



Proyecto Final de FP

Desarrollo de una Aplicación Web para un Mercado Municipal

El Mercado

Autora:

Cintia García Ruiz

Curso:

2º Desarrollo de Aplicaciones Web

ÍNDICE

INTRODUCCIÓN	4
1.1. Objetivos	4
1.2. Requisitos	4
1.3. Elección de la aplicación	6
DISEÑO SOLUCIÓN	7
2.1 Metodología y Entorno de Trabajo	7
2.2. Lenguajes Softwares y Tecnologías	8
2.2.1 Backend	8
2.2.2 Front-end	8
2.3. BBDD	9
2.3.1 Modelo E/R	9
2.3.2 Tablas y columnas	10
2.4. Despliegue	13
2.4.1 Despliegue Local	13
2.4.2 Despliegue AWS	13
DESARROLLO E IMPLEMENTACIÓN	15
3.1 Módulo General	15
3.1.1 Ventana Inicial	15
3.1.2 Registro	16
3.1.3 Login	18
3.1.4 Visualización Puesto (usuario no logueado)	19
3.1.5 Visualización Producto (usuario no logueado)	19
3.2 Módulo Cliente	20
3.2.1 Histórico de Pedidos	20
3.2.2 Realización de Pedidos	21
3.2.3 Puestos Favoritos	21
3.2.4 Modificación Datos Cliente	22
3.3 Vendedor	22
3.3.1 Creación de un puesto	23
3.3.2 Gestión de Puestos	24
3.3.3 Gestión de Pedidos	24
3.3.4 Ventana de Ayuda al Usuario	25
3.4 Administrador	26
3.4.1 Gestión de Productos	26
3.4.2 Gestión de Puestos	27
3.4.3 Gestión de Usuarios	27
DESPLIEGUE	28

4.1. Despliegue Local	28
4.2. Despliegue AWS	30
CONCLUSIONES	32
BIBLIOGRAFÍA	33

1. INTRODUCCIÓN

1.1. Objetivos

El objetivo de este documento es describir la memoria del proyecto final del curso.... En este proyecto, se propone una aplicación para un mercado online que pondrá en contacto tanto a los comerciantes como a los usuarios del mercado.

Los consumidores podrán encontrar sus puestos de confianza (frutería, pescadería, carnicería,...) y realizar la compra de forma online, sin necesidad de desplazarse físicamente al mercado. Por otra parte, los responsables de los puestos, dispondrán de una nueva alternativa para vender sus productos o incluso llegar a un nuevo tipo de público al que no pueden vender sus productos actualmente.

Por tanto, el objetivo del proyecto es diseñar y desarrollar una aplicación que cumpla la funcionalidad anteriormente indicada tomando como referencia los conocimientos adquiridos durante el curso en las distintas asignaturas y cumpliendo los requisitos establecidos en cada una de ellas.

1.2. Requisitos

Los requisitos que deben cumplir la aplicación son los siguientes.

- **Desarrollo**
 - Patrón de diseño MVC.
 - Tecnologías o software específico
 - Javascript, ES6 y JQuery.
 - CSS3.
 - HTML5.
 - Angular, Vue o React.
 - INKSCAPE o similar.
 - Si se utiliza PHP podrá optarse por Laravel , usar frameworks como Vue.js , y para la comunicación asíncronas en background Axios.
 - Funcionalidades:
 - Validación en formulario de cliente.

- Paginación Ajax (incremental or Page numbers).
 - Autocompletar búsqueda
 - CRUD AJAX y JQuery UI.
 - Eliminar fila con fade out.
 - Añadir registro
 - Galería de imágenes.
 - Banner.
 - El diseño de la interfaz deberá ser “Responsive” utilizando Media Queries, FlexBox y Grid layout.
 - Utilizar el preprocesador SASS para estructurar los archivos css en un único main.css con @import a los demás css (colores, cabecera, pie, cuerpo...).
 - Insertar un elemento multimedia de cada tipo: video, sonido, canvas y SVG.
 - Se podrán utilizar las clases Chartjs y Snap.svg.
 - Se debe ofrecer un servicio a los usuarios registrados, para que puedan realizar peticiones con la API para obtener información JSON.
- **Base de Datos**
 - Uso de BBDD relacionales como MySQL, MariaDB, PostgreSQL...
 - Debe tener 4 tablas como mínimo y entregar el modelo E/R al profesor.
- **Despliegue de la aplicación y generación de código.**
 - Despliegue en local de la aplicación usando Servidor Apache o Servidor de Aplicaciones.
 - Despliegue en un hosting externo.
 - Documentación del código fuente en inglés.
 - Generar de forma automática la Documentación Técnica del mismo en HTML y PDF (se deben de emplear herramientas del tipo JavaDoc, PhpDocumentor, JsDoc, o similar).
 - Uso de un Sistema de Control de Versiones. Recomendado GitHub
 - Acreditar un seguimiento de versiones. Ramas.
 - Acreditar un despliegue automatizado, si es posible.
 - Acreditar uso de SSH y/o FTP en el desarrollo y/o despliegue, si es posible.

1.3. Elección de la aplicación

Se ha decidido desarrollar esta aplicación tras observar la necesidad de acercar el mercado tradicional a todas las personas que a día de hoy realizan sus compras por internet, y además intentar darle un giro a la idea del mercado como tal para actualizarlo. La idea surgió porque los sábados es el único día que yo puedo ir hacer la compra allí, ya que son establecimientos que no abren por la tarde y entonces las personas que están trabajando tienen el acceso más reducido. Esto hace que por lo general, sea un tipo de establecimiento muy alejado de los jóvenes, que suelen ser llevar a cabo sus compras en las grandes superficies o por internet. Por tanto, mi objetivo es intentar acercar el mercado a todos los usuarios.

Además en esta situación con el COVID-19, se ha visto reflejado como las ventas online de productos frescos se han incrementado considerablemente, y donde algunos mercados municipales no han podido ofrecer este servicio siendo necesario desplazarse a ellos para poder adquirir sus productos.

2. DISEÑO SOLUCIÓN

2.1 Metodología y Entorno de Trabajo

El diseño de la aplicación se basa en MVC (**m**odelo, **v**ista, **c**ontrolador) que es un patrón de diseño software que separa la aplicación en tres capas, en donde cada capa maneja un aspecto de la aplicación.

Los modelos acceden a la base de datos y recuperan la información, la modifica o la elimina y mediante el controlador se transforma esa información para devolvérsela a la vista.

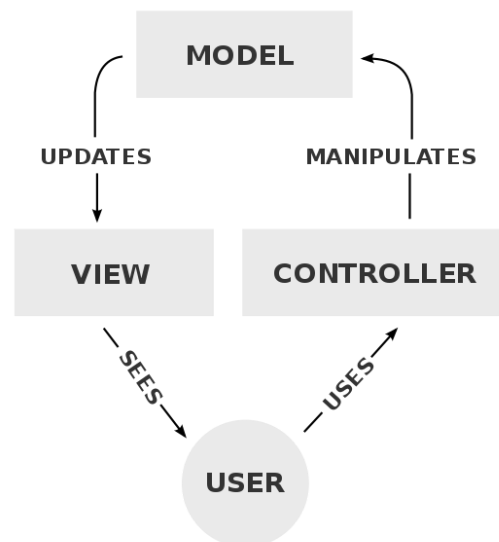


Figura 1. Modelo Vista Controlador [[Wikipedia](#)]

Para ayudarme a implementar esta metodología, he utilizado el framework de **Laravel** que implementa este tipo de arquitectura y además, se adapta bien con las necesidades de la aplicación a desarrollar y proporciona más sencillez de programación en **PHP**.

Para llevar un control de las versiones y gestión del código se ha empleado **GitHub** debido, principalmente, a su sencillez y la ventaja de poder compartir código fácilmente. Sobre la metodología empleada para el uso de esta plataforma, he empezado una rama master donde estaban las partes del proyecto cuyo desarrollo estaba finalizada. Por otro lado, he creado una rama llamada InicioMigraciones donde tenía el código sobre el que estaba trabajando y desde la que efectuaba un *merge* tras finalizar una funcionalidad o solventar un error.

