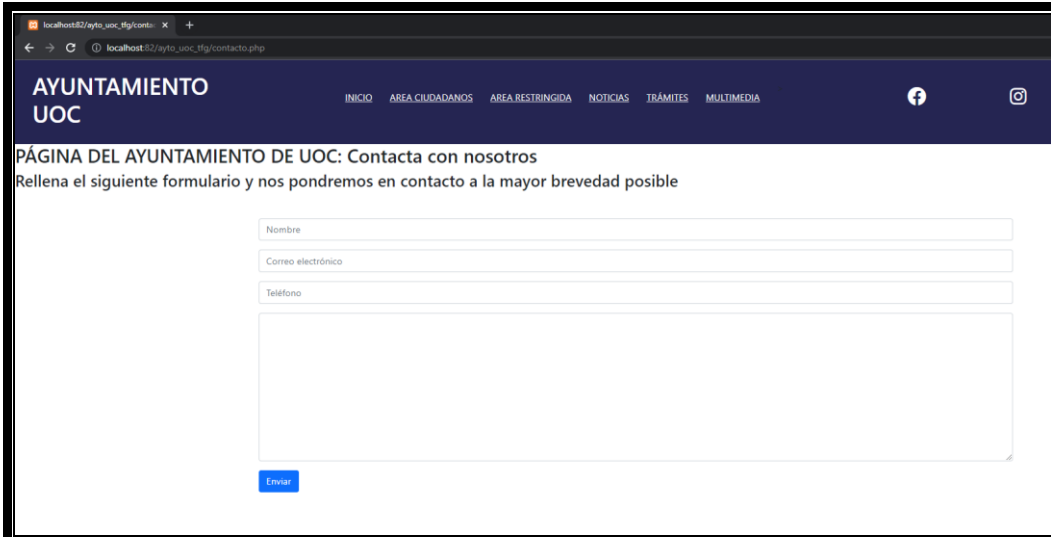


Figura 52: Interfaz de la sección Contacta con nosotros del portal web del Ayuntamiento UOC.

Para insertar el formulario se ha recurrido a la página Bootstrap [17] donde, gracias al CSS se puede hacer referencia a las configuraciones de los campos de formularios y simplemente es crear los cuadros de una manera más sencilla. Cabe destacar que para la presente memoria se ha tratado de realizar un poco de variedad a la hora de desarrollar el código y por este motivo, hay partes de código realizadas manualmente y otras que se obtiene el desarrollo de páginas como esta para así poder ofrecer alternativas y demostrar todos los conocimientos y capacidades que se han adquirido a lo largo del Grado y que son posibles utilizar ante este tipo de proyectos.

Por ello, se ha hecho referencia en el encabezado de la siguiente manera (Figura 53):

```
<html class="no-js">
 <head>
 <meta charset="utf-8">
 <meta http-equiv="X-UA-Compatible" content="IE=edge">
 <title></title>
 <meta name="description" content="">
 <meta name="viewport" content="width=device-width, initial-scale=1">
 <link rel="stylesheet" href="main.css">
 <!-- Fuentes de Google -->
 <link rel="preconnect" href="https://fonts.googleapis.com">
 <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
 <link href="https://fonts.googleapis.com/css2?family=Montserrat:wght@400;500;600;700&family=Raleway:wght@300;400;500;700&display=swap" rel="stylesheet">
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kIBq781YhF1dvKuhFTAU6au" data-bbox="188 818 265 845">
 <!-- Icons -->
 <script src="https://kit.fontawesome.com/5bb9e37ec7.js" crossorigin="anonymous"></script>
```

Figura 53: Código HTML donde se puede observar el enlace donde hace referencia a las configuraciones de los campos de formularios.







- Sección – *container box*: Creación de un contenedor donde se publicarán cada una de las oferta de empleo.
- *div*: creación de cada una de las divisiones donde en su interior contarán las características de las ofertas de empleo.
  - *img src*: vincula a la imagen que debe aparecer sobre la oferta de empleo.
  - *div class*: título de la oferta.
  - *small*: fecha de la oferta de empleo

```

<section class="container-box">
 <div class="cajon-noticia">
 <div class="foto">

 </div>

 <div class="titulo">Oferta de empleo: 1 plaza en el Banco UOC</div>
 <small>31 de marzo de 2022</small>
 <p class="descripcion">Se necesita un Ingeniero Informático para realizar labores de mantenimiento en equipos del Ban

 </div>

 <div class="cajon-noticia">
 <div class="foto">

 </div>

 <div class="titulo">Oferta de empleo: 3 plazas en el Colegio UOC</div>
 <small>05 de abril de 2022</small>
 <p class="descripcion">Se necesita 3 Ingenieros Informáticos para realizar labores de mantenimiento en equipos del Ba

 </div>

```

Figura 58: Código HTML donde se programa cada uno de los contenedores donde se insertan las noticias a publicar.

