

Diseño de una aplicación web de recetas que ayuda a gestionar la alimentación diaria

Memoria de Proyecto Final de Máster

Máster Universitario en Desarrollo de sitios y aplicaciones web

Autor: Antonio Jesús Rodas Rueda

Consultor: Carlos Caballero González

Profesor: César Pablo Córcles Briongos

28 de Septiembre de 2021

Créditos/Copyright

Licencias



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada

[3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

Abstract

El objetivo de este proyecto es el desarrollo y documentación de una aplicación web que sea capaz de responder a una pregunta muy común entre todos: **¿Qué hago hoy de comer?** A partir de una serie de datos recogidos, como recetas consumidas en los últimos días o ingredientes disponibles en un momento determinado, la aplicación podrá recomendar una lista de recetas entre las que el usuario podrá filtrar la que le apetece o puede preparar con los recursos de los que dispone.

Palabras clave: Comer, Recetas, Alimentos, Ingredientes

Abstract (English versión)

The aim of this project is the development and documentation of a web application that can answer a very common question: What should I eat today? Based on a series of data collected, such as recipes consumed in the last few days or ingredients available at a given time, the application will be able to recommend a list of recipes among which the user can filter the one they want or can prepare with the resources available to them.

Keywords: Eat, Recipes, Food, Ingredients

Índice

1. Introducción	13
1.1. Contexto y justificación del trabajo	13
2. Objetivos	15
2.1. Objetivos principales	15
2.2. Objetivos secundarios	16
2.3. Enfoque y método seguido	16
3. Planificación del trabajo	17
3.1. Etapas del proyecto	17
3.2. Diagrama de Gantt	18
4. Análisis de mercado y viabilidad	21
4.1. Mercado	21
4.2. Viabilidad	21
5. Descripción del proyecto	23
5.1. Funciones y características	23
5.2. Usuarios destino	23
6. Catálogo de requisitos	25
6.1. Requisitos funcionales	25
6.2. Requisitos no funcionales	26
7. Análisis del sistema	27
7.1. Casos de uso	27
7.2. Diagrama de casos de uso	32
7.3. Modelo conceptual de datos	33
7.4. Arquitectura de la aplicación	35
8. Plataforma de desarrollo	37
8.1. Entorno de desarrollo	37
8.1.1. Eclipse	37
8.1.2. Visual Studio Code	37
8.1.3. Maven	37
8.1.4. Git	37
8.1.5. Apache Tomcat	37
8.1.6. Base de datos	38
8.2. Lenguajes y librerías	38

8.2.1. Backend.....	38
8.2.2. Frontend	38
8.3. Hardware.....	38
9. Prototipos.....	39
9.1. Diseños de escritorio.....	39
9.2. Diseños en dispositivos móviles	48
10. Usabilidad/UX	55
11. Instrucciones de instalación	57
11.1. Construir paquete.....	57
11.1.1. Construir <i>frontend</i>	57
11.1.2. Construir <i>backend</i>	57
11.1.3. Base de datos	58
11.1.4. Despliegue local	58
11.1.5. Despliegue en <i>heroku</i>	58
12. Proyección a futuro	63
12.1. Registro e inicio de sesión con redes sociales	63
12.2. Notificaciones <i>push</i>	63
12.3. Añadir videos o galería de imágenes a las recetas	63
12.4. Exportación de recetas a pdf	64
12.5. Sistema más complejo de valoración de recetas	64
12.6. Estudio en profundidad de la alimentación	64
12.7. Compra de ingredientes.....	64
13. Conclusiones	65
Anexo 1. Entregables del proyecto.....	67
Anexo 2. Librerías externas.....	69
<i>Bootstrap</i>	69
<i>Animate css</i>	69
<i>@kolkov/angular-editor</i>	70
<i>ng-circle-progress</i>	70
Anexo 3. Guía de uso	71
Guía de usuarios.....	71
Guía de administradores.....	82
Anexo 4. Bibliografía.....	85