El entorno de desarrollo empleado es **Visual Studio Code**, dado que proporciona extensiones relacionadas con el código como Vue.js Extension Pack, Vue VSCode Snippets, Visual Studio IntelliCode, VS Code JavaScript (ES6) snippets...

2.2. Lenguajes Softwares y Tecnologías

La aplicación se ha dividido en dos partes diferenciadas, una para el back y otra para el front donde las tecnologías usadas son distintas.

2.2.1 Backend

Es la parte del desarrollo web que se encarga de que toda la lógica de una página web funcione, un conjunto de acciones que pasan en una web pero que no vemos, como por ejemplo, la comunicación con el servidor o los acceso a BBDD.

Para esta parte, se han empleado principalmente **PHP** con **Laravel**. Con este lenguaje, se ha desarrollado para toda la lógica de los controladores de la aplicación y los modelos que se utilizan.

2.2.2 Front-end

Es una especialidad para el desarrollo web, que trabaja la interfaz web y hace que el usuario pueda interactuar con nuestra web. Está orientado a lenguaje de marcas y al lenguaje de programación web de ejecución en equipos clientes, sin necesidad de uso de servidores externos. Casi todo lo que ves en la pantalla cuando accedes a una web es desarrollo frontend.

Los lenguajes empleados para esta parte son:

- **JavaScript**: ha permitido desarrollar funcionalidades tales como verificar Email en el registro, comprobar si existe un DNI, autocompletar buscador...
- **Axios**: para hacer peticiones a la api, conecta el back con el front.
- **HTML 5**: en todas las vistas para la visualización de mi web y todos sus contenidos.
- **Bootstrap Y CSS3**: para todos los estilos de la web para ello he utilizado sass para unificar los estilos.
- **VUE**: como componentes dentro de Laravel, para cosas como mostrar la contraseña...

2.3. BBDD

La base de datos elegida es **MySQL**, ya que era necesario emplear una base de datos relacionar con las condiciones de la aplicación y se integra fácilmente con el framework de Laravel, permitiendo un mejor manejo de la conexión con los modelos. Como gestor para la base de datos en local utilizo **phpMyAdmin**.

2.3.1 Modelo E/R

Antes de crear toda la lógica de BBDD sobre MySQL, se ha realizado el diseño del **Modelo Entidad/Relación** (figura 2) a partir del cual se han obtenido 6 tablas en la base de datos y sus atributos correspondientes.

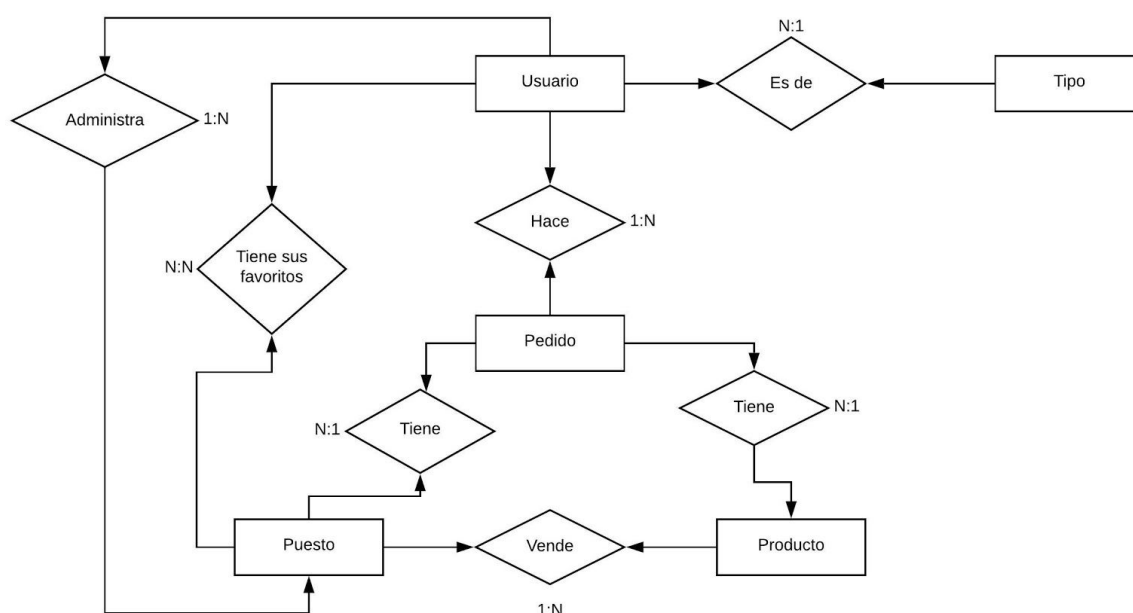


Figura 2. Modelo entidad relación del proyecto.

- **User:** Representa a las personas que se conectan a la aplicación, el objetivo y visión de cada usuario dependerá del rol de cada usuario.
- **Tipo:** Es la diferenciación de los usuarios según su rol: Cliente, Vendedor y Administrador.
- **Pedido:** Petición de un producto realizada por un usuario Cliente a un puesto, que es administrado por un Vendedor de un puesto.

- **Puesto**: Representación de un puesto virtual, donde el proveedor de producto los ofrece y el comprador puede adquirirlos.
- **ProductoPuesto**: Son los alimentos que se venden en el mercado.
- **Favorito**: Representa la relación de los puestos favoritos de los usuarios. Resultado de la relación de mucho a muchos entre usuario y puesto.

2.3.2 Tablas y columnas

La base de datos elegida es MySQL, ya que era necesario emplear una base de datos relacionar con las condiciones de la aplicación y se integra fácilmente con el framework de Laravel, permitiendo un mejor manejo de la conexión con los modelos. Como gestor para la base de datos en local utilizo phpMyAdmin. A continuación, se describen todas las columnas de las distintas tablas.

User

Esta tabla contiene a los usuarios de la aplicación.

Nombre	Tipo	Descripción
idUsu(pk)	Integer	Identificador y clave privada del usuario.
nombre	Varchar	Nombre del usuario.
apellido	Varchar	Apellido de la persona a la que pertenece el usuario.
dni	Varchar	DNI de la persona a la que corresponde el usuario.
email	Varchar	Correo electrónico.
password	Varchar	Contraseña del usuario.
telefono	Integer	Telefono usuario.
fecNac	Date	Fecha de Nacimiento.
idTipo (fk)	Integer	Identificador del tipo de usuario.
vendedor	Boolean	Indica si el usuario es o no vendedor.
direccion	Varchar	Dirección del usuario.
tarjeta	Big Integer	Tarjeta de crédito usuario.
caducidad	Varchar	Fecha caducidad tarjeta.

cvc	Integer	CVC de la tarjeta.
foto	Varchar	Foto usuario.
nif	Varchar	NIF empresa. Sólo necesario para usuarios de tipo vendedor.

Tabla 1. User.

Favorito

Se utiliza para registrar los puestos favoritos de cada usuario.

Nombre	Tipo	Descripción
idUsu(fk) (pk)	Integer	Identificador del usuario.
idPuesto (fk) (pk)	Integer	Identificador del puesto.

Tabla 2. Favorito.

Puesto

Esta tabla se emplea para registrar al puesto y sus propiedades.

Nombre	Tipo	Descripción
idPuesto (pk)	Integer	Identificador y clave privada del puesto.
idUsu (fk)	Integer	Id del usuario.
foto	Varchar	Foto del puesto.
telefono	Varchar	Teléfono del puesto.
info	Varchar	Información sobre el puesto.

Tabla 3. Puesto.

ProductoPuesto

El objetivo de esta tabla es relacionar cada producto con su puesto.

Nombre	Tipo	Descripción
idproPues (pk)	Integer	Identificador y clave privada del producto.
idPuesto (fk)	Integer	Identificador del puesto.
nombre	Varchar	Nombre del producto.
descripcion	Varchar	Información acerca del producto.
precio	Double	Precio o coste del producto.
stock	Boolean	Indica si queda producto disponible.