### 3.2.12. Área de deportes

El aspecto de la presente área queda de la forma que se aprecia en la Figura 59:

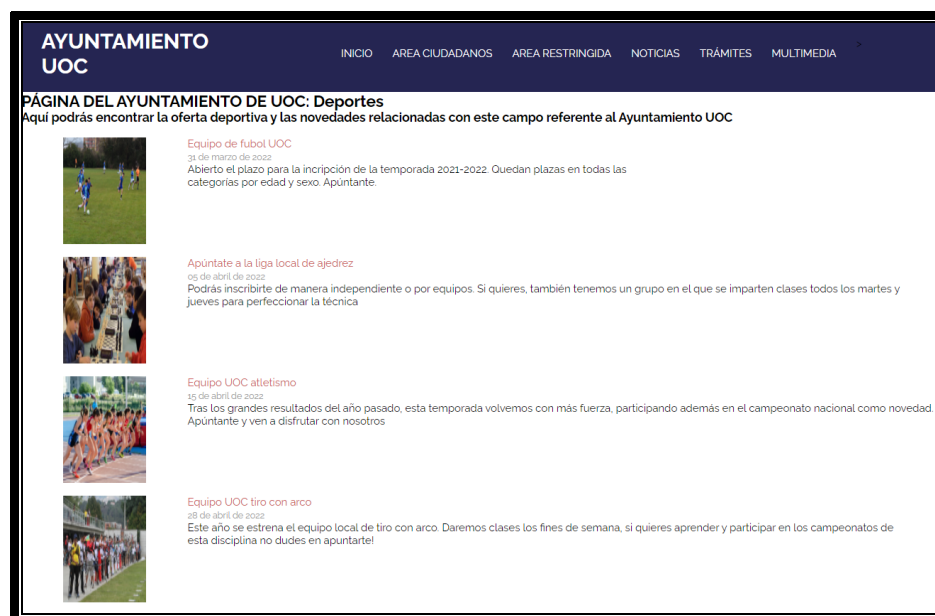


Figura 59: Interfaz de la sección deportes de la página web del Ayuntamiento UOC.

Para programarlo se ha tomado como referencia el área de noticias. Una vista rápida del código creado es la mostrada en la Figura 60 y que se explica a continuación:

- Sección – *container box*: Creación de un contenedor donde se publicarán cada una de las noticias deportivas.
- *div*: creación de cada una de las divisiones donde en su interior contarán las características de las noticias deportivas.
  - *img src*: vincula a la imagen que debe aparecer sobre la noticia deportiva.
  - *div class*: título de la noticia deportiva.
  - *small*: fecha de la noticia deportiva

```
<section class="container-box">
 <div class="cajon-noticia">
 <div class="foto">

 </div>

 <div class="titulo">Equipo de futbol UOC</div>
 <small>31 de marzo de 2022</small>
 <p class="descripcion">Abierto el plazo para la inscripción de la temporada 2021-2022. Quedan plazas

 </p>
 </div>

 <div class="cajon-noticia">
 <div class="foto">

 </div>

 <div class="titulo">Apúntate a la liga local de ajedrez</div>
 <small>05 de abril de 2022</small>
 <p class="descripcion">Podrás inscribirte de manera independiente o por equipos. Si quieres, también

 </p>
 </div>
</section>
```

Figura 60: Código HTML donde se programa los contenedores de la sección deportes.

### 3.2.13. Área bandos

El aspecto de la página queda de la manera mostrada en la Figura 61 y que se explica resumidamente a continuación:



Figura 61: Interfaz de la sección de bandos del portal web del Ayuntamiento UOC.

El código creado es el tomado como referencia para el área de noticias. A continuación (Figura 62) se inserta una captura de pantalla donde se muestra el desarrollo del citado código y se detalla lo más interesante del código:

- Sección – *container box*: Creación de un contenedor donde se publicarán cada una de los bandos.
- *div*: creación de cada una de las divisiones donde en su interior contarán las características de los bandos.
  - *img src*: vincula a la imagen que debe aparecer sobre el bando.
  - *div class*: título del bando.
  - *small*: fecha del bando.
  - *descripcion*: descripción detallada del citado bando.

```
<section class="container-box">
 <div class="cajon-noticia">
 <div class="foto">

 </div>

 <div class="titulo">Aprobación de asfaltado de la C/Suerte</div>
 <small>31 de marzo de 2022</small>
 <p class="descripcion">Se ha aprobado el asfaltado de la C/Suerte. Las mejoras se realizarán a mediados del

 </div>

 <div class="cajon-noticia">
 <div class="foto">

 </div>

 <div class="titulo">Corte de agua previsto el día 15 de abril de 2022</div>
 <small>05 de abril de 2022</small>
 <p class="descripcion">Por la necesidad de realizar mejoras en el suministro de agua del municipio, se va a

 </div>
 </div>
```

Figura 62: Código HTML donde se programa los contenedores de la sección bandos.

