



10 DE JUNIO DE 2025

WECOOK – RED SOCIAL DE RECETAS

DESARROLLO DE APLICACIONES WEB

SERGIO ÁLVAREZ DE RON
JONATHAN HIDALGO
SERGIO MONTOIRO

IES LUIS BRAILLE



Índice de contenidos

1. Resumen	3
2. Abstract.....	3
3. Introducción.....	4
4. Objetivos	5
5. Alcance del proyecto	6
5.1. Requisitos funcionales y no funcionales.....	6
Requisitos funcionales	6
Requisitos no funcionales	7
5.2. Prototipado.....	8
Wireframing	8
Bocetado de vistas.....	9
5.3. Tecnologías utilizadas	18
Laravel.....	18
Blade	18
JavaScript	19
Figma	21
Miro	21
Git	20
GitHub.....	20
Visual Studio Code	20
MySQL.....	19
MySQL Workbench	19
Bootstrap	19
5.4. Diagrama de Entidad-Relación	22
5.5. Diagrama de Casos de Usos.....	24
6. Marco práctico	25

Estructura	25
Controladores	29
Modelos	29
Rutas	29
Vistas.....	29
Base de datos.....	30
Vista de inicio.....	31
Vista de recordar contraseña:.....	32
Cabecera común para todas las vistas.....	32
Pie común para todas las vistas	32
Vista de listado de recetas:.....	33
Vista de detalle de receta	33
Vista de creación/edición de receta:	34
Vista de perfil de usuario.....	34
Vista de seguidores/seguídos del usuario	35
Modal de seguidores/seguídos.....	35
Modal recetas guardadas	36
Modal de editar receta	36
Vista de administrador de recetas.....	36
Vista de administrador de usuarios	37
7. Conclusiones.....	38
8. Bibliografía	39
9. Webgrafía.....	39

1. RESUMEN

Este proyecto consiste en el desarrollo de una aplicación web que funciona como red social para compartir recetas de cocina. El objetivo principal ha sido diseñar y construir una plataforma donde los usuarios puedan registrarse, publicar sus propias recetas, comentar las de otros usuarios y gestionar su contenido de forma segura.

Para ello se ha utilizado Laravel 12, un framework de PHP moderno y robusto, que ha facilitado la implementación de funcionalidades habituales en aplicaciones web como autenticación, rutas, controladores, relaciones entre modelos, validación de formularios, entre otras. A través de este proyecto hemos aprendido a estructurar el backend de una aplicación web de manera eficiente y a utilizar herramientas comunes en el desarrollo profesional de software. Además, hemos podido aplicar conocimientos de bases de datos relacionales, diseño responsive y buenas prácticas de seguridad.

2. ABSTRACT

This project focuses on the development of a web application that acts as a social network for sharing cooking recipes. The main goal was to design and implement a platform where users can register, publish their own recipes, comment on other users' posts, and manage their content securely.

The project has been developed using Laravel 12, a modern and powerful PHP framework that simplifies the creation of web applications through built-in features such as authentication, routing, MVC structure, form validation, and database relationships. Throughout this project, we have gained valuable knowledge about how to structure and build scalable and maintainable web applications using current technologies, while applying key concepts learned during the degree.

3. INTRODUCCIÓN

En el presente trabajo se propone el diseño y desarrollo de una red social centrada en compartir recetas de cocina, utilizando tecnologías modernas como Astro, Laravel y MySQL. El auge de las redes sociales y el creciente interés por la gastronomía han impulsado la creación de comunidades virtuales dedicadas a compartir experiencias culinarias. Esta plataforma busca ofrecer un espacio donde los usuarios puedan intercambiar recetas, consejos, fotos y videos, fomentando una experiencia interactiva y colaborativa.

El desarrollo de esta red social se sustentará en el uso de Astro, un framework frontend que permite crear aplicaciones rápidas y ligeras, aprovechando la posibilidad de integrar diversas tecnologías y herramientas. Laravel, un potente framework de PHP, se utilizará para el desarrollo del backend, gestionando la lógica de negocio, la autenticación de usuarios y las interacciones dentro de la plataforma. Por otro lado, MySQL será la base de datos elegida para almacenar toda la información relacionada con los usuarios, las recetas, comentarios y otros datos esenciales de la aplicación.

El objetivo principal de este proyecto es ofrecer una solución práctica, escalable y fácil de usar para los amantes de la cocina, permitiendo a los usuarios compartir sus creaciones de manera sencilla y accesible. Además, se incluirán funciones que fomenten la interacción social, como comentarios, valoraciones y recomendaciones.

4. OBJETIVOS

- Desarrollar una aplicación web funcional que permita a los usuarios registrarse, iniciar sesión y gestionar su cuenta de manera segura y sencilla, utilizando Laravel como framework.
- Implementar un sistema para la publicación de recetas de cocina, en el que los usuarios puedan crear, editar y eliminar sus propias recetas incluyendo título, descripción, ingredientes, pasos de preparación y una imagen ilustrativa.
- Diseñar una interfaz web intuitiva, atractiva y responsive, garantizando una experiencia de usuario óptima en dispositivos móviles y de escritorio.
- Definir y gestionar una base de datos relacional en MySQL, destinada a almacenar de forma eficiente y segura la información de usuarios, recetas y demás entidades necesarias para el funcionamiento de la aplicación.
- Incorporar un sistema de búsqueda y filtrado de recetas, que permita a los usuarios encontrar contenidos relevantes según nombre, ingredientes o categoría.
- Aplicar mecanismos de seguridad en el desarrollo de la aplicación, tales como validación de formularios, protección contra ataques comunes (como inyecciones SQL) y almacenamiento seguro de contraseñas.
- Aplicar buenas prácticas de desarrollo web tanto en el frontend como en el backend, incluyendo el uso de control de versiones con Git, documentación del código y separación de responsabilidades entre componentes.
- Realizar pruebas funcionales y de usuario, con el objetivo de verificar el correcto funcionamiento de todas las funcionalidades implementadas y asegurar una buena experiencia de uso.

5. ALCANCE DEL PROYECTO

5.1. Requisitos funcionales y no funcionales

Requisitos funcionales

Los requisitos funcionales describen las funcionalidades que el sistema debe ofrecer al usuario final. Para esta aplicación web de recetas de cocina, se han identificado los siguientes:

- Registro de usuarios: La aplicación debe permitir a nuevos usuarios registrarse mediante un formulario con nombre, correo electrónico y contraseña.
- Inicio de sesión y cierre de sesión: Los usuarios registrados deben poder autenticarse para acceder a funcionalidades personalizadas, así como cerrar sesión de forma segura.
- Gestión de perfil de usuario: Cada usuario debe poder visualizar y editar la información de su perfil, incluyendo su nombre, foto (opcional) y contraseña.
- Seguir usuarios: cada usuario tiene la opción de seguir a otros usuarios y viceversa, para poder ver las recetas de estos en su página de inicio de forma prioritaria.
- Publicación de recetas: Los usuarios autenticados podrán crear recetas introduciendo título, descripción, ingredientes, pasos de preparación e imagen.
- Edición y eliminación de recetas: Los usuarios podrán editar o eliminar únicamente las recetas que hayan creado.
- Visualización de recetas: Cualquier visitante podrá visualizar todas las recetas disponibles, incluyendo su contenido completo y la información del autor.
- Buscador de recetas: La aplicación debe ofrecer un sistema de búsqueda por palabras clave (título, ingredientes, etc.).
- Filtrado de recetas: El usuario podrá filtrar recetas por categorías predefinidas (por ejemplo: entrantes, platos principales, postres...).
- Sistema de comentarios: Los usuarios autenticados podrán comentar en las recetas de otros usuarios.
- Marcado de recetas favoritas: Los usuarios podrán marcar recetas como favoritas para consultarlas posteriormente.

Requisitos no funcionales

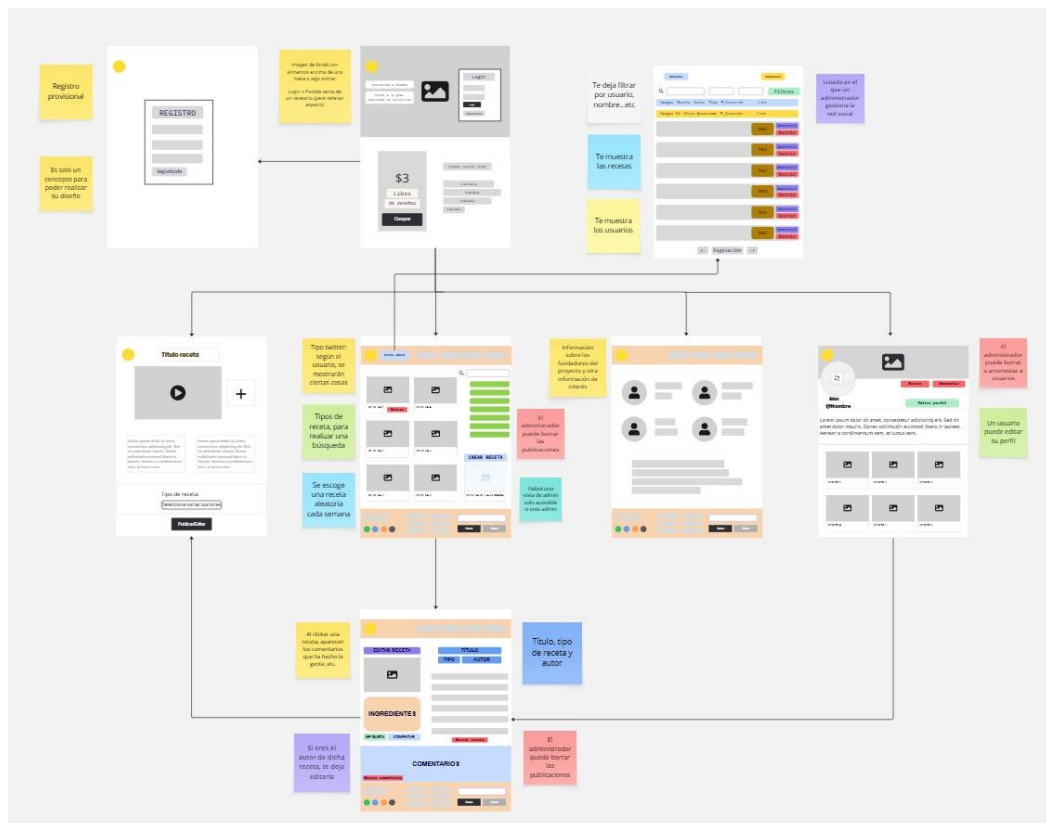
Los requisitos no funcionales definen características de calidad y restricciones técnicas que debe cumplir el sistema:

- Seguridad: Las contraseñas de los usuarios deben almacenarse cifradas. Se deben validar las entradas del usuario para evitar ataques como inyección SQL. Se debe implementar protección contra ataques CSRF en los formularios.
- Usabilidad: La interfaz debe ser intuitiva, fácil de usar y accesible para usuarios sin conocimientos técnicos. Se debe garantizar una experiencia fluida tanto en dispositivos móviles como en ordenadores.
- Rendimiento: Las páginas deben cargarse en un tiempo razonable (<2 segundos en condiciones normales). El sistema debe ser capaz de manejar múltiples usuarios y recetas sin disminuir significativamente su rendimiento.
- Compatibilidad: La aplicación debe ser compatible con los navegadores web más utilizados (Chrome, Firefox, Edge, Safari).
- Mantenibilidad: El código debe estar estructurado de forma clara y modular, utilizando buenas prácticas de programación para facilitar su mantenimiento y ampliación.
- Escalabilidad: La arquitectura basada en frontend y backend separados debe permitir futuras ampliaciones (por ejemplo, integración con APIs, app móvil, etc.).
- Portabilidad: El proyecto debe poder instalarse y desplegarse fácilmente en otros entornos utilizando herramientas comunes como Docker o entornos LAMP/WAMP.

5.2. Prototipado

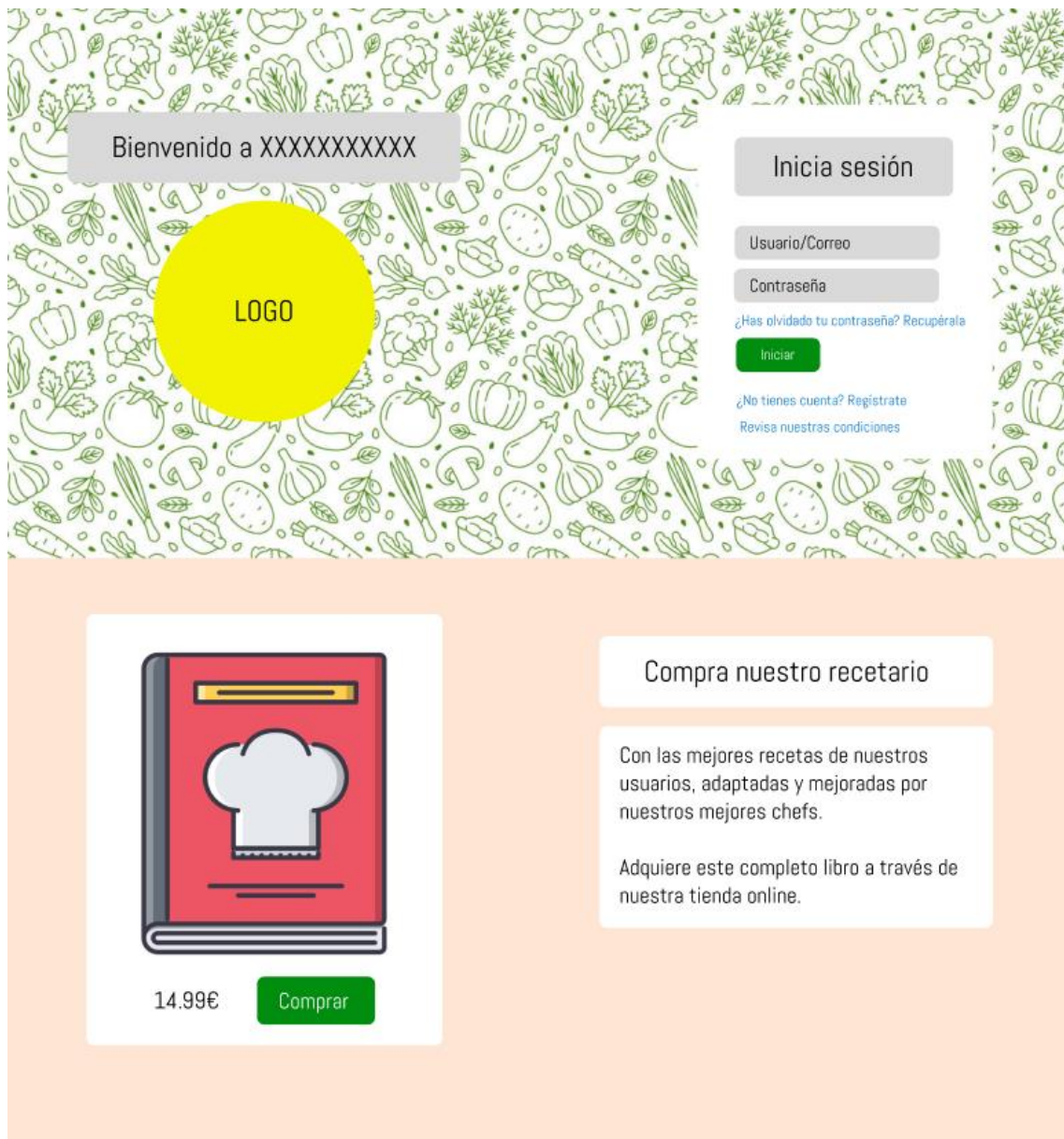
Wireframing

https://miro.com/welcomeonboard/eGptdE1iVEdpZmcvM1M1RIJwTERXUmZmTkZjN3FzRGgrTEFoc0JKbINkekZ2eE8xYWR6UXpFbm9vWEEwR1pLcFd6WE8rSUNDRnZiTGfFpakZEN244ZUGyTIVIZVhZNGNGdXJzbXJFNmFqNW5FSmJxY1lhQWFjNE9JNEI3SSs4SHhBd044SHFHaVIWYWk0d3NxeHNmeG9BPT0hdjE=?share_link_id=690549974316



Bocetado de vistas

- Landing page:



- About us:



- Publicar y editar receta:

Pulpo a la gallega



+

Ingredientes:

- Pulpo
- Pimentón Dulce(o picante).
- Sal gruesa
- Aceite de oliva

Procedimiento:


1. Cuece las patatas
2. Cuece el pulpo
3. Emplatamos colocando primero las patatas, luego el pulpo...

Tipo de receta:

Moluscos

Publicar/Editar


- Perfil de usuario:




@Pepe24

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed sit amet dolor mauris. Donec sollicitudin euismod libero in laoreet. Aenean a condimentum sem, at luctus sem.