Tabla 4. ProductoPuesto

Pedido

Registro de pedidos.

Nombre	Tipo	Descripción
idPedido (pk)	Integer	Identificador y clave privada del pedido.
idUsu (fk)	Integer	Identificador del usuario que hace el pedido.
cantidad	Integer	Cantidad que solicita el cliente.
peso	Integer	Peso que puede tener.
fecha	Date	Fecha de realización del pedido.
pagado	Boolean	Si el pedido está pagado.
enviado	Boolean	Si el pedido está enviado.
total	Double	Precio total del pedido.

Tabla 5. Pedido.

Tipo

Tipo o roles de usuario.

Nombre	Tipo	Descripción
idTipo (pk)	Integer	Identificador y clave privada del tipo de usuario.
nombre	Varchar	Nombre del rol del usuario.

Tabla 6. Tipo.

2.4. Despliegue

Por requisitos del proyectos, se van a efectuar dos tipos de despliegue uno en local y otro en la plataforma cloud de **Amazon Web Service (AWS)**.

2.4.1 Despliegue Local

Para el despliegue de la aplicación en local se ha usado un **servidor Apache**, usando para el despliegue los conocimientos adquiridos durante el curso y la realización de distintas prácticas, el sistema operativo en el que he desplegado el servidor ha sido un **Windows Server 2016**.

Apache es un servidor web **HTTP** de código abierto y multiplataforma que implementa el protocolo HTTP y que a partir de la ejecución de la aplicación se encarga de responder a las peticiones web de los usuarios, sirviendo a estos los recursos y ficheros correspondientes según la lógica de la aplicación web.

2.4.2 Despliegue AWS

Para el despliegue de la aplicación en internet que permita el acceso a los usuarios de forma pública, se ha optado por hacer uso de un servicio cloud, principalmente por estos motivos:

- Facilidad de despliegue del código y rapidez para instanciar otros servicios necesarios como la propia Base de Datos.
- Además, nos permite obtener una independencia del hardware y sistema operativo, porque no es necesario mantenerlo y por tanto nos permite centrarnos en el desarrollo de la aplicación que es el objetivo.
- Crecimiento en el uso e importancia de este tipo de servicios en los últimos años.

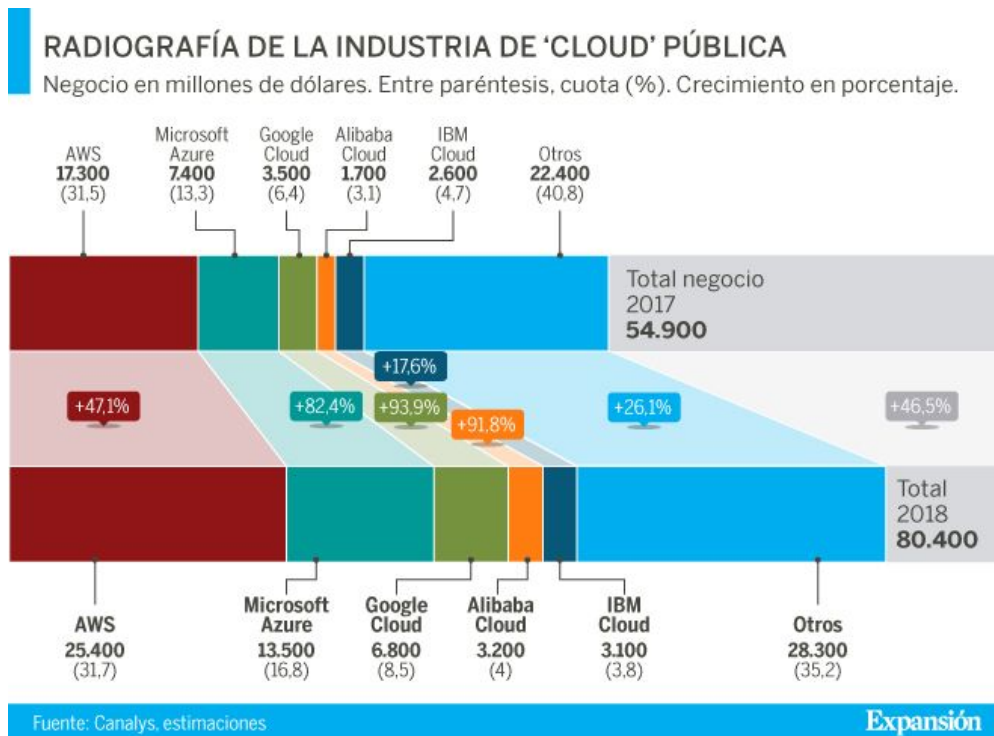


Figura 3. Comparativa año 2018 de Servicios Cloud[*Expansión*]

Debido a las facilidades ofrecidas a estudiantes para este tipo de proyectos, a que es la empresa más conocida del mercado he considerado apropiado hacer uso de AWS (Amazon Web Services). Además, este proveedor es compatible con las tecnologías empleadas (PHP 7.2, MySQL, Javascript, Laravel,...).

3. DESARROLLO E IMPLEMENTACIÓN

El desarrollo de la aplicación se ha dividido en cuatro partes o módulos por funcionalidades.

- **Módulo General.** Es la primera parte global y es el módulo por el cual el usuario accede a la aplicación gestionando el login de los usuarios. Recoge todas las funcionalidades comunes a cualquier tipo de usuario.
- **Módulo Cliente.** El segundo módulo, está destinado a los compradores y está formado por las ventanas para la visualización y selección de productos. También, la parte para la realización de pedidos.
- **Módulo Vendedor.** Cubre la necesidad de que los responsables de los puestos puedan publicar sus productos y gestionar los pedidos realizados.
- **Módulo de Administración.** Destinado a los administradores de la aplicación.

3.1 Módulo General

Aquí se encuentran las vistas comunes a todos los que se conectan a la web, pueden visualizar los puestos del mercado y sus productos, los usuarios podrán hacer uso de las funcionalidades comunes a todo los tipos de usuario, tales como comprar o añadir un puesto como favorito. También pueden acceder al registro de usuarios, para poder hacer compras o solicitar ser vendedor. En este módulo, también se incluye el control de login de los usuarios.

3.1.1 Ventana Inicial

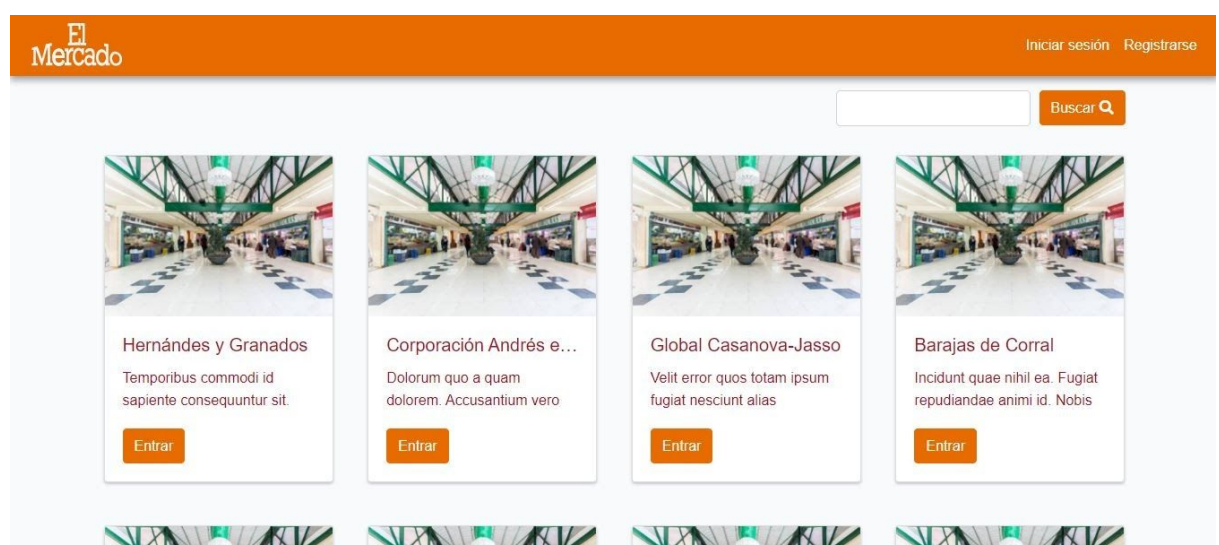


Figura 4. Ventana inicio.