## 4. FASE TESTEO

Entre los problemas que se pudo encontrar al principio fue uno de los comentados más arriba en el que se indicaba que el usuario “admin” estaba dando problemas debido a que en Visual Studio, al realizar programación sobre tal usuario lo consideraba con una acción en vez de como un usuario en sí por lo que hubo que modificar el nombre de “admin” a “administrador”. Esto se pudo conseguir gracias a una consulta en sql que permitiera la modificación de tal valor.

No obstante, en la fase final de la memoria se realiza una nueva fase de testeo para observar que todo funciona correctamente, apreciando lo siguiente:

- Página principal: Funcionan todos los enlaces correctamente.
- Función de registro: se realiza correctamente. Se fuerza que aparezcan los mensajes de error en caso de repetir correo o introducir mal la contraseña observando que funcionan correctamente.
- Función de login: fase superada correctamente.
- Contacta con nosotros: se observa que el formulario aparece correctamente.
- Sobre nosotros: el mapa figura de manera correcta y la ubicación deseada también.
- Formularios online: se observa que conecta bien con la base de datos para mostrar todos los trámites que se pueden hacer online y que han sido grabados en la base de datos.
- Sección multimedia: al pasar el ratón se realizan los efectos especiales deseados. Además, al pulsar en una imagen para agrandarla y pulsar la tecla de izquierda o derecha, avanza o retrocede de manera correcta siguiendo la secuencia adecuada.
- Resto de páginas del portal: se visualizan de manera correcta.

## 5. Conclusiones

Una vez finalizado el trabajo, se observa que no es nada sencillo enfrentarse a este tipo de tareas finales. Pues, es de las asignaturas que más requieren concentración y esfuerzo para poder presentar un resultado con el que se pueda superar el Grado de Ingeniería Informática.

En mi caso, me caracterizo por ser una persona constante y que lleva las tareas bastante al día, me gusta entregar todo en el primer momento en vez de dejarlo para el último día, pero en este caso, debido a complicaciones en el trabajo, las primeras tareas fueron un poco apuradas de tiempo y quizás no pude pensar lo suficiente los objetivos propuestos en la primera entrega inicial.

Como lecciones aprendidas, se ha conseguido realizar la página web que se proponía como tarea inicial. Se ha adquirido conocimientos en cuando al desarrollo del portal, pues nunca me había enfrentado a desarrollar una página al 100% en el que se trataran diversas áreas y funciones.

Entre los objetivos, creo que he podido conseguir todos los propuestos en el Diagrama de Gantt inicial, e, incluso, se han añadido varios objetivos más, tales como, el portal de sobre nosotros, portal de empleo, portal de deportes, portal de novedades o portal de contacta con nosotros.

Por otro lado, en el diagrama de Gantt se exponía como dato adicional el apartado de pagos / TPV, el cual se indicaba que sería un apartado extra si daba tiempo a programar todo lo anterior. En este caso, no ha dado tiempo a cumplimentar ese apartado, pero se ha compensado con el resto de los apartados adicionales indicados en el párrafo anterior.

En cuanto a la planificación, se ha cumplido con los plazos establecidos en el diagrama de Gantt con exactitud. Se ha aprovechado cada una de las PECs para realizar los consejos del profesor tutor para entregar una Memoria con mejores condiciones.

Para finalizar, como líneas de futuro, se podría centrar en el diseño de la página, darle un toque más visual. Además, se podría finalizar con desarrollar los formularios con mejor calidad y que estos puedan estar conectados con la base de datos.

## 6. Glosario

**Diagrama de Gantt:** herramienta gráfica para la previsión de tareas y objetivos a entregar en un plazo establecido.

**XAMPP:** se trata de un paquete de software libre cuya función es la gestión de bases de datos de tipo MySQL, servidores Apache e realizar intérprete de lenguajes de PHP y Perl.

**MySQL:** sistema para realizar la gestión de las bases de datos relacionales.

**HTML:** lenguaje de marcado para el desarrollo de páginas web.

**PHP:** se trata de otro lenguaje, esta vez de programación, también utilizado para el desarrollo de las páginas web.

**Sede electrónica:** se hace referencia con este nombre a aquellas Administraciones Públicas que introducen las tecnologías de la información y las comunicaciones (TIC).