Figuras y tablas

Figura 1: Diagrama de Gantt	19
Figura 2: Diagrama de casos de uso.....	32
Figura 3: Diagrama conceptual de datos.....	33
Figura 4: Arquitectura del sistema.....	36
Figura 5: Página de inicio	40
Figura 6: Lista de ingredientes	41
Figura 7: Lista de recetas	42
Figura 8: Detalle de una receta	43
Figura 9: Formulario de receta	44
Figura 10: Recomendar receta.....	45
Figura 11: Mi alimentación.....	47
Figura 12: Inicio <i>responsive</i>	48
Figura 13: Ingredientes <i>responsive</i>	49
Figura 14: Lista de recetas <i>responsive</i>	50
Figura 15: Detalle de una receta <i>responsive</i>	51
Figura 16: Formulario receta <i>responsive</i>	52
Figura 17: Recomendar receta <i>responsive</i>	53
Figura 18: Mi alimentación <i>responsive</i>	54
Figura 19: Registro <i>heroku</i>	59
Figura 20: Nueva aplicación	59
Figura 21: <i>Add-on</i> de base de datos	60
Figura 22: Elección de repositorio <i>heroku</i>	60
Figura 23: Cliente <i>heroku</i>	61
Figura 24: <i>Bootstrap</i>	69
Figura 25: <i>Animate css</i>	69
Figura 26: Editor texto enriquecido.....	70
Figura 27: <i>ng-circle-progress</i>	70
Figura 28: Inicio de sesión.....	71
Figura 29: Pantalla de registro	72
Figura 30: Pantalla de inicio	72
Figura 31: Buscador inicio	73
Figura 32: Página de recetas	74
Figura 33: Página de detalle de receta.....	74
Figura 34: Página de ingredientes.....	75
Figura 35: Crear nuevo ingrediente.....	76
Figura 36: Página de recetas recomendadas.....	77
Figura 37: Desplegable usuario.....	77

Figura 38: Página mis recetas	78
Figura 39: Formulario de receta	79
Figura 40: Crear ingrediente desde receta	79
Figura 41: Página de mi alimentación	80
Figura 42: Ingredientes últimos siete días	81
Figura 43: Histórico de consumiciones	81
Figura 44: Gestión de usuarios	82
Figura 45: Editar ingredientes	83

1. Introducción

En la actualidad, una de las preocupaciones más comunes entre la sociedad es la alimentación. Muchas personas intentan tener una alimentación lo más variada y sana posible, pero no todos lo consiguen. Una de las causas puede ser la fuerza de voluntad de cada uno, la falta de tiempo o incluso el apartado económico, pero el principal problema es la **imaginación** o el **conocimiento**.

La falta de imaginación o conocimiento de una aceptable variedad de recetas o ingredientes, hace que el objetivo se vuelva algo más mentalmente costoso de llevar, lo que provoca abandonos, a veces, incluso antes de empezar.

En plena era digital, si hay algo que se puede solucionar, es el problema de tener toda la información posible al alcance de la mano y de forma instantánea.

1.1. Contexto y justificación del trabajo

La alimentación ha cambiado mucho conforme han ido pasando los años. Poco a poco se han ido añadiendo muchos más alimentos que provienen de los animales y con altos porcentajes de grasa y azúcares. Esto, provoca que en este siglo, las personas son más propensas a padecer enfermedades como la obesidad, la hipertensión la diabetes, etc.

Esto está provocado por los malos hábitos que se están llevando a cabo en la sociedad actual. Siempre es mucho más fácil consumir comida precocinada, o la denominada “comida basura” que preparar un plato rico en frutas, verduras o pastas.

Lo peor de todo es que esto sigue avanzando para mal. En las generaciones más jóvenes, donde se consume mucho más internet, videos, música, aplicaciones..., la mayoría de anuncios son de bollería industrial, bebidas azucaradas o restaurantes de comida rápida o precocinada, lo que provoca que poco a poco se vayan orientando a consumir cierto tipo de productos que no son del todo buenos pero si son muy fáciles de conseguir.