Recetas



Garbanzos con espinacas

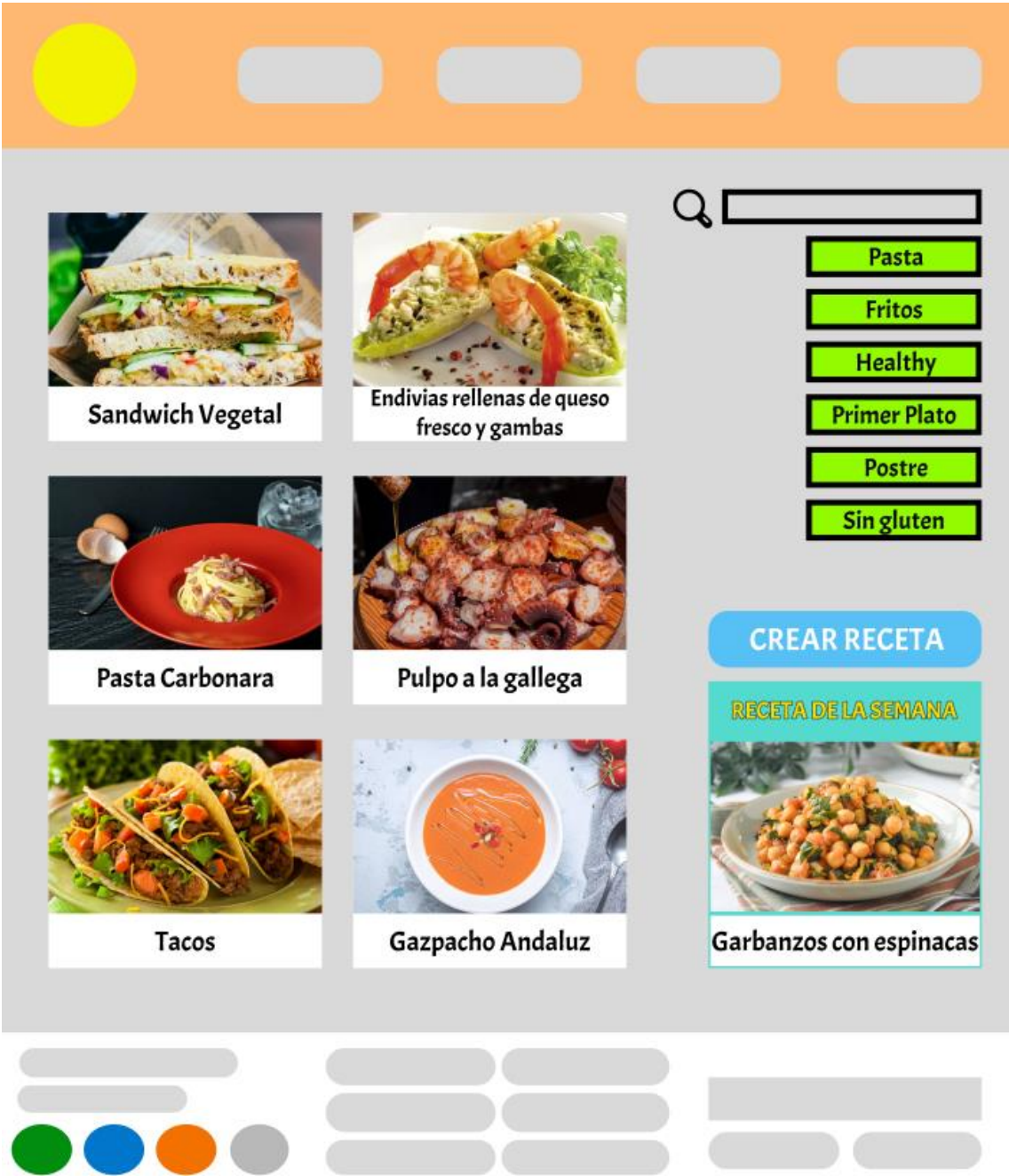


Tacos



Pulpo a la gallega

- Página principal:



- Página de receta:



EDITAR RECETA



- Garbanzos cocidos
- Dientes de ajo
- Pan de hogaza rebanada
- Comino molido
- Cilantro molido
- Pimentón dulce
- Pimienta blanca molida pizca
- Salsa de tomate
- Vinagre de manzana



Comentar

Garbanzos con espinacas

Legumbre

Por: Pepe 24

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed sit amet dolor mauris. Donec sollicitudin euismod libero in laoreet. Aenean a condimentum sem, at luctus sem.



@Jonathan20
¡Gracias por la receta! La probé anoche y quedó buenísima



@Álvarez26
Mi profesor de Javascript me recomendó esta página, ¡me alegro de haberla encontrado por fin!



- Página de administrador:

Recetas

Usuarios

Q

Id

Alias

@Username

F_Creación

Filtrar

Imagen	Id	Alias	@Username	F_Creación	Link	
	12	Pepito	@tuPepitoFav	20/02/2025	Ver	<div>Amonestar</div> <div>Borrar</div>
	134	Álvaro	@Alvaro845	13/04/2025	Ver	<div>Amonestar</div> <div>Borrar</div>
	211	Sergio	@SxrgitoLoco	27/04/2025	Ver	<div>Amonestar</div> <div>Borrar</div>
	215	Jonathan	@JxnaCook	1/05/2025	Ver	<div>Amonestar</div> <div>Borrar</div>

←

1

2

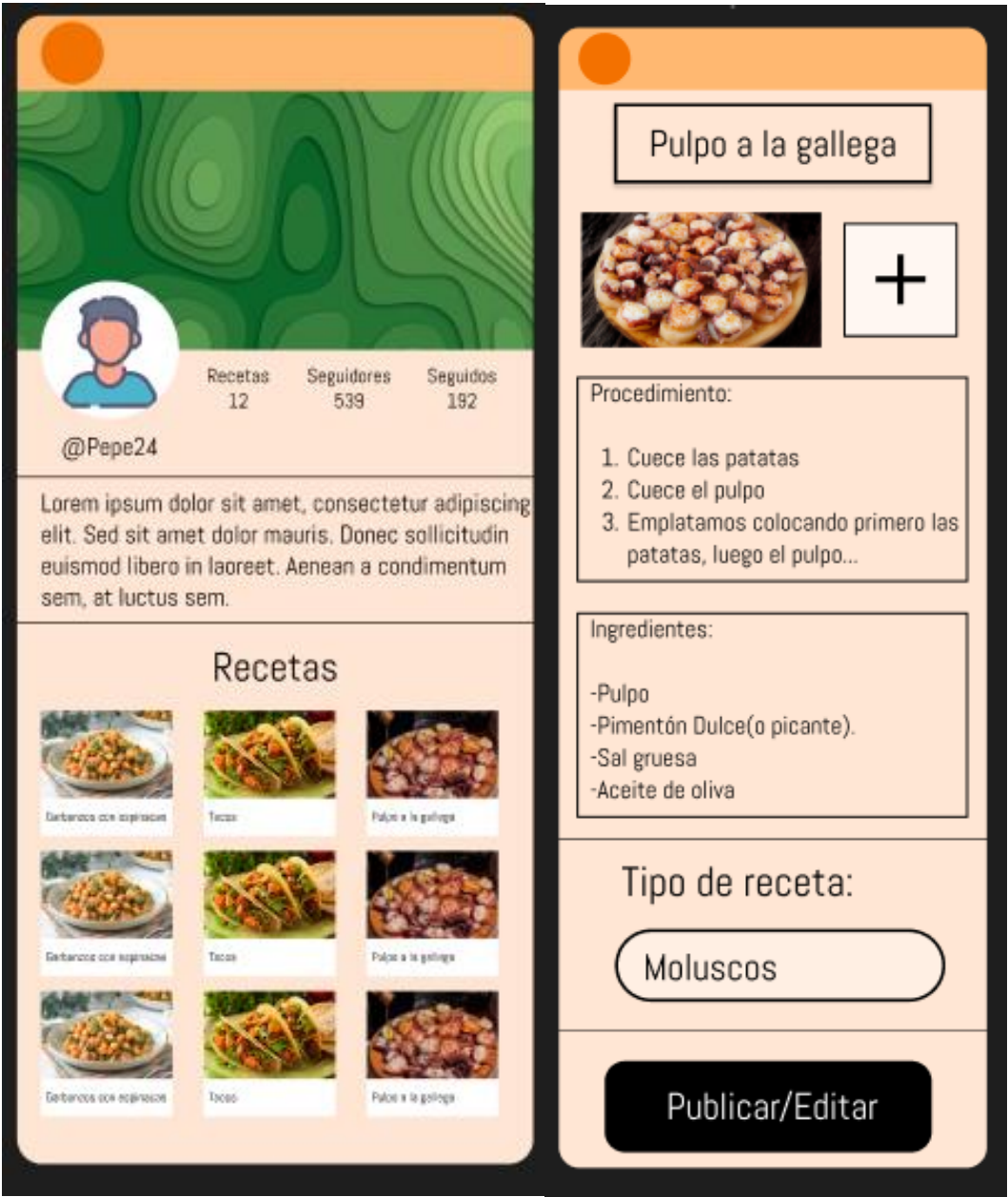
3

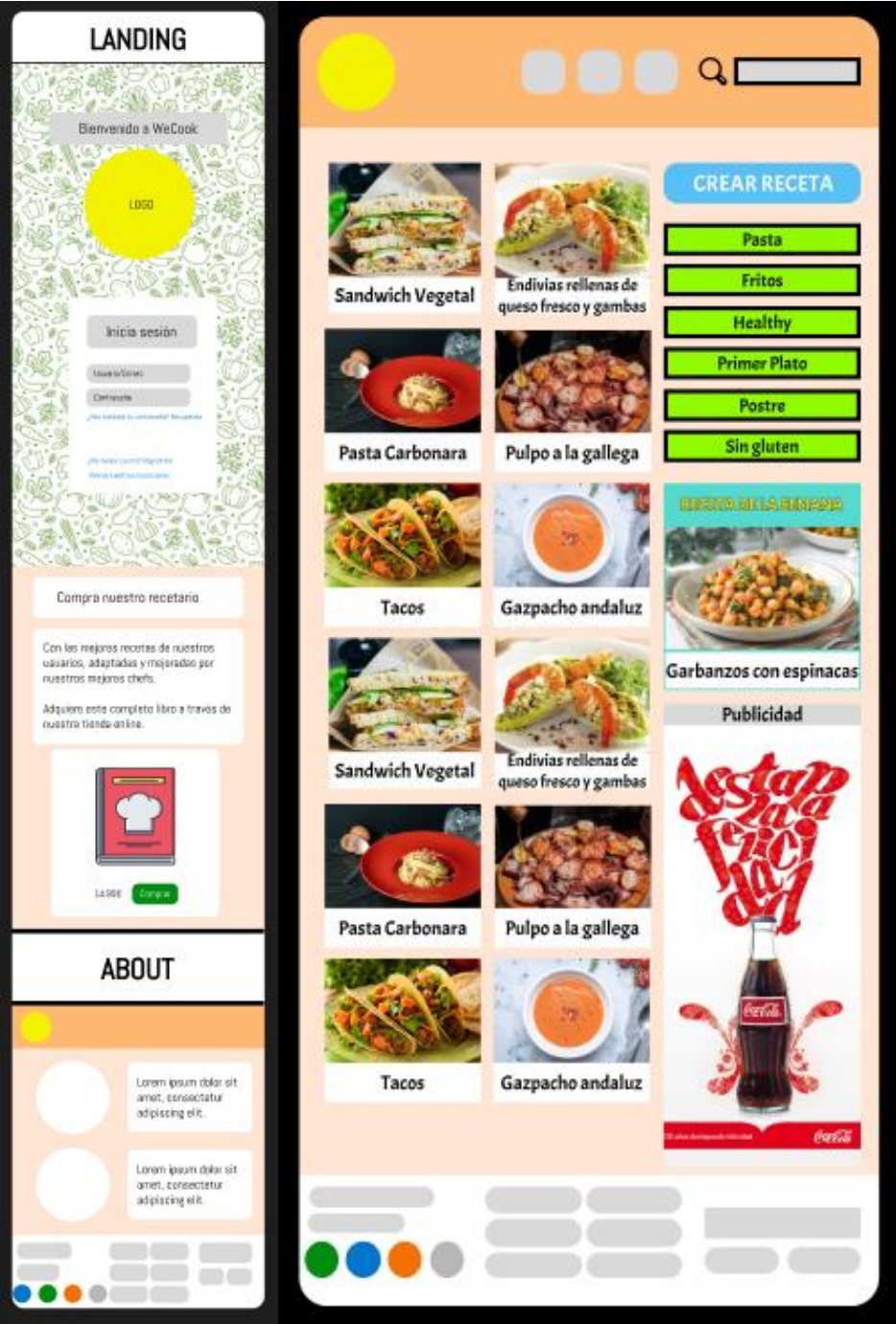
...