**Aplicación Web:** herramienta con la que los usuarios acceden a cierta información a través del acceso a un servidor web mediante la navegación de Internet.

**Aplicación móvil:** programa con acceso directo en el dispositivo móvil que permite al usuario acceder a cierta información a través del citado *smartphone*.

**Prestashop:** sistema de código abierto y gestora de contenidos que permite construir con cierta facilidad y desde el inicio tiendas online para el comercio electrónico.

**Brecha de seguridad:** también denominado agujero de seguridad o vulnerabilidad, tratándose de un fallo en el sistema de información que permite violar la seguridad de esa aplicación, sistema o dispositivo en el que se produzca.

## 7. Bibliografía

- [1] Información extraída de la página oficial del Ayuntamiento de Borox:  
- **Web:** <https://www.ayuntamientodeborox.com/>, 04 de abril de 2022
- [2] Información extraída de la página oficial del Ayuntamiento de Antequera:  
- **Web:** <https://www.antequera.es/>, 04 de abril de 2022
- [3] Idea basada en un artículo de una revista digital:  
- **Artículo de revista:** Mariana Pérez, Web mobile o App, ¿cuál es la mejor opción para tu empresa?, 01 de abril de 2019.  
- **Web:** <https://www.marketinet.com/blog/web-mobile-app#gref>, 04 de abril de 2022
- [4] Herramienta XAMPP, programa oficial utilizado:  
- **Web:** <https://www.apachefriends.org/es/index.html>, 01.03.2022
- [5] Herramienta phpMyAdmin, programa oficial utilizado:  
- **Web:** <https://www.phpmyadmin.net/>, 01.03.2022
- [6] Lenguaje HTML para el desarrollo:  
- **Web:** <https://developer.mozilla.org/es/docs/Web/HTML>, 01.03.2022
- [7] Lenguaje PHP para el desarrollo:  
- **Web:** <https://www.php.net/>, 01.03.2022
- [8] Herramienta para el desarrollo de la página Visual Studio Code:  
- **Web:** <https://code.visualstudio.com/>, 01.03.2022
- [9] Lenguaje CSS para el desarrollo:  
- **Web:** <https://developer.mozilla.org/es/docs/Web/CSS>, 01.03.2022
- [10] Lenguaje SQL para el desarrollo de bases de datos:  
- **Web:** <https://www.w3schools.com/sql/>, 01.03.2022
- [11] Idea basada en las fuentes de Google – Google Fonts Montserrat:  
- **Web:** <https://fonts.google.com/specimen/Montserrat>, 15 de marzo de 2022
- [12] Idea basada en las fuentes de Google – Google Fonts Raleway:  
- **Web:** <https://fonts.google.com/specimen/Raleway>, 15 de marzo de 2022



[13] Información consultada en la documentación sobre el funcionamiento del archivo config.ini proporcionada en la página oficial de PHP:

- **Documentación:** PHP, el fichero de configuración (Configuración en tiempo de ejecución), 04 de abril.

- **Web:** <https://www.php.net/manual/es/configuration.file.php>, 04 de abril de 2022.

[14] Información del funcionamiento de .htaccess consultada en un artículo de una página web sobre proveedor de hosting web:

- **Artículo de la página:** Gustavo B, Qué es el archivo .htaccess: Guía completa, 25 de marzo de 2022.

- **Web:** <https://www.hostinger.es/tutoriales/que-es-el-archivo-htaccess>, 04 de abril de 2022.

[15] Información sobre el funcionamiento de la página web consulta en la página de desarrollador de Mozilla:

- **Artículo de la página:** Mozilla, Cómo está estructurado CSS.

-**Web:**

[https://developer.mozilla.org/es/docs/Learn/CSS/First\\_steps/How\\_CSS\\_is\\_structured](https://developer.mozilla.org/es/docs/Learn/CSS/First_steps/How_CSS_is_structured), 04 de abril de 2022

[16] Información consultada sobre los peligros de la utilización de la cuenta administrador en una revista digital:

- **Artículo de revista:** Yolagny Díaz Bermúdez, ¡Alerta! Peligro al utilizar la cuenta de Administrador para todo, 16 de octubre de 2014

- **Web:** <https://revista.jovenclub.cu/el-peligro-de-utilizar-la-cuenta-de-administrador/>, 04 de abril de 2022

[17] Página para utilizar frameworks de CSS de manera más sencilla:

- **Web:** <https://getbootstrap.com/>, 04 de abril de 2022.

[18] Instrucciones para obtener la inserción de Google Maps en la propia web de Google Maps:

- **Web:** <https://support.google.com/maps/answer/144361>, 04 de abril de 2022.