Y es que, hay cientos de aplicaciones que están desarrolladas para facilitar el obtener ese tipo de alimentos, comida las aplicaciones de los restaurantes de comida rápida o las que ayudan a estos restaurantes a repartir a domicilio estos alimentos.

Con tantas facilidades, se está consiguiendo que, cada vez menos, las personas se preparen su propia comida y se vayan perdiendo ciertas recetas que pasaban de generación en generación, agravando esa orientación a consumir productos preparados, que no son tan sanos, que a la larga pueden provocar

enfermedades y que apagan poco a poco una de las mejores características de este país, como es la gastronomía variada, saludable y de buenísima calidad.

Para evitar esto, sería necesario facilitar el acceso a la información a las recetas saludables, que se puedan almacenar y compartir en una plataforma digital, provocando que sea igual de fácil escoger el camino entre consumir comida preparada y obtener una receta que podamos preparar nosotros mismos con los ingredientes con los que disponemos en ese instante.

2. Objetivos

El objetivo de este proyecto es la implementación de una aplicación que sea capaz de responder a la pregunta que en algún momento todo el mundo se ha hecho: **¿Qué puedo comer hoy?** En la aplicación se podrán almacenar recetas creadas por los usuarios para que puedan compartirse entre sí, almacenando sus ingredientes, su proceso de elaboración e incluso imágenes de cómo quedaría la receta ya elaborada.

Cada usuario, guardará la fecha y hora en la que ha consumido una receta determinada, obteniendo así un historial de recetas consumidas.

Con este historial de recetas, se obtendrá la información necesaria sobre la alimentación que ha llevado a cabo el usuario durante todo el tiempo que ha estado usando correctamente la aplicación.

Estos datos recopilados servirán para elaborar un análisis de los ingredientes que ha consumido un usuario en un tiempo determinado y nos permitirá crear una lista de recetas que se le recomendará consumir al usuario para que pueda llevar una dieta equilibrada y sana.

Esta lista de recomendaciones, podrá ser filtrada por el propio usuario, estableciendo los ingredientes que tiene disponible, los comensales que van a participar, el tiempo del que dispone para preparar la receta o la dificultad de la receta que marca si hay que tener un nivel avanzado o no en lo que a cocinar se refiere.

En cuanto a usabilidad, se pretende desarrollar una aplicación web que sea intuitiva y fácil de usar, para que no discrimine a usuario de cualquier género o edad. Se pretende que tenga un diseño atractivo, una estructura fácil de entender y que su uso sea fluido.

Además, la aplicación tendrá un diseño *responsive*, lo que permitirá que se ejecute en pantallas de dispositivos de todo tipo de tamaños y formatos.

Por último, la aplicación permitirá que sea instalada en cualquier dispositivo, actuando con un navegador integrado y funcionando en cualquier tipo de sistema operativo.

2.1. Objetivos principales

Los objetivos principales que se quieren conseguir con la realización de este proyecto son:

- Poner en práctica los conocimientos adquiridos a lo largo de la realización de cada una de las asignaturas de este máster.
- Obtener experiencia en el desarrollo y documentación de un proyecto realizado de principio a fin.
- Crear un producto que pueda ser usado por el público.

2.2. Objetivos secundarios

Los objetivos secundarios que se quieren conseguir con la realización de este proyecto son:

- Ofrecer una aplicación web que permita almacenar y compartir recetas introducidas por los propios usuarios.
- Que la aplicación sea capaz de elaborar un informe o estadísticas de la alimentación que está llevando a cabo un usuario en un periodo de tiempo determinado.
- Desarrollar una aplicación que sea completamente *responsive*.
- Que la aplicación web, también esté disponible cuando no se disponga de conexión a internet.
- Desarrollar una *Progressive Web App* (PWA).
- Ofrecer un sistema de gestión de usuarios y recetas.

2.3. Enfoque y método seguido

En primer lugar se ha consensuado con el tutor de la asignatura la temática que abordará la aplicación. En este caso una aplicación orientada al tema culinario, de recetas y acerca de llevar una dieta sana y equilibrada.