23

→

Vistas en versión móvil







EDITAR RECETA

Garbanzos con espinacas

Legumbre

Por: Pepe 24



Ingredientes:

- Garbanzos cocidos
- Dientes de ajo
- Pan de hogaza rebanada
- Comino molido
- Cilantro molido
- Pimentón dulce
- Pimienta blanca molida
- Salsa de tomate
- Vinagre de manzana

Lorem ipsum dolor
sit amet,
consectetur
adipiscing elit. Sed
sit amet dolor
mauris. Donec
sollicitudin euismod
libero in laoreet.
Aenean a
condimentum sem,
at luctus sem.



43



4



@Jonathan20

¡Gracias por la receta! La probé anoche y quedó buenísima



@Álvarez26

Mi profesor de Javascript me recomendó esta página, ¡me alegro de haberla encontrado por fin!



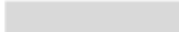
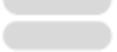
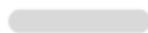
@Jonathan20

¡Gracias por la receta! La probé anoche y quedó buenísima



@Álvarez26

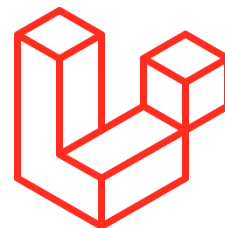
Mi profesor de Javascript me recomendó esta página, ¡me alegro de haberla encontrado por fin!



5.3. Tecnologías utilizadas

Laravel

Laravel es un framework de desarrollo web basado en PHP que permite construir aplicaciones modernas de forma estructurada y eficiente. Utiliza el patrón de arquitectura Modelo-Vista-Controlador (MVC), lo que facilita la organización del código separando la lógica de negocio, la interfaz y los datos. Laravel proporciona herramientas integradas como el sistema de rutas, migraciones de base de datos, autenticación, validación de formularios y plantillas Blade, lo que reduce y facilita significativamente el desarrollo del proyecto, a la vez que es sencillo de aprender.



Hemos utilizado esta tecnología debido a su popularidad en el entorno backend y a su capacidad para manejar fácilmente tareas complejas como la gestión de usuarios, seguridad, y acceso a bases de datos, además de estar basado en PHP, lenguaje con el que estamos muy familiarizados tras haber trabajado con él durante todo el curso. Además, su sintaxis clara favorece el mantenimiento y escalabilidad del proyecto.

Blade

Blade es el motor de plantillas que viene integrado en Laravel y se utiliza para construir la parte visual de la aplicación. Permite escribir vistas de forma sencilla y estructurada, combinando HTML con una sintaxis específica que facilita la inclusión de lógica directamente en las plantillas, como bucles, condicionales y estructuras heredadas.



Una de las principales ventajas de Blade es que permite reutilizar componentes de la interfaz mediante plantillas y secciones, lo que promueve una mejor organización del código frontend. También permite el uso de directivas como `@if`, `@foreach`, `@include` o `@yield`, que simplifican mucho la creación de páginas dinámicas sin necesidad de mezclar directamente PHP con HTML.

Hemos elegido Blade como sistema de plantillas porque se integra perfectamente con Laravel y proporciona una forma clara, limpia y mantenible de construir las vistas. Su uso nos ha permitido separar correctamente la lógica del backend y la presentación del frontend, favoreciendo así el desarrollo ágil y ordenado de la aplicación.

JavaScript

JavaScript es un lenguaje de programación interpretado, ampliamente utilizado en el desarrollo web para añadir interactividad y dinamismo a las páginas. Se ejecuta del lado del cliente, lo que permite modificar el contenido del HTML y CSS en tiempo real sin necesidad de recargar la página.



Esta tecnología es fundamental para nuestro proyecto ya que permite crear funcionalidades interactivas como validación de formularios, menús desplegables, animaciones y comunicación asíncrona con el servidor mediante AJAX. Gracias a su versatilidad y compatibilidad con todos los navegadores, JavaScript es esencial en cualquier aplicación web moderna.

Bootstrap

Bootstrap es un framework front-end que proporciona estilos CSS y componentes JavaScript listos para usar. Fue creado por Twitter y está enfocado en el diseño responsive, permitiendo que las aplicaciones se adapten automáticamente a diferentes tamaños de pantalla. Hemos utilizado Bootstrap para construir una interfaz visual atractiva y funcional sin necesidad de escribir mucho CSS personalizado. Gracias a sus componentes predefinidos como formularios, botones, menús y grillas, pudimos acelerar el desarrollo de la parte visual de nuestra aplicación y garantizar una buena experiencia en dispositivos móviles.



MySQL

MySQL es un sistema de gestión de bases de datos relacional de código abierto. Permite almacenar, consultar y gestionar grandes cantidades de datos estructurados de manera eficiente mediante el uso de lenguaje SQL.



Hemos utilizado MySQL como base de datos principal del proyecto, ya que permite definir relaciones entre entidades como usuarios, recetas o comentarios. Su compatibilidad con Laravel mediante Eloquent ORM facilita el acceso a los datos de forma sencilla y segura.

MySQL Workbench

MySQL Workbench es una herramienta visual que permite modelar, administrar y consultar bases de datos MySQL de manera gráfica. Incluye funciones para diseñar esquemas, realizar consultas SQL, gestionar usuarios y analizar el rendimiento.



La hemos utilizado para diseñar y mantener la estructura de nuestra base de datos, visualizar relaciones entre tablas y ejecutar consultas de prueba, lo cual ha sido de gran ayuda en la fase de desarrollo y prueba del proyecto.

Git

Git es un sistema de control de versiones distribuido que permite gestionar el historial de cambios en el código fuente de un proyecto. Facilita la colaboración entre varios desarrolladores al permitir trabajar en ramas independientes y luego fusionar cambios de manera fácil y segura.



En nuestro TFG, Git ha sido fundamental para trabajar en equipo sin conflictos, mantener un historial de versiones y poder recuperar estados anteriores del proyecto en caso de errores. Su integración con plataformas como GitHub lo convierte en una herramienta indispensable en cualquier entorno de desarrollo profesional.

GitHub

GitHub es una plataforma web para alojar repositorios de Git. Proporciona funcionalidades colaborativas y gestión de versiones, además de facilitar el trabajo en equipo y la revisión de código.



Hemos utilizado GitHub para centralizar nuestro proyecto, compartir avances, y realizar control de versiones de forma remota. Su interfaz amigable y sus herramientas de colaboración han permitido una organización clara y efectiva del flujo de trabajo durante el desarrollo de la aplicación.

Visual Studio Code

Visual Studio Code (VS Code) es un editor de código fuente ligero, potente y multiplataforma desarrollado por Microsoft. Incluye herramientas de depuración, resaltado de sintaxis, autocompletado y un amplio ecosistema de extensiones.



En nuestro proyecto, VS Code ha sido nuestro entorno de desarrollo principal, facilitando la escritura de código en Laravel, JavaScript y otros lenguajes gracias a su integración con Git, terminal integrada y compatibilidad con diversas extensiones específicas para cada tecnología utilizada.

Figma

Figma es una herramienta de diseño gráfico basada en la nube, especialmente orientada al diseño de interfaces de usuario (UI) y experiencia de usuario (UX). Permite trabajar de forma colaborativa en tiempo real, lo que facilita la creación de prototipos interactivos, wireframes y maquetas de alta fidelidad antes del desarrollo.



Hemos utilizado Figma para diseñar visualmente las pantallas de nuestra aplicación web, asegurándonos de que la experiencia de usuario sea intuitiva y atractiva. Gracias a sus funcionalidades de colaboración y su enfoque centrado en el diseño responsive, ha sido clave para definir la estructura visual del proyecto.