La pantalla de inicio es la vista que es común a todos los usuarios estén logeados o no (figura 4). Aquí se encuentra el buscador, que está compuesto por un componente Vue y una petición get a la **API con Axios**. Dicho buscador, nos devuelve las coincidencias que existan en la Base de Datos permitiendo al usuario la selección el puesto deseado tras la búsqueda.

3.1.2 Registro

El usuario también dispone de la opción de registro o alta en la plataforma.

Figura 5. Pantalla Registro.

En la pantalla de registro (Figura 5) se validan los campos con la librería Validator que implementa Laravel, pero además con un componente Vue, se añade el campo email para comprobar que el dicho email no existe en la base de datos. Por otro lado, el campo DNI el dispone de dos tipos de comprobaciones, una para ver si ya está registrado y el otro para validar que se cumple con el formato genérico del DNI.

Si el usuario quiere ser vendedor tiene que seleccionar la opción de solicita ser vendedor y rellenar el campo del NIF relacionado de la empresa, para que el admin pueda darle permisos.

Por último, en el campo de fecha de nacimiento también se comprueba que la persona que se loguee sea mayor de edad.

A continuación, se muestra el código empleado para implementar estas funcionalidades.

```
<template>
  <div class="form-group col-md-6">
    <label for="dni" >DNI</label>
    <input id="dni" type="text" class="form-control "
name="dni" placeholder="00000000G" value="" required
@keyup="dniCheck" />
```




```

        <strong v-if="this.respuesta == 'incorrecto'">El
dni no es correcto.</strong>
        <strong v-if="this.respuesta == 'existe'">El dni ya
existe.</strong>
    </div>
</template>
<script>
    const axios = require('axios').default ;
    export default {
        data() {
            return{
                respuesta:'',
                dni:'',
            }
        },
        methods:{
            dniCheck : function(){
                this.dni = $('#dni').val();
                axios.get('api/dni/'+
this.dni).then((response) => {this.respuesta =
response.data;});
            }
        },
        mounted() {console.log('Component mounted great.')}
    }
</script>

```

3.1.3 Login

Tras darse de alta en la plataforma, el usuario podrá loguearse a través de la siguiente ventana.



The image shows a login form titled "Iniciar Sesión". At the top center is a logo consisting of a stylized white letter 'M' on a background of three overlapping vertical bars in orange, red, and dark red. Below the logo, there are two input fields. The first is labeled "Email" and contains the text "mata.victoria@hotmail.co". The second is labeled "Contraseña" and contains the text "12345cya". To the left of the password input field is an eye icon, indicating a toggle for password visibility. Below these fields is a checkbox labeled "Recordar". At the bottom of the form is a large orange button with the text "Entrar".

Figura 6. Pantalla login.

El login también está compuesto con **Vue** para añadir la funcionalidad del ojo para poder visualizar la contraseña, garantizando así la privacidad del usuario.

3.1.4 Visualización Puesto (usuario no logueado)

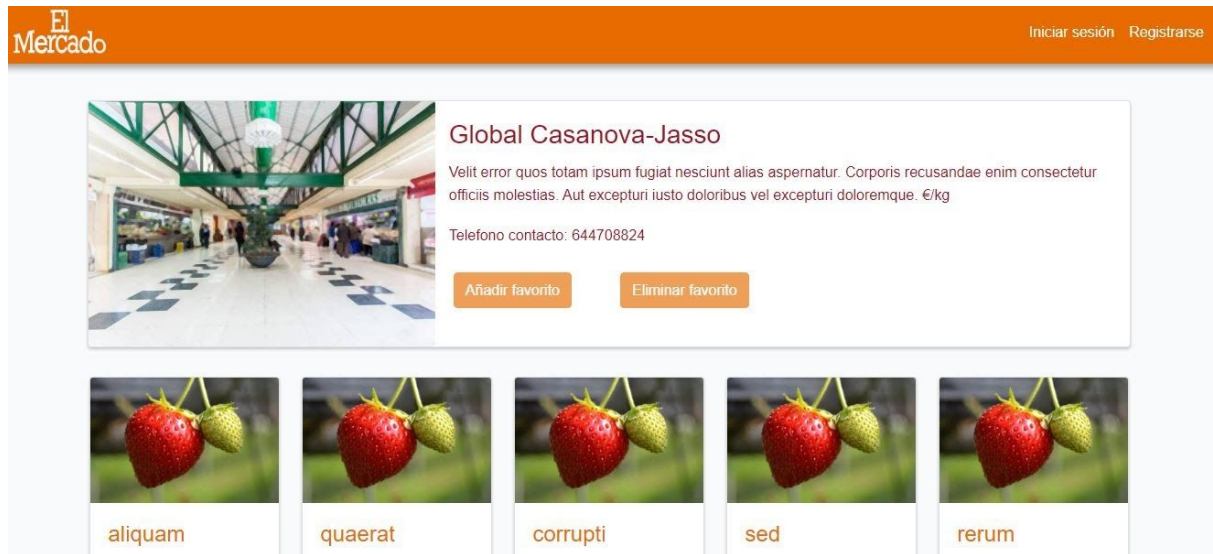


Figura 7. Pantalla puesto, sin loguearse.

En la ventana del puesto, si no eres un cliente logueado, los botones con los que se puede añadir o eliminar como favorito un puesto aparecen desactivados, ya que un puesto favorito debe de estar relacionado con un usuario. Esta misma situación, se dá cuando se intenta comprar un producto, el botón de añadir a la cesta se desactiva (figura 8).

3.1.5 Visualización Producto (usuario no logueado)

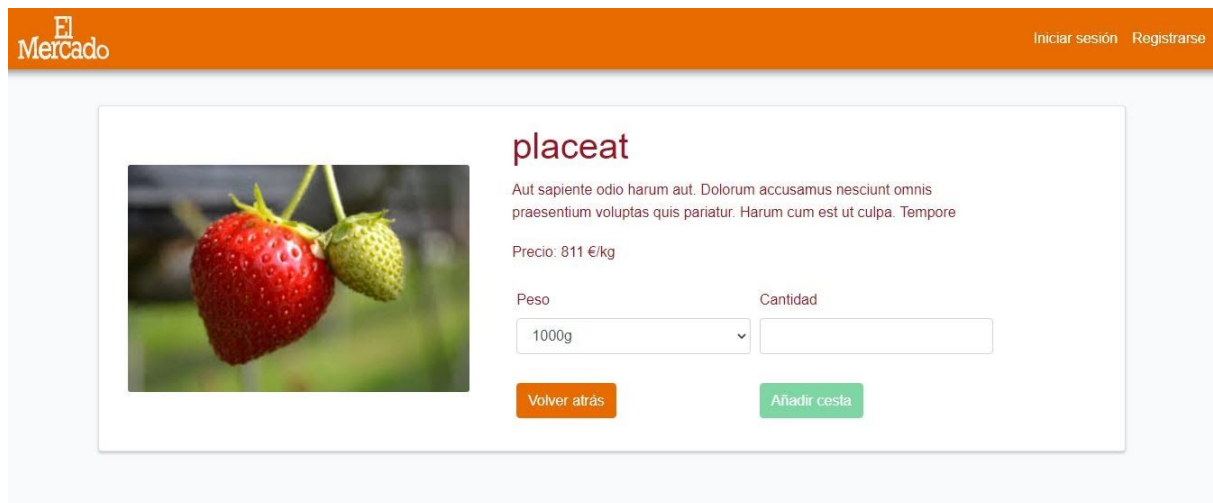


Figura 8. Producto sin loguearse

En la pantalla de producto, se ha añadido un seleccionable para que el usuario pueda poner las cantidades y el peso del producto que requiere, ya internamente el controlador hace la operación para calcular el total y realizar el pedido.

```

$total    = $cantidad * $peso;
$precio    = $producto->precio /1000;
$totalPedido = $total * $precio ;

```

3.2 Módulo Cliente

Una vez el usuario se registra como cliente, este tiene acceso a otras vistas diferentes en las que ya puede realizar compras, ver el estado de tus pedidos o ver los productos que tienes en la cesta de la compra. Además, este tipo de usuario también puede añadir los puestos a favoritos para tener un acceso más rápido a estos y en otra ventana tiene la posibilidad de modificar sus datos de registro.