Una vez elegido la temática, se han elegido las tecnologías con las que se van a desarrollar las distintas partes de las que consta la aplicación, el *backend*, el *frontend* y la base de datos.

Hecho esto, se realizará un estudio de requisitos de todas las funcionalidades que podría realizar una aplicación con esta temática estableciendo tiempos de desarrollo, importancia en la aplicación y dificultad de cada una de las funcionalidades que se quieren implementar.

Teniendo estos datos, se establece un orden de desarrollo en el que se tiene en cuenta las funcionalidades que son las más necesarias de implementar para que la aplicación tenga sentido con cada desarrollo que se vaya acabando.

Una vez realizado todo este análisis, se comenzara con el desarrollo. Cada ciclo de desarrollo implementará una funcionalidad nueva en la aplicación y dejará el producto en un estado en el que se puede usar, evitando inconsistencias o errores no controlados.

3. Planificación del trabajo

La planificación del proyecto, viene condicionada por las entregas de las PEC que se van realizando a lo largo del curso, por lo que las fechas y etapas del proyecto, coincidirán con las fechas en las que se realizan las entregas de estas PEC y el trabajo que conlleva que se realice una buena entrega.

3.1. Etapas del proyecto

A continuación se muestran las fechas de comienzo y final de cada PEC con el trabajo que conlleva realizar la entrega de lo que pide cada enunciado:

- **PEC 1: 15/09/2021 – 28/09/2021**
 - Elección de la temática
 - Elección de tecnologías
 - Consulta con el tutor
 - Definición de funcionalidades
 - Comienzo de la documentación
- **PEC 2: 29/09/2021 – 27/10/2021**
 - Revisión de la documentación
 - Análisis de mercado y viabilidad.
 - Preparación de entorno de trabajo
 - Definición de diagramas y detalles de casos de uso
 - Estudio de usabilidad
 - Estudio de la arquitectura
 - Comienzo del desarrollo
- **PEC 3: 28/10/2021 – 05/12/2021**
 - Revisión de la documentación
 - Desarrollo del backend
 - Gestión de bases de datos
 - Gestión de usuarios
 - Frontend
 - Gestión de recetas
 - Despliegue en entorno de pruebas
 - Video demostración

- **Entrega final: 06/12/2021 – 03/01/2022**
 - Revisión de la documentación
 - Revisión de la aplicación
 - Presentación de la aplicación
 - Presentación en video de la aplicación
 - Realizar autoinforme de evaluación

3.2. Diagrama de Gantt

A continuación se muestra un diagrama de Gantt realizado con la herramienta GanttProject [1] en el que se detalla la planificación de las diferentes fases por las que pasa el proyecto:

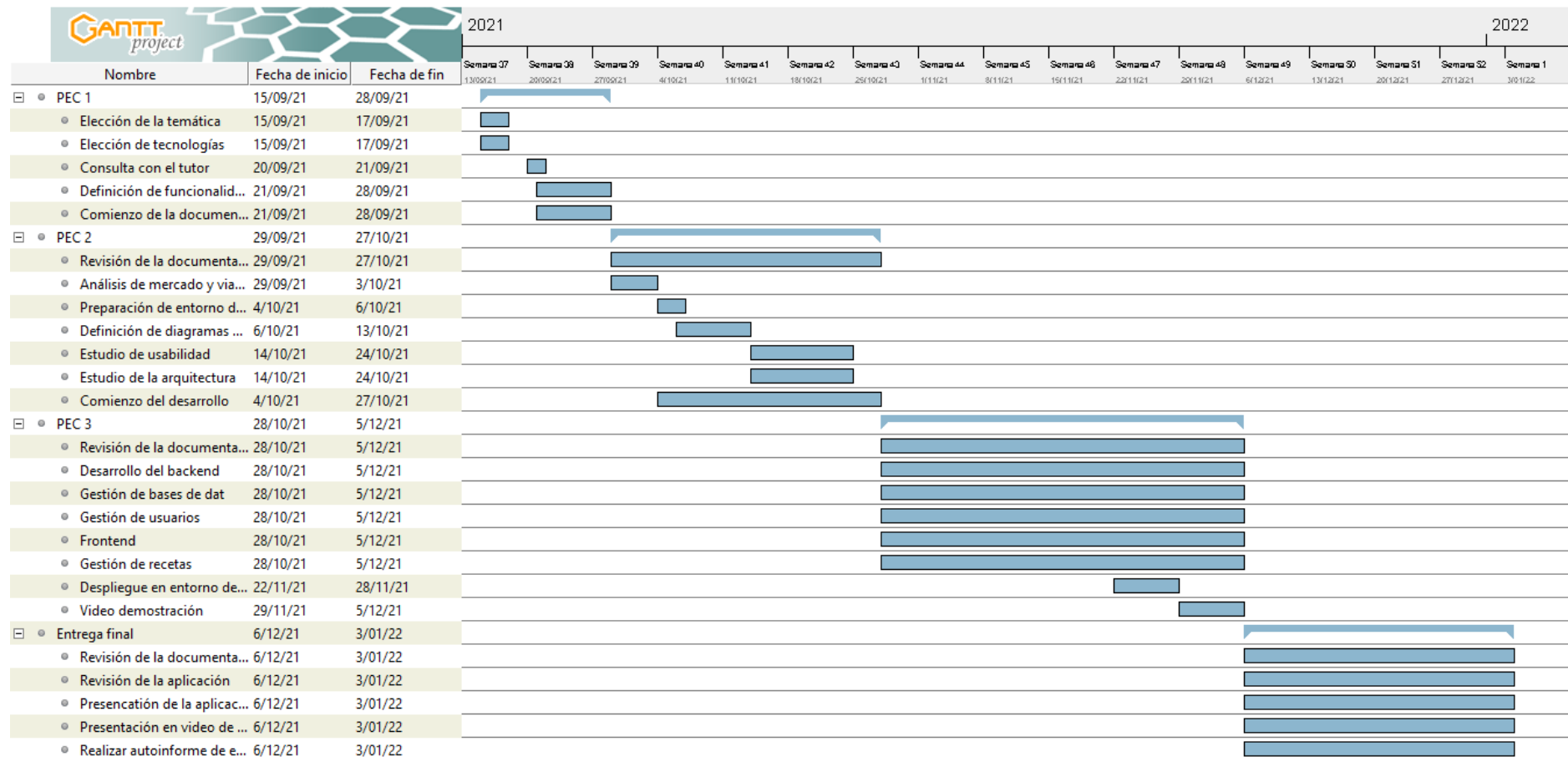


Figura 1: Diagrama de Gantt

4. Análisis de mercado y viabilidad

Cualquier persona que sea mínimamente responsable de llevar a cabo su alimentación o la de las demás personas que habiten con ella, se habrá hecho una pregunta muy formulada en este ámbito: **¿Qué hago de comer hoy?**

Esta pregunta se la hacemos a nuestra pareja, nuestros hijos, nuestros padres... pero, puede que sea más útil, si se la pudiéramos hacer a un sistema que tenga la capacidad de almacenar infinidad de recetas diferentes creadas por multitud de personas, quizás con más creatividad y experiencia que uno mismo, y que sea capaz de recomendarnos exactamente los alimentos que necesitamos.

4.1. Mercado

Buscando aplicaciones sobre recetas de cocina, se encuentran en el mercado numerosas versiones que vienen a abordar el mismo tema, pero no ofrecen las mismas prestaciones que se implementan en este proyecto.

La mayoría de aplicaciones, son simplemente una base de datos en la que vienen predefinidas un número determinado de recetas que están introducidas por los propios administradores y que los usuarios solo pueden consultar y en ocasiones valorar.

Hay algunas aplicaciones que van un poco más allá y permiten a sus usuarios que introduzcan sus propias recetas para tenerlas también almacenadas en la aplicación y que formen parte del recetario que ofrecen.