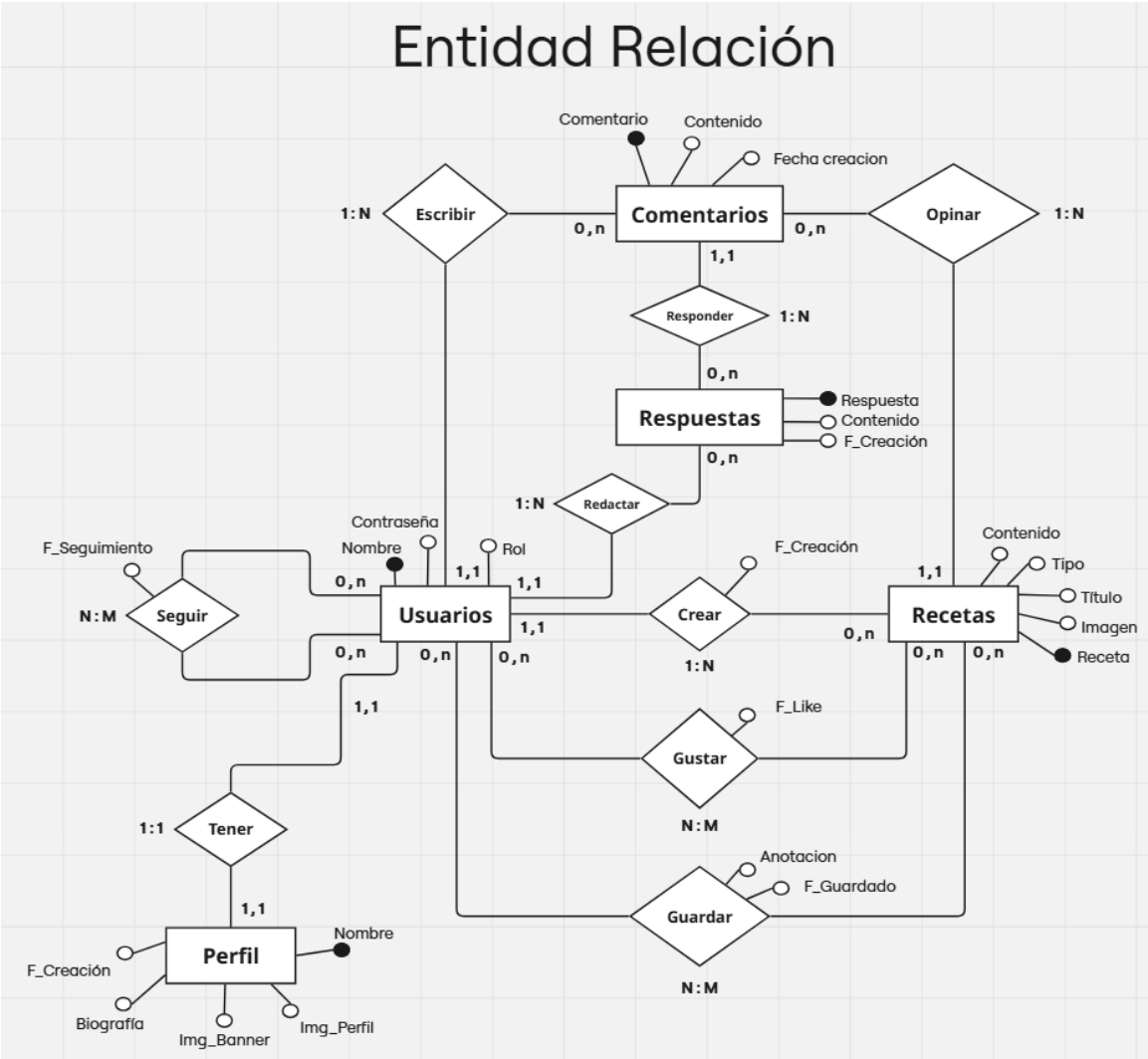
Miro

Miro es una plataforma colaborativa de pizarras digitales en línea diseñada para facilitar la planificación, organización y trabajo en equipo de forma visual. Permite a varios usuarios interactuar en tiempo real sobre una misma pizarra, añadiendo notas, diagramas, mapas mentales, esquemas y otros elementos visuales.

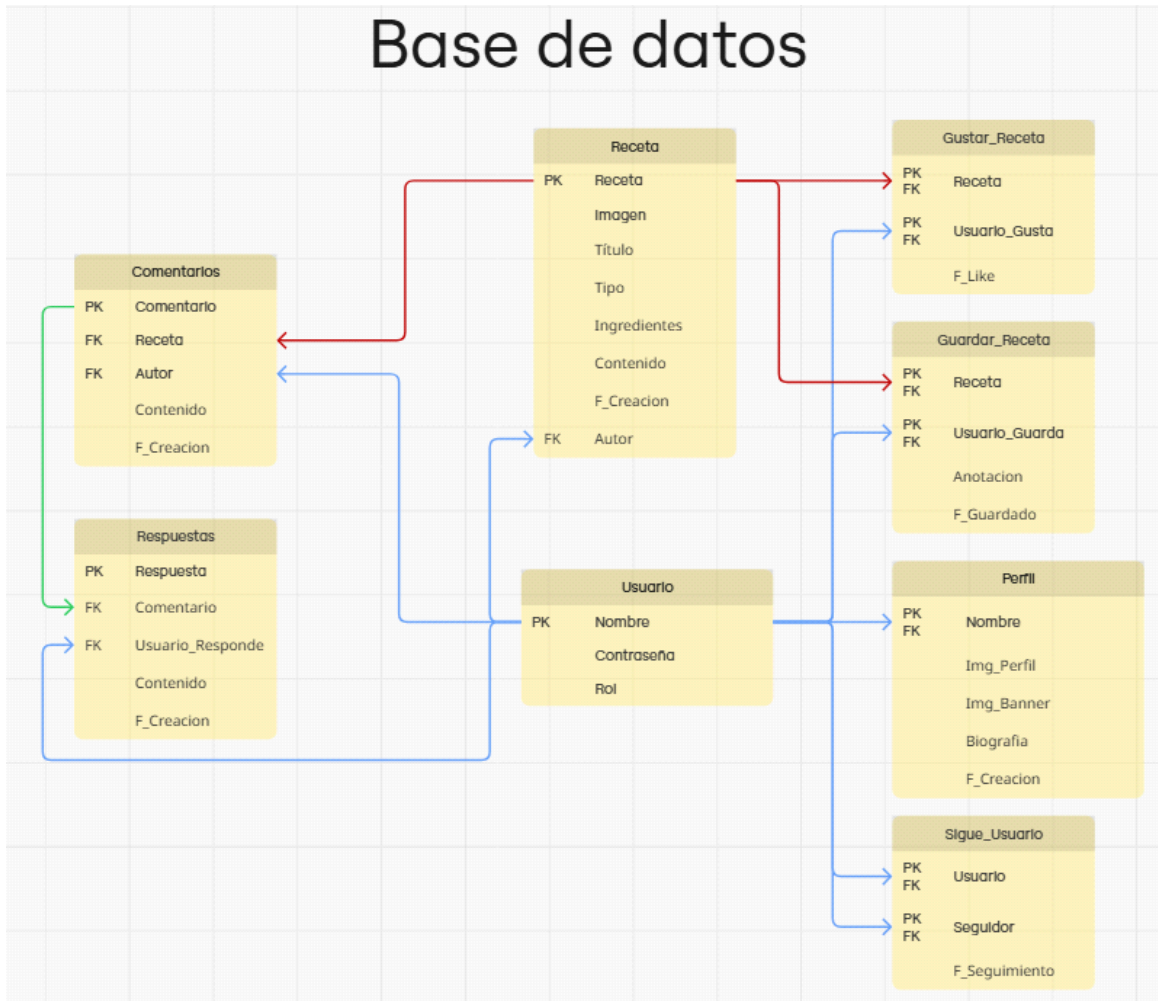


En nuestro TFG hemos utilizado Miro principalmente para planificar el proyecto, organizar ideas y definir estructuras de la aplicación como diagramas de flujo o esquemas de navegación. Su interfaz intuitiva y sus múltiples plantillas nos han permitido coordinar mejor el trabajo en grupo, clarificar tareas y tener una visión global del desarrollo.