3.2.1 Histórico de Pedidos



	Nombre	Puesto	Cantidad	Peso	Fecha	Enviado	Pagado
	totam	Valdivia, Pagan y Porras e Hijos	2	250 g	12/06/2020	No	0.5 €
	totam	Valdivia, Pagan y Porras e Hijos	4	250 g	12/06/2020	No	1 €
	doloremque	Ureña-Mireles	1	500 g	12/06/2020	Si	4.5 €

Figura 9. Pantalla Mis pedidos del cliente

En la pestaña de Mis Pedidos se ha añadido una tabla un listado con los pedidos llevados a cabo por el cliente. Dispone de los siguientes datos para cada pedido:

- **Nombre**: Nombre del pedido
- **Puesto**: Puesto virtual del mercado al que se ha realizado el pedido.
- **Cantidad**: Cantidad del producto adquirida a través del pedido.
- **Fecha**: Fecha de realización del pedido.
- **Enviado**: Indica si el proveedor ha realizado el envío del producto o no.
- **Pagado**: Coste del pedido.

3.2.2 Realización de Pedidos



Figura 10. Pantalla mis pedidos del cliente.

En el icono de la **cesta del navbar**, se indica el número de productos que el usuario tiene seleccionados pendiente de confirmar mediante el pago.

Al entrar en la cesta, se presentan los datos de los productos seleccionados en una tabla. En la parte izquierda, se ha incorporado un checkbox azul mediante el cual el cliente puede seleccionar los distintos productos para efectuar el pago. Por tanto, cada vez que se activa o desactiva ese checkbox se actualiza la cantidad de pago total (figura 10), sumándose sólo aquellos productos con el checkbox activo.

3.2.3 Puestos Favoritos

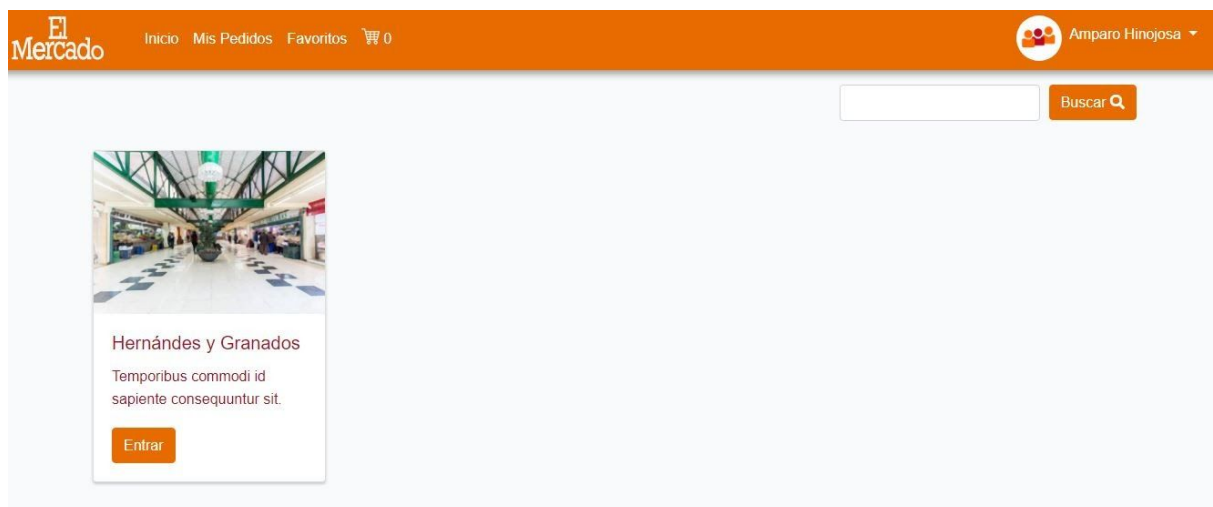
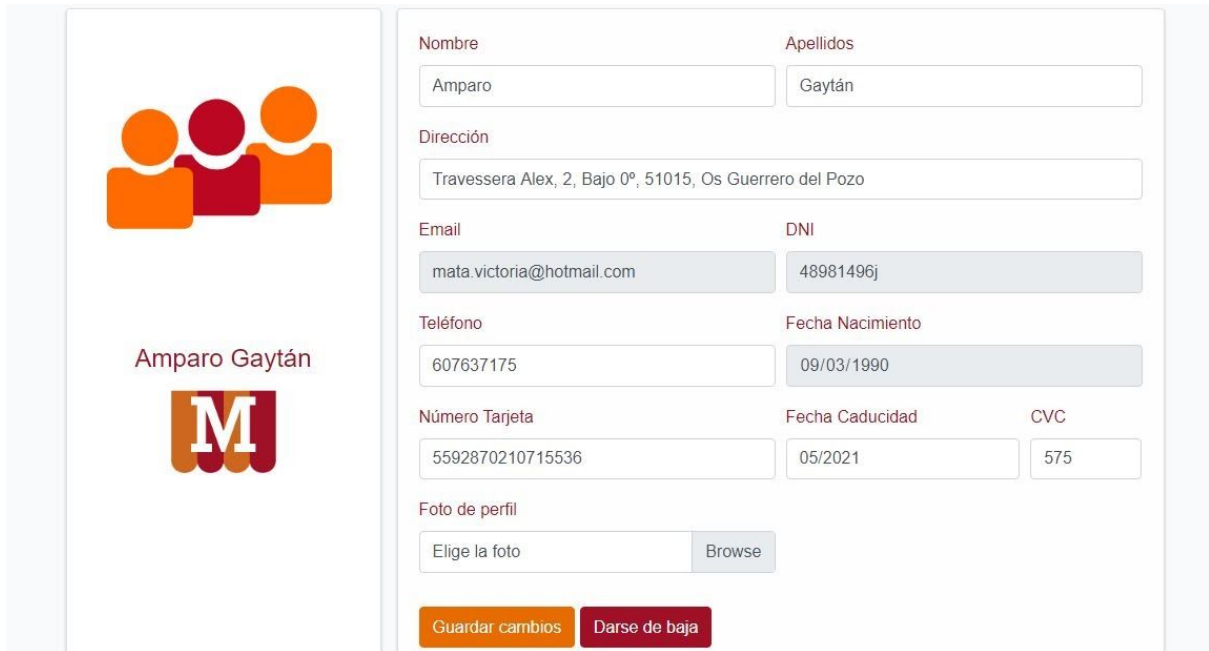


Figura 11. Pantalla favoritos

Esta ventana proporciona al usuario una forma rápida de localizar sus puestos favoritos y de confianza. En esta ventana, sólo se listan aquellos puestos que el cliente haya etiquetado previamente como favoritos.

3.2.4 Modificación Datos Cliente



The screenshot shows a user profile modification interface. On the left, there is a profile card for 'Amparo Gaytán' with a stylized 'M' logo. The main form on the right contains the following fields:

Nombre		Apellidos	
Amparo		Gaytán	
Dirección			
Travessera Alex, 2, Bajo 0º, 51015, Os Guerrero del Pozo			
Email		DNI	
mata.victoria@hotmail.com		48981496j	
Teléfono		Fecha Nacimiento	
607637175		09/03/1990	
Número Tarjeta		Fecha Caducidad	CVC
5592870210715536		05/2021	575
Foto de perfil			
Elige la foto		Browse	

At the bottom of the form are two buttons: 'Guardar cambios' (orange) and 'Darse de baja' (dark red).