Y por último, hay aplicaciones que actúan como una red social de recetas, en la que cada usuario cuelga su receta, y se van valorando entre sí, realizando comentarios, o colgando fotos de los platos realizados. Además se puede seguir a otros usuarios y que se notifique cuando alguien a quien sigues cuelgue una nueva receta.

4.2. Viabilidad

De todas las aplicaciones consultadas, no se ha encontrado ninguna que sea capaz de almacenar infinidad de recetas, compartirlas con los usuarios y además que almacene la información necesaria para realizar un análisis de la dieta que está llevando a cabo un usuario concreto pudiendo recomendar a cada uno de estos usuarios, una lista de recetas personalizada dependiendo de las recetas que haya consumido durante un periodo de tiempo determinado y además que estas recomendaciones vayan cambiando dependiendo del día en el que se consulte este listado.

Además, la mayoría de aplicaciones funcionan para un sistema operativo determinado como Windows, *Android*, o *IOS*. Esto se pretende solucionar desarrollando una aplicación web *responsive*, que se podrá descargar en cualquier dispositivo, y funcionara con un navegador integrado que dará la apariencia de que se está ejecutando una aplicación nativa, por lo que será una aplicación multiplataforma.

5. Descripción del proyecto

A continuación se realizará una descripción de las diferentes características, funcionalidades y requisitos que contiene este proyecto.

5.1. Funciones y características

La aplicación está pensada para que la use cualquier tipo de usuario, y por eso se ha pretendido realizar una interfaz sencilla e intuitiva que no produzca confusión, ya que se pretende llegar tanto a personas acostumbradas a vivir el día a día usando la tecnología, como a los que no han usado nunca una aplicación y se está iniciando en el mundillo solo porque le interesa la temática que se va a tocar en este proyecto.

Las principales funciones que debe cubrir la aplicación serían:

- Almacenar y mostrar datos de los ingredientes que componen una receta como pueden ser: el tipo de ingrediente o la cantidad de veces a la semana que se recomienda su consumo.
- Almacenar datos de recetas, así como el tipo de receta que es (postre, plato, bebida...), y, según los ingredientes que la compongan, hacer una recomendación de las veces por semana que se debería consumir esa receta como máximo.
- Gestionar un sistema de usuarios con su respectivo "Inicio de sesión", "Registro de usuario", "Editar perfil" o recordar y cambiar la contraseña de acceso.
- Realizar una recomendación de una receta según unos parámetros introducidos como: tipo de plato, ingredientes disponibles, tiempo de preparación o comensales.
- Mostrar un listado de ingredientes.
- Mostrar un listado de recetas ordenadas y filtradas por distintos parámetros.
- Mostrar información y estadísticas acerca de la alimentación que lleva a cabo un usuario.

5.2. Usuarios destino

La aplicación está diseñada para que pueda usarla el mayor número de usuarios posible y en el mayor número de dispositivos distintos.

Para ello, se ha realizado un diseño muy fácil e intuitivo basado en rellenar formularios y establecer filtros de búsqueda. Además la aplicación cuenta con una interfaz completamente *responsive* y tiene la característica de ser una *Progressive Web App* (PWA), que permitirá ejecutarla tanto como aplicación de escritorio, como aplicación móvil utilizando un navegador interno, lo que posibilita que no haya problemas de ejecución en dispositivos con distintos sistemas operativos como *Android* o *IOS*.

6. Catálogo de requisitos

A continuación se describen los diferentes requisitos de los que consta la aplicación desarrollada en este proyecto.