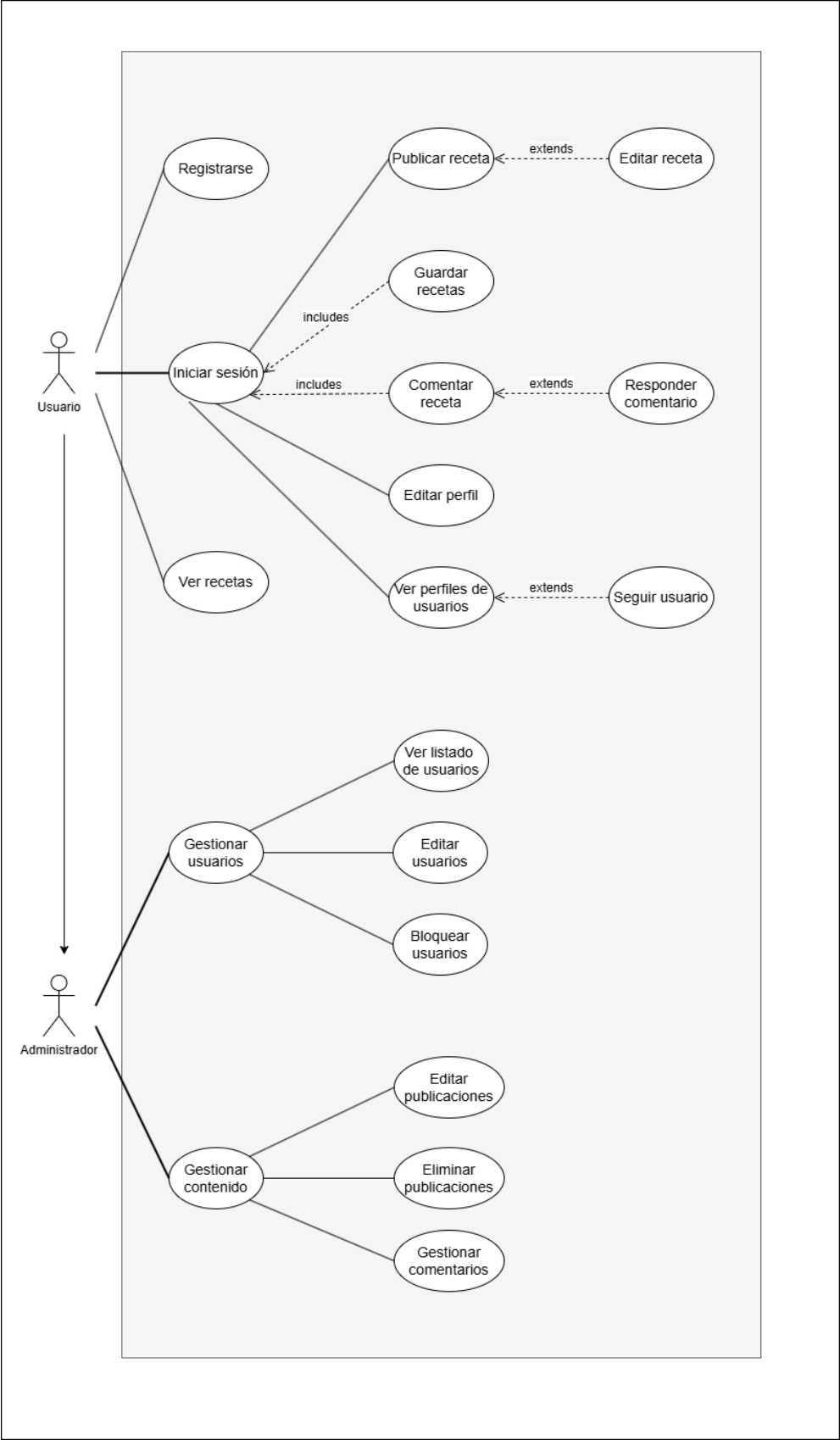
5.4. Diagrama de Entidad-Relación



Base de datos



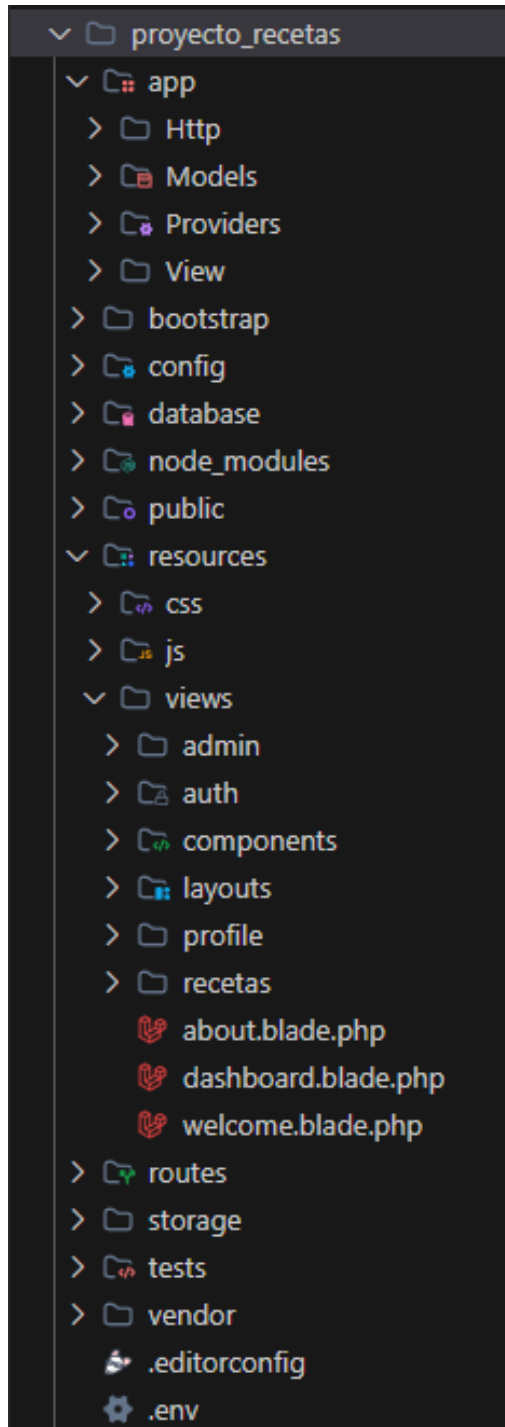
5.5. Diagrama de Casos de Usos



6. MARCO PRÁCTICO

Estructura

La estructura de carpetas y ficheros de este proyecto es la que trae Laravel en su versión 12 por defecto, como se muestra y se explica a continuación:



- Raíz del Proyecto (/)

Aquí se encuentran las carpetas y otros archivos importantes como:

- .env → Configuración del entorno (base de datos, mail, claves API...).
- artisan → Script de consola para ejecutar comandos Laravel.
- composer.json → Gestión de dependencias PHP.
- package.json → Dependencias y scripts del frontend.

- app/

Contiene el núcleo de la aplicación (la lógica de negocio). Aquí se agrupan:

- Http/Controllers/ → Controladores de la lógica web (patrón MVC).
- Models/ → Modelos que representan entidades y gestionan los datos de la base de datos.
- Http/Middleware/ → Filtros que se ejecutan antes o después de procesar una petición HTTP.
- Providers/ → Servicios que configuran e inyectan funcionalidades en la aplicación (por ejemplo, registro de servicios).

- bootstrap/

Contiene el archivo app.php, que se ejecuta al iniciar Laravel. Se usa para cargar y configurar el framework. También incluye la carpeta cache/ para el almacenamiento de archivos cacheados que mejoran el rendimiento.

- config/

Contiene archivos .php de configuración para todos los componentes de Laravel: base de datos (database.php), correo (mail.php), sesiones (session.php), etc. Permite modificar fácilmente el comportamiento de la aplicación.

- database/

Contiene todo lo relacionado con la base de datos:

- migrations/ → Archivos para crear/modificar la estructura de la base de datos.
- seeders/ → Archivos que insertan datos de prueba o iniciales.
- factories/ → Generadores de datos de prueba para testing.

- public/

Carpeta accesible públicamente desde el navegador. Contiene:

- index.php → Punto de entrada a la aplicación.
- Archivos públicos como imágenes, hojas de estilo (CSS), scripts (JS), etc.

- resources/

Contiene los archivos "front-end" de la aplicación:

- views/ → Archivos Blade (.blade.php) que definen la interfaz del usuario.
- lang/ → Archivos de traducción para aplicaciones multilenguaje.
- css/y js/ → Archivos fuente que serán compilados para el frontend (opcional, si se usa Laravel Mix/Vite).

- routes/

Define las rutas de la aplicación. Laravel organiza las rutas por contexto:

- web.php → Rutas para la web (HTML, vistas Blade).
- api.php → Rutas para APIs REST.
- console.php → Comandos de consola (Artisan).
- channels.php → Rutas para canales de broadcasting (WebSockets).

- storage/

Almacena archivos generados o utilizados por la aplicación:

- logs/ → Archivos de registro de errores.

- `app/` → Archivos del usuario (como subidas).
- `framework/` → Archivos temporales de cache, sesiones, vistas compiladas, etc.

Esta carpeta no debe ser accesible desde el navegador, excepto el subdirectorio `storage/app/public` si se enlaza correctamente ejecutando el comando `php artisan storage:link`.

- `tests/`

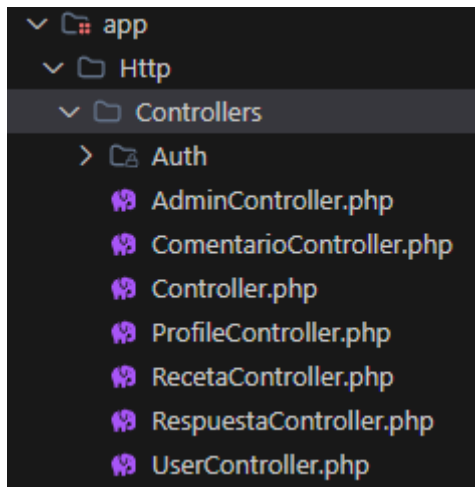
Contiene los tests automatizados:

- `Feature/` → Pruebas de alto nivel de funcionalidades completas.
- `Unit/` → Pruebas unitarias de clases específicas.

Laravel usa PHPUnit por defecto, pero puede integrarse con Pest u otros frameworks.