Figura 12. Perfil de usuario

Los clientes también disponen de una pestaña para los datos del usuario donde podrán modificar sus datos personales, o cambiar la imagen de usuario dentro de la aplicación web. En esta pantalla el usuario podrá darse de baja en cualquier momento (figura 12).


3.3 Vendedor

Para este tipo de usuario las vistas cambian, en este caso tendría acceso a una parte diferente en la que podría consultar los nuevos pedidos recibidos y a que puesto pertenece en el caso de que disponga de varios puestos activos en la aplicación. También dispondrá de una ventana de históricos de pedidos cerrados por el vendedor. Además de la gestión de los pedidos, se incluye la funcionalidad de gestión de puestos mediante una pantalla específica para este tipo de usuarios, desde aquí se accede también a los productos que hay en cada puesto y puede realizar el CRUD de ellos.

3.3.1 Creación de un puesto

En esta ventana, el usuario vendedor tendrá la posibilidad de crear sus propios puestos donde añadir sus productos, información acerca del puesto, añadir una foto y algunos datos más relacionados con el puesto (figura 15).

Cuando se crea el puesto por primera vez, no se podrán añadir los productos, y para que el usuario tenga constancia de ello justo debajo le indico los pasos a seguir (figura 16).



Nombre: Pedroza, Echevarría y Ayala y Asoc.

Teléfono: 683523452

Información puesto: Ipsum tempore placeat veniam ratione sunt. Nihil simillique ab veniam officiis quia quia qui aliquid. Deserunt culpa laboriosam impedit est ratione beatae. Cupiditate et sunt id

Foto del puesto: Elige la foto [Browse]

Guardar

Nuevo Producto





	Nombre	Precio	Descripción	Stock	Editar	Borrar
	in	1606818 €/kg	Delectus quisquam consequatur amet incidunt omnis distinctio. Quod voluptas porro fugiat. Quod qui quis ut hic. Perferendis eum	No		

Figura 15. Puesto para modificar.



Nombre: |

Teléfono:

Información puesto:

Foto del puesto: Elige la foto [Browse]

Guardar

Información: Una vez creado el puesto, se podrán editar y añadir los productos deseados.

Figura 16. Nuevo puesto.

3.3.2 Gestión de Puestos

En la parte de visualización de puestos, se le muestra un listado de sus puestos al usuario vendedor. Desde dicho listado, se permite dar de baja el puesto o modificar los datos de un puesto en concreto como el nombre, la fotografía, teléfono de contacto o la información sobre su propio puesto (figura 19).






	Nombre	Teléfono	Info	Editar	Borrar
	Pedroza, Echevarría y Ayala y Asoc.	683523452	Ipsum tempore placeat veniam ratione sunt. Nihil simillique ab veniam officiis quia quia qui aliquid. Deserunt culpa laboriosam impedit est ratione beatae. Cupiditate et sunt id consequuntur. Ex occaecati natus perspiciatis fugiat. Alias qui ipsam architecto simillique qui voluptas. g		
	Silva de Valles e Hijos	637336718	Provident aperiam et cum in. Eos perferendis adipisci sunt ducimus tempora eum maxime. Beatae aliquid facilis veniam inventore ab sit incidunt. Ea est fugiat expedita quis aliquam rerum nemo. Eveniet in vel cum asperiores. Itaque qui aliquam aut necessitatibus optio rerum. g		

Figura 19. Pantalla puesto del usuario vendedor.

3.3.3 Gestión de Pedidos

El vendedor tiene la posibilidad de visualizar los pedidos que han realizado los clientes desde sus cuentas, para ello se ha representado en una tabla todos los detalles necesarios para que el vendedor proceda al envío del producto de forma correcta (figura 18).



#	Producto	Puesto	Cantidad	Cliente	Dirección	Teléfono	Fecha	Precio
28	vitae	Barajas de Corral	50 g	Amparo Gaytán	Travessera Alex, 2, Bajo 0º, 51015, Os Guerrero del Pozo	607637175	11/06/2020	199.65 €

Figura 17. Pedidos enviados.

Tras finalizar el envío, el vendedor deberá cambiar el estado del pedido mediante el checkbox del campo Enviar y confirmar los cambios sobre el botón Completar (figura 17).

El Mercado									
Inicio Puesto Pedidos Pedidos Enviados Ayuda									
Amparo									
#	Producto	Puesto	Cantidad	Cliente	Dirección	Teléfono	Fecha	Precio	Enviar
28	vitae	Barajas de Corral	50 g	Amparo Gaytán	Travessera Alex, 2, Bajo 0º, 51015, Os Guerrero del Pozo	607637175	11/06/2020	199.65 €	<input checked="" type="checkbox"/>
Completar									

Figura 18. Pantalla pedidos del vendedor.

3.3.4 Ventana de Ayuda al Usuario

Para ayudar al vendedor ante cualquier duda acerca de la web se ha habilitado un apartado específico (figura 13,14), donde el usuario pueda encontrar un vídeo que ayudaría al usuario a registrar su propio puesto. Para cualquier otra duda, se exponen los datos de contacto de los gestores del sistema con el objetivo de que el usuario pueda trasladar sus dudas por las vías indicadas.



Figura 13. Ayuda vendedor



Figura 14. Ayuda vendedor

Por último, se han incluido unos mandos donde el usuario podría poner música de fondo (figura 14).

3.4 Administrador

Esta parte es la que tiene el control sobre los usuarios y sus contenidos dentro de la web, tiene tres vistas desde las que puede visualizar los usuarios , puestos y productos.

3.4.1 Gestión de Productos

En la pantalla productos como se puede ver en la figura 20, aparece una tabla con todos los productos añadidos en la aplicación, de este modo si el administrador tuviera conocimiento de algún fallo podría o producto inapropiado podría eliminarlo. Esta funcionalidad también podría ser empleada para ayudar en caso de duda a los vendedores.



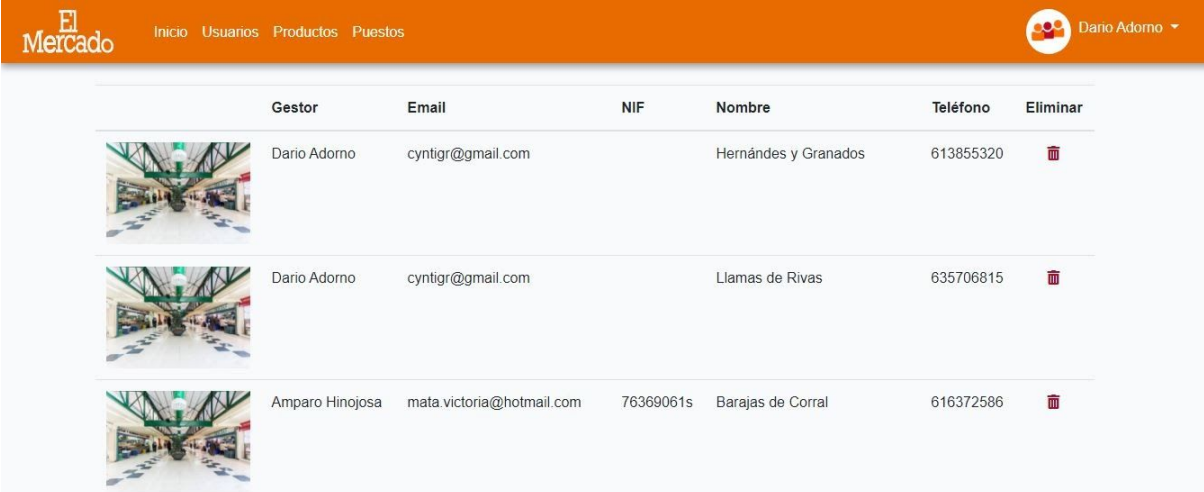
El Mercado								
Inicio Usuarios Productos Puestos								
Dario Adorno								
Producto	Descripción	Precio / kg	Stock	Puesto	Gestor	NIF	Eliminar	
	omnis Nostrum maxime rerum quidem. Cupiditate veritatis quisquam esse soluta ullam. Aliquid inventore quisquam repellat maiores dolor aperiam molestiae. Aut expedita impedit impedit nihil at aperiam.	3 €	No	Hernández y Granados	Dario Adorno			
	sunt Fugit id optio amet explicabo quod voluptatem quia. Qui sed expedita dolorem dolorem expedita. Vel cupiditate laboriosam quo omnis iure modi.	66121 €	Si	Hernández y Granados	Dario Adorno			

Figura 20. Pantalla productos administrador.