6.1. Requisitos funcionales

Los requisitos funcionales que debe satisfacer la aplicación son los siguientes:

- **Registrar un usuario:** La aplicación debe ser capaz de registrar un usuario nuevo, almacenarlo en la base de datos y que el nombre de usuario de todos los usuarios registrados sean únicos.
- **Iniciar sesión:** El sistema debe ser capaz de crear una sesión para cada usuario obteniendo su nombre de usuario o correo electrónico y su contraseña, comprobando que existe un usuario en base de datos con esas credenciales.
- **Cerrar sesión:** El sistema debe ser capaz de eliminar la sesión de un usuario que haya accedido a la aplicación introduciendo sus credenciales.
- **Gestionar contraseña:** El sistema debe ser capaz de ofrecer al usuario el poder cambiar su contraseña, sin haber iniciado sesión, mediante el envío de un email a su dirección de correo electrónico o bien habiendo iniciado sesión cambiándola mediante un formulario.
- **Gestionar datos de perfil:** El sistema debe ser capaz de ofrecer al usuario la posibilidad de cambiar los datos personales con los que se registró en el sistema.
- **Registrar nuevos ingredientes:** El sistema debe ser capaz de almacenar los datos de los ingredientes, como el tipo o las veces a consumir por semana, que forman parte de las recetas.
- **Mostrar lista de ingredientes:** El sistema debe ser capaz de mostrar una lista de los ingredientes almacenados en la base de datos.
- **Registrar nueva receta:** El sistema debe ser capaz de almacenar los datos de las recetas como su título, instrucciones de elaboración o los ingredientes que contiene.
- **Editar recetas:** El sistema debe ser capaz de ofrecer al usuario la posibilidad de editar los datos de las recetas que ya ha introducido así como publicarlas o despublicarlas para que aparezcan o no en la lista de recetas.

- **Mostrar y filtrar lista de recetas:** El sistema debe ser capaz de mostrar una lista de las recetas almacenadas en la base de datos permitiendo que sean filtradas, ordenadas y buscadas.
- **Recomendar una receta:** El sistema será capaz de elaborar una lista de recetas recomendadas dependiendo de las recetas que haya consumido el usuario en un periodo de tiempo determinado, el cual se podrá filtrar por ingredientes, tiempo de elaboración, dificultad o comensales.
- **Información y estadísticas de alimentación:** El sistema debe ser capaz de almacenar y mostrar estadísticas al usuario de que alimentos consume más o qué tipo de ingredientes son los que más le gusta, dependiendo de las recetas que haya marcado como que ha consumido.

6.2. Requisitos no funcionales

Los requisitos no funcionales que debe satisfacer la aplicación son los siguientes:

- **Rendimiento:** La aplicación debe funcionar fluidamente sea cual sea la carga de trabajo y la cantidad de usuarios que la estén usando.
- **Tiempo de respuesta:** El sistema debe contar con unos tiempos de respuesta cortos para que la navegación sea fluida y no dañar la experiencia de usuario.
- **Disponibilidad:** La aplicación debe estar disponible siempre, incluso dependiendo de la conexión a internet. El sistema lleva un sistema de cacheado que permite que muchas de las funcionalidades puedan seguir ejecutándose incluso sin contar con conexión a internet.
- **Seguridad:** La aplicación debe mantener la integridad y seguridad de los datos almacenados en ella.
- **Mantenibilidad:** La aplicación debe ser fácilmente mantenible, necesitando el menor número de modificaciones posibles si el sistema tuviera algún tipo de fallo en su funcionamiento. El *backend* del sistema, está desarrollado en *Java* mediante programación orientada a objetos con clases independientes con poco nivel de acoplamiento, lo que hace más fácil su mantenimiento.
- **Fiabilidad:** El sistema debe ser completamente fiable y tolerante a fallos que se puedan producir, mediante la captura de excepciones e informando al usuario de lo que ocurre en cada momento.
- **Responsive:** El sistema debe tener una interfaz que debe estar completamente preparada para que se pueda ejecutar en cualquier tipo de dispositivo.

7. Análisis del sistema

A continuación se realizara un análisis de los casos de uso del sistema desarrollado en este proyecto.

7.1. Casos de uso

- **Caso de uso:** Registro.
 - **Descripción:** Registro de un nuevo usuario en la aplicación.
 - **Actores:** Usuario.
 - **Resumen:** El usuario introduce sus datos en la aplicación y genera unas credenciales con