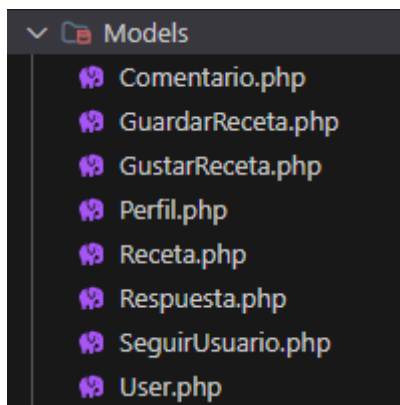
Controladores

Abc



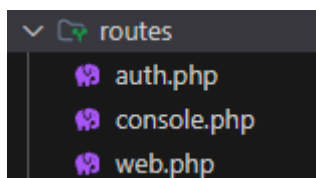
Modelos

Abc



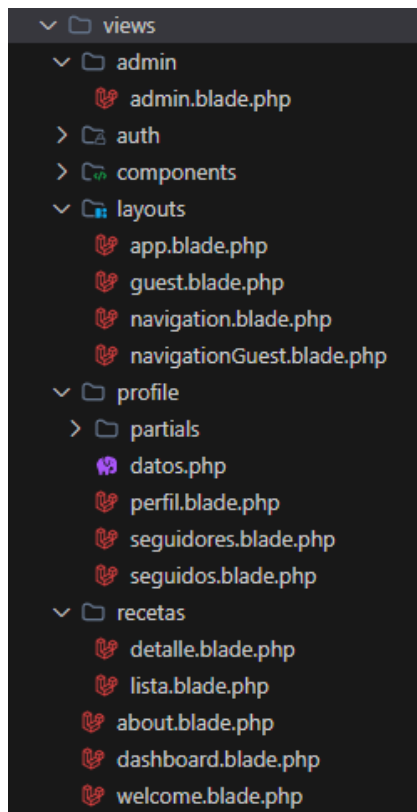
Rutas

Abc



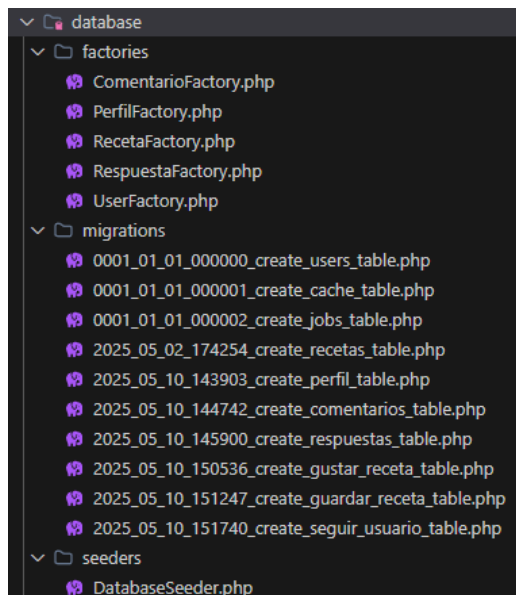
Vistas

Abc

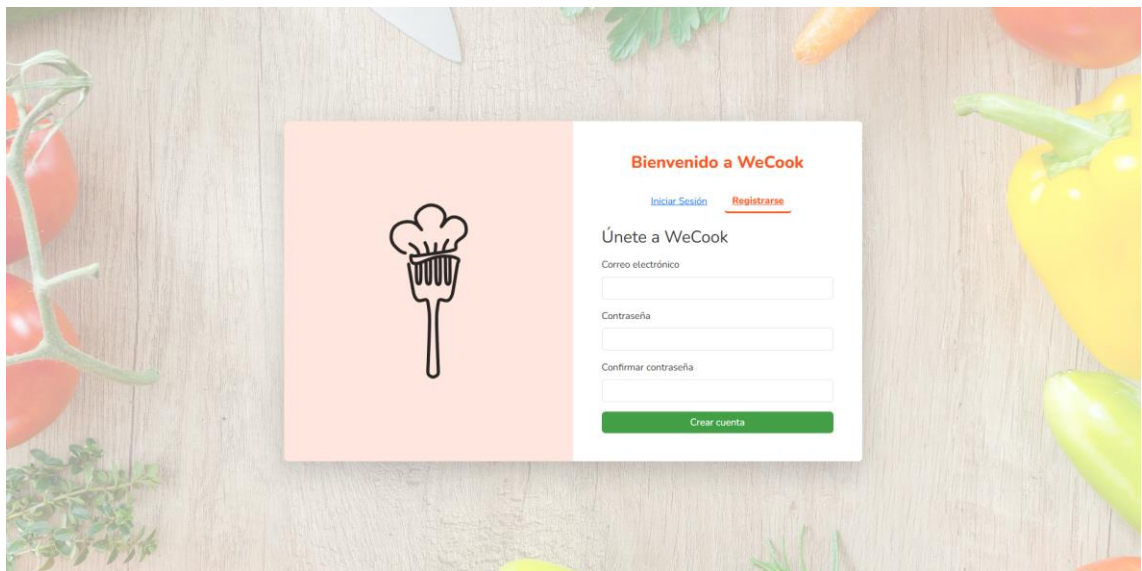
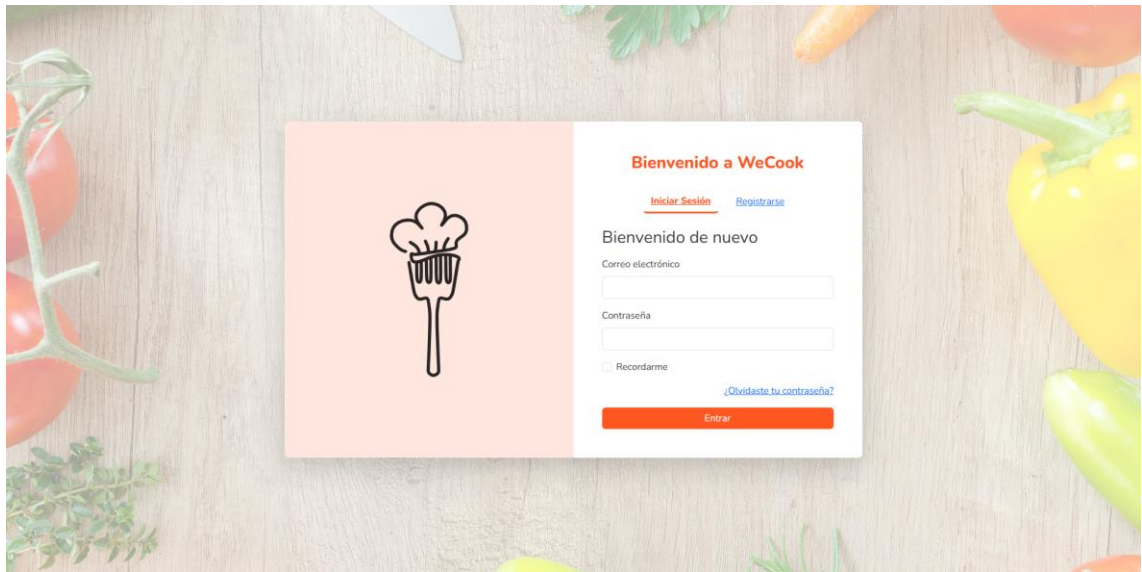


Base de datos


Abc



Vista de inicio



Vista de recordar contraseña:



Forgot your password? No problem. Just let us know your email address and we will email you a password reset link that will allow you to choose a new one.

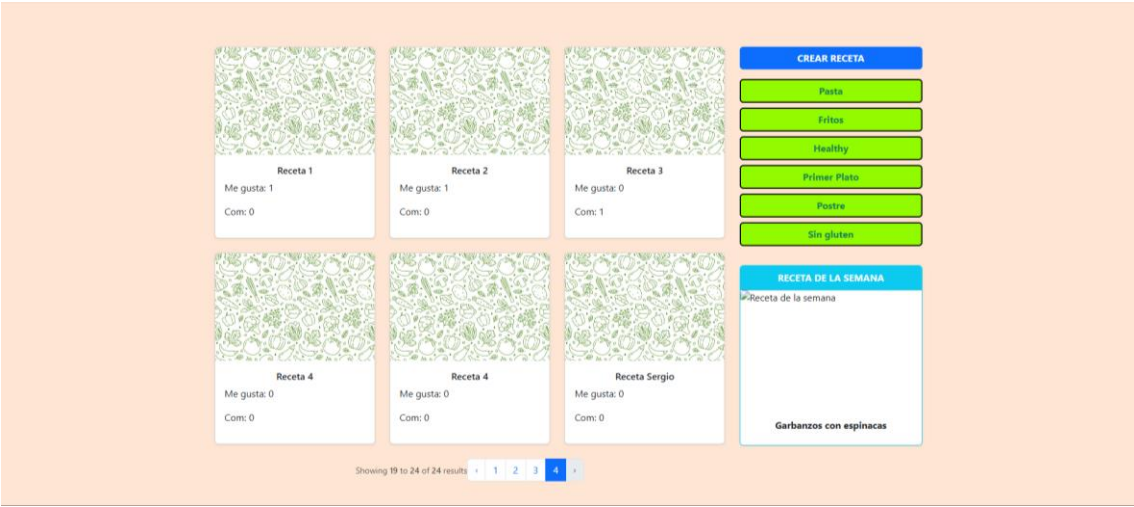
Email

EMAIL PASSWORD RESET LINK

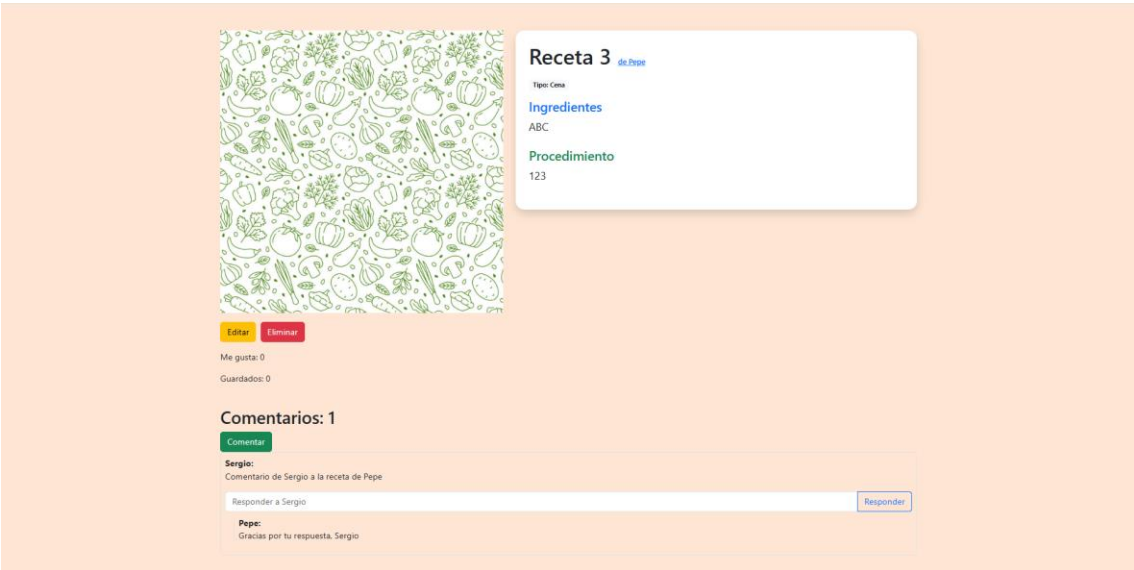
Cabecera común para todas las vistas

Pie común para todas las vistas

Vista de listado de recetas:



Vista de detalle de receta



Vista de creación/edición de receta:

Editar Receta 1

Título:

Receta 1

Imagen:

Seleccionar archivo

Ningún archivo seleccionado



Ingredientes:

ABC

Procedimiento:

123


Tipo:

Comida

Actualizar

Cancelar

Vista de perfil de usuario



@pepe

Biografía de Pepe


[Ver seguidores](#)
[Ver seguidos](#)
[Ver recetas guardadas](#)

1 Seguidores

0 Seguidos


[Editar perfil](#)

Recetas publicadas | [Me gustas](#)



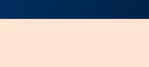
Receta 1

Publicado por 11




Receta 2

Publicado por 11



Receta 3

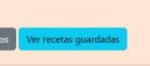
Publicado por 11



Receta 4

Publicado por 11

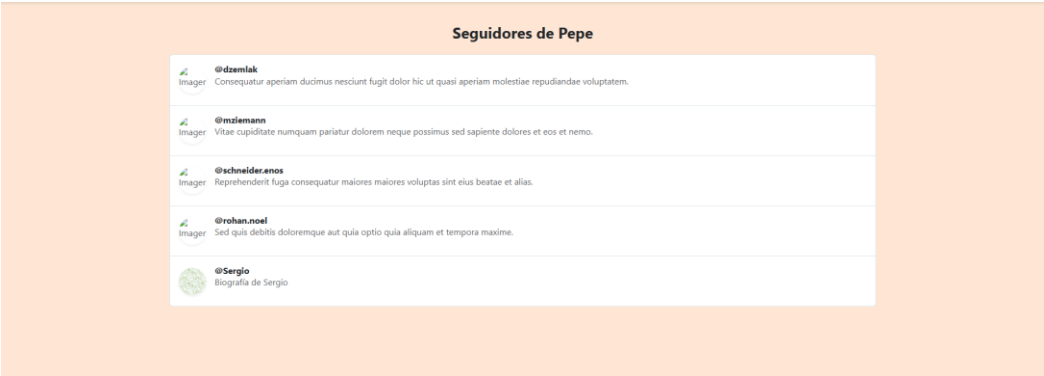
Recetas publicadas | **Me gustas**



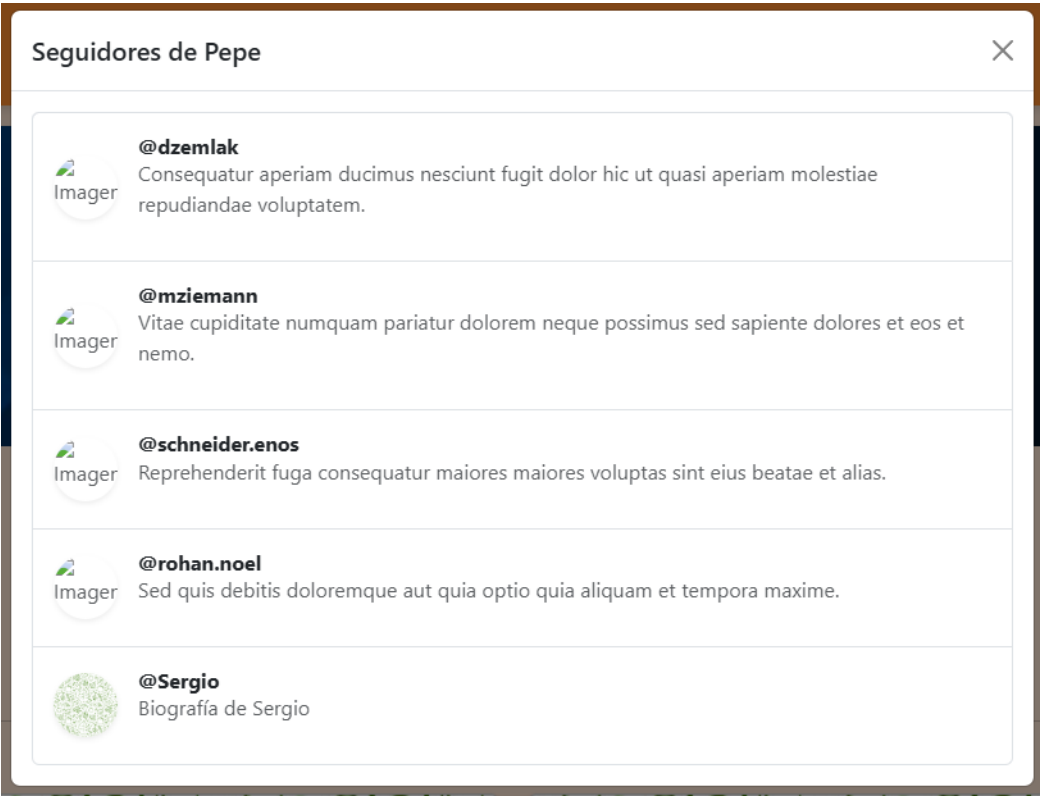
Receta Sergio

Publicado por 12

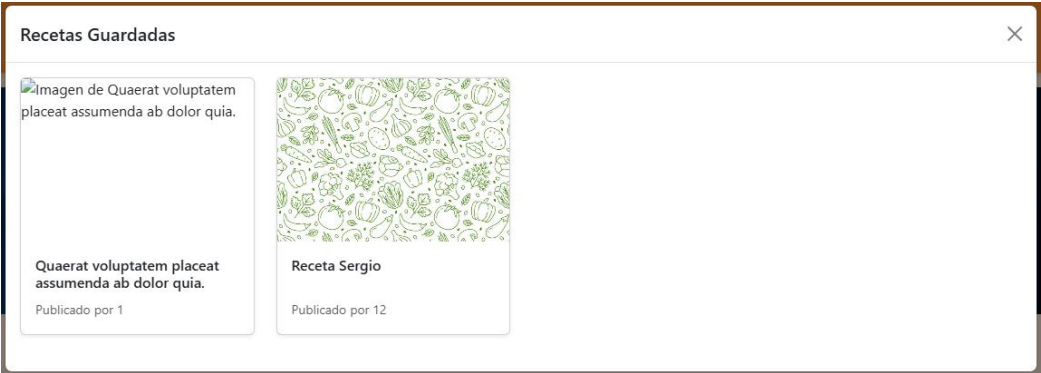
Vista de seguidores/seguidos del usuario



Modal de seguidores/seguidos



Modal recetas guardadas



Modal de editar receta

Edición de perfil

Nombre

Pepe

Descripción

Biografía de Pepe

Imagen de perfil actual

Nueva imagen de perfil

Seleccionar archivo Ningún archivo seleccionado

Banner actual

Nuevo banner

Seleccionar archivo Ningún archivo seleccionado

Cancelar Guardar cambios

Vista de administrador de recetas

Recetas

Recetas

Usuarios

Id Alias @Username F_Creación Filtrar

ID	Nombre	Tipo	Autor	Fecha de creación	Acciones	
1	Quaerat voluptatem placeat assumenda ab dolor quia.	Cena	runte.verda@example.com	N/D	Editar	Eliminar
2	Eveniet facilis quis sed quos et enim.	Postre	runte.verda@example.com	N/D	Editar	Eliminar
3	Dolorem illo eos atque unde.	Postre	runte.verda@example.com	N/D	Editar	Eliminar
4	Aut qui error ratione ab.	Desayuno	runte.verda@example.com	N/D	Editar	Eliminar
5	Sunt minus ullam culpa et ipsam mollitia.	Desayuno	runte.verda@example.com	N/D	Editar	Eliminar

Nueva receta

Showing 1 to 5 of 24 results < 1 2 3 4 5 >

Vista de administrador de usuarios

Recetas

Usuarios

Usuarios

Id

Alias

@Username

F_Creación

Filtrar

ID	Email	Rol	Fecha de registro	Acciones
1	runte.verda@example.com	0	2025-05-22 07:43:26	<div>Eliminar</div>
2	myah93@example.org	0	2025-05-22 07:43:26	<div>Eliminar</div>
3	maxine14@example.com	0	2025-05-22 07:43:26	<div>Eliminar</div>
4	darby31@example.net	0	2025-05-22 07:43:26	<div>Eliminar</div>
5	beer.ila@example.org	0	2025-05-22 07:43:26	<div>Eliminar</div>

Showing 1 to 5 of 12 results

<

1

2

3

>

7. CONCLUSIONES

La realización de este proyecto ha supuesto un gran reto y una excelente oportunidad de aprendizaje. Laravel, aunque completamente nuevo para nosotros al comienzo del desarrollo, ha demostrado ser una herramienta muy útil, bien documentada y con una comunidad activa que facilita el aprendizaje.

Gracias a su estructura basada en el patrón MVC (el cual ya conocíamos) y sus numerosas funcionalidades integradas, hemos podido comprender mejor cómo se construye una aplicación web profesional desde cero, siendo además un trabajo más cómodo de realizar que con PHP simple.

También he mejorado mis conocimientos de bases de datos relacionales (MySQL), seguridad en aplicaciones web y diseño de interfaces. En definitiva, este proyecto nos ha permitido aplicar conocimientos teóricos adquiridos a lo largo del grado y desarrollar competencias clave para nuestro futuro profesional en el ámbito del desarrollo web.

8. BIBLIOGRAFÍA

No hay bibliografía consultada.

9. WEBGRAFÍA

Bootstrap. (s. f.). <https://getbootstrap.com/>

Laravel. (s. f.). <https://laravel.com/docs/12.x>

MySQL Documentation. (s. f.). <https://dev.mysql.com/doc/>

Stack Overflow. (s. f.). <https://stackoverflow.com/>

W3Schools. (s. f.). <https://www.w3schools.com/>

Composer. (s. f.). <https://getcomposer.org/doc/>

Laravel Breeze. (s. f.). <https://laravel.com/docs/12.x/starter-kits#breeze>

Font Awesome. (s. f.). <https://fontawesome.com/docs>

Google Fonts. (s. f.). <https://fonts.google.com/>