3.4.2 Gestión de Puestos

Al igual que en los productos, en la pantalla puestos se visualiza en una tabla con los datos correspondientes, es importante mostrar el correo y el teléfono por si el administrador tuviera que ponerse en contacto con el vendedor por algún problema en la web. Desde aquí también he habilitado la opción de dar de baja un puesto.









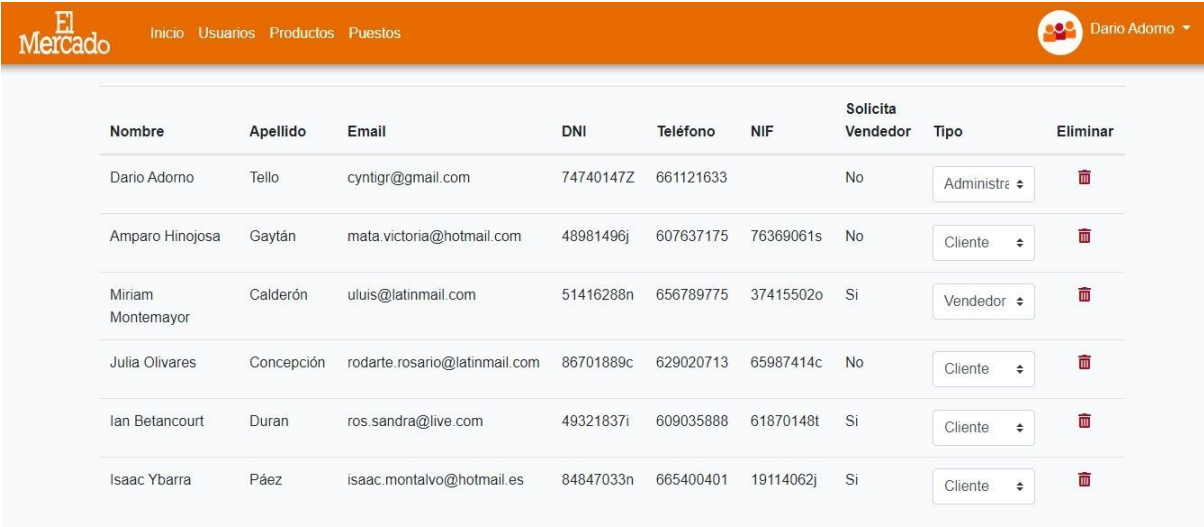
Gestor	Email	NIF	Nombre	Teléfono	Eliminar
	Dario Adorno	cyntigr@gmail.com	Hernández y Granados	613855320	
	Dario Adorno	cyntigr@gmail.com	Llamas de Rivas	635706815	
	Amparo Hinojosa	mata.victoria@hotmail.com	76369061s	Barajas de Corral	616372586 

Figura 21. Pantalla puestos del administrador.

3.4.3 Gestión de Usuarios

Por último, se encuentra la pantalla de datos de usuario, desde la que el administrador tiene el control para cambiar el tipo de usuario con **Ajax**, para no tener que recargar la página cada vez que haga un simple cambio de usuario (figura 22).









Nombre	Apellido	Email	DNI	Teléfono	NIF	Solicita Vendedor	Tipo	Eliminar
Dario Adorno	Tello	cyntigr@gmail.com	74740147Z	661121633		No	Administre ▾	
Amparo Hinojosa	Gaytán	mata.victoria@hotmail.com	48981496j	607637175	76369061s	No	Cliente ▾	
Miriam Montemayor	Calderón	uluis@latinmail.com	51416288n	656789775	37415502o	Si	Vendedor ▾	
Julia Olivares	Concepción	rodarte.rosario@latinmail.com	86701889c	629020713	65987414c	No	Cliente ▾	
Ian Betancourt	Duran	ros.sandra@live.com	49321837i	609035888	61870148t	Si	Cliente ▾	
Isaac Ybarra	Páez	isaac.montalvo@hotmail.es	84847033n	665400401	19114062j	Si	Cliente ▾	

Figura 22. Pantalla usuarios del administrador.

4. DESPLIEGUE

Tras finalizar el desarrollo de la aplicación, se procede al despliegue en la misma en un entorno local y en la plataforma AWS.

4.1. Despliegue Local

Para este apartado se ha montado **XAMPP** que dispone de un servidor Apache ya integrado. Una vez instalado, se debe de añadir una carpeta con el nombre del proyecto (mercado) en la carpeta htdocs de **XAMPP**.

Después en el archivo de configuración de Apache (httpd) añado el **virtualHost** que hace referencia al proyecto para que cuando acceda por el puerto 80 redirija a la web por https (figura 23).

```
<VirtualHost *:80>
ServerName mercado.cgr-ies.edu
Redirect / https://mercado.cgr-ies.edu
ErrorDocument 404 'No me encuentro,prueba con otro recurso'
</VirtualHost>
```

Figura 23. Archivo de configuración de apache httpd.conf.

A continuación, para que sea accesible desde https la aplicación web, se ha añadido la configuración a https.conf y así poder acceder de forma segura (figura 24).

```
298 <VirtualHost *:443>
299     DocumentRoot "C:/xampp/htdocs/mercado/public"
300     ServerName mercado.cgr-ies.edu
301     ServerAlias www.mercado.cgr-ies.edu
302     SSLEngine on
303     SSLCertificateFile "conf/ssl.crt/claveSSL.crt"
304     SSLCertificateKeyFile "conf/ssl.key/claveSSL.key"
305 </VirtualHost>
```

Figura 24. Archivo de configuración de apache httpd-ssl.conf.

Y por último, se procede a la configuración de **DNS zona inversa** (figura 25) y **zona directa** (figura 26) para poder acceder desde la red interna o dentro del sistema en el que el usuario se encuentre en dicho momento.

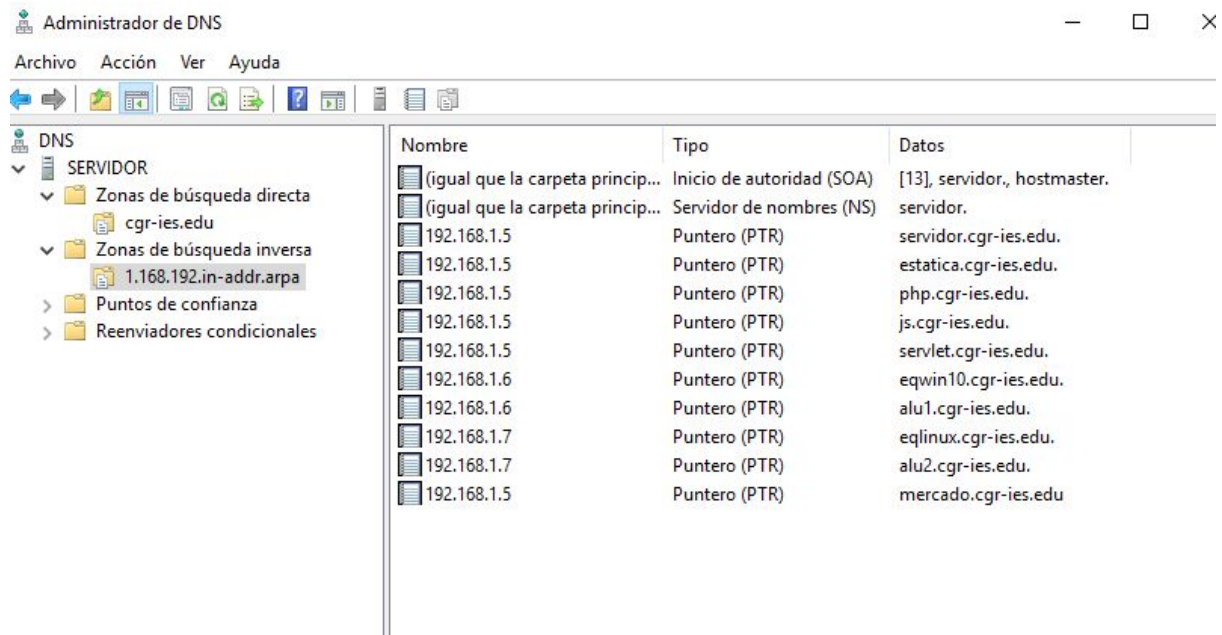


Figura 25. Configuración DNS zona inversa.

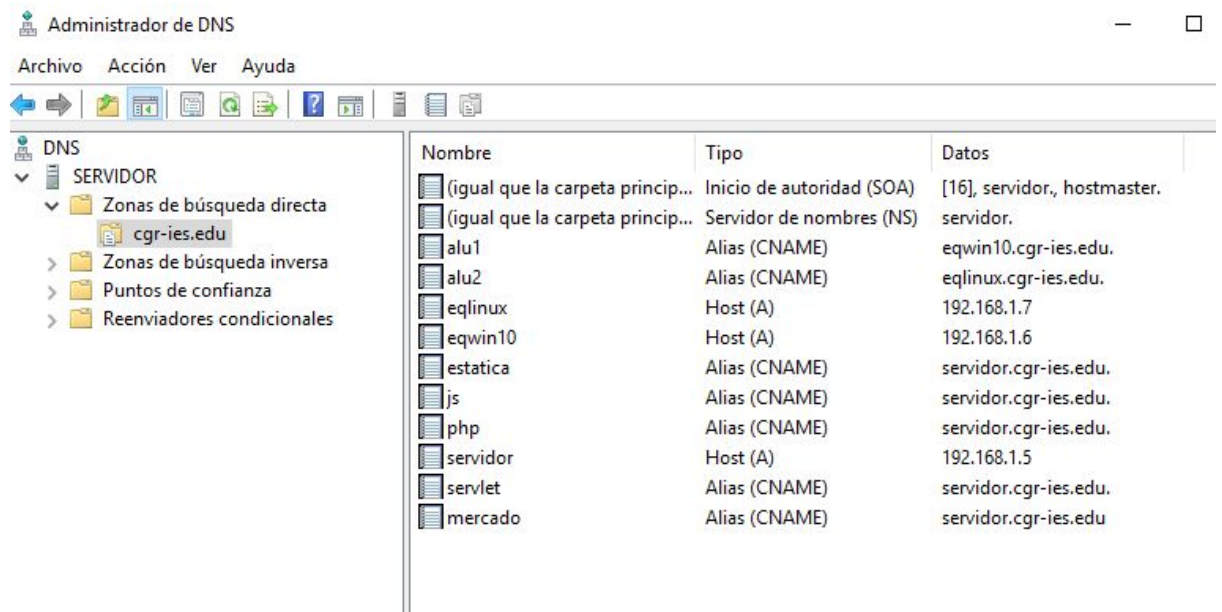


Figura 26. Configuración DNS zona directa.

4.2. Despliegue AWS

Para el despliegue en el cloud de Amazon he utilizado el componente **Elastic Beanstalk**, que permite implementar y administrar rápidamente aplicaciones en la nube de AWS de forma independiente a la infraestructura sobre la que se ejecuta.

Los pasos a seguir en el despliegue son:

- Añadir al proyecto **Laravel** una carpeta llamada “.ebextensions” con un archivo de configuración que lance las migraciones y seed de la aplicación, este archivo puede ser de código **YAML** o **JSON**.

```
container_commands:
  01-migrate:
    command: php artisan migrate
  02-seed:
    command: php artisan db:seed
```

Código para lanzar comandos artisan en el servidor de AWS

- En la carpeta *bootstrap/cache* borrar el archivo *config.php* para que no importe la configuración del proyecto en desarrollo.
- Después se modifica el archivo *database.php* que se encuentra en la carpeta *config*, para que la aplicación importe la configuración de la base de datos del servidor automáticamente, sin tener que pasarle los datos de la conexión a mano.

```
'host' => env('RDS_HOSTNAME', '127.0.0.1'),
'port' => env('RDS_PORT', '3306'),
'database' => env('RDS_DB_NAME', 'forge'),
'username' => env('RDS_USERNAME', 'forge'),
'password' => env('RDS_PASSWORD', ''),
```

Código archivo de configuración database.php

- Una vez hechos todos estos cambios en el proyecto se seleccionan todos los archivos que existentes dentro de la carpeta del proyecto y se comprimen en un archivo ZIP.
- A continuación, desde la consola de administración se crea un entorno de servidor web.
- Se añade el nombre de la aplicación , información del entorno que es la parte donde hay que darle nombre a la web, plataforma que es donde defino que mi proyecto es de tipo PHP y la versión la 7.2 y código de la aplicación que escojo la opción cargar código.
- Luego se añaden dos opciones más:

- En opciones de contenedor cambió el raíz del documento que corresponde a /public, porque el archivo index de la aplicación se encuentra en este directorio..
- Activo una base de datos MySQL en la que se añade usuario y contraseña.
- Por último, se lanza y el sistema crea la aplicación en la web. Después dentro de la consola se puede llevar un control de versiones de la aplicación. Además, se puede tener las versiones en la cloud y volver atrás en el momento que se quiera.

Elastic Beanstalk > Aplicaciones > Marketstall > Versiones de la aplicación

Versiones de la aplicación

Acciones ▼
Configuración
Cargar
Actualizar

<input type="checkbox"/>	Etiqueta de la versión	Descripción	Fecha de creación	Fuente	Se implementa en
<input type="checkbox"/>	marketstall-source		2020-06-12T10:48:21+02:00	2020164BDc-Proyect-Definitiv.zip	Marketstall-env

5. CONCLUSIONES

Una vez finalizado el desarrollo y el despliegue en las distintas modalidades, he obtenido las siguientes conclusiones.

- He desarrollado una aplicación relacionada con algo cercano a mi día y día en un entorno que conozco, lo que ha supuesto una motivación extra.
- Además de cumplir con los requisitos de desarrollo indicados, este proyecto a supuesto un repaso y profundización de muchas de las tecnologías estudiadas durante el curso como PHP, Laravel, Javascript, html5, CSS3...

Se han cumplido los requisitos de despliegue al llevar a cabo un despliegue con Apache y otro con AWS. Con el despliegue en local, he puesto en práctica los conocimientos de clase mientras que con AWS he podido conocer un entorno real de publicación de aplicaciones web. Ahora entiendo un poco mejor el crecimiento y la importancia de este tipo de plataformas y los beneficios que nos pueden reportar.

6. BIBLIOGRAFÍA

Páginas para Bootstrap:

- <https://www.w3schools.com/bootstrap4/default.asp>
- <https://bootstrap-vue.org/docs/components/table>
- <https://getbootstrap.com/docs/4.5/getting-started/introduction>

Código JS:

- https://www.w3schools.com/js/js_intro.asp

Documentación PHP:

- <https://www.phpdoc.org/>

Código PHP:

- <https://www.php.net/docs.php>

Laravel:

- <https://laravel.com>

Amazon Web Services:

- https://docs.aws.amazon.com/es_es/elasticbeanstalk/latest/dg/Welcome.html

Ayuda y fallos:

- <https://es.stackoverflow.com/>
- [https://blog.pleets.org/article/mvc-en-laravel#:~:text=MVC%20\(modelo%2C%20vista%2C%20controlador,un%20aspecto%20de%20la%20aplicaci%C3%B3n.](https://blog.pleets.org/article/mvc-en-laravel#:~:text=MVC%20(modelo%2C%20vista%2C%20controlador,un%20aspecto%20de%20la%20aplicaci%C3%B3n.)
- <https://rafarjonilla.com/que-es/backend/>
- <https://www.hostinger.es/tutoriales/que-es-mysql/>
- https://es.wikipedia.org/wiki/Servidor_HTTP_Apache

Apuntes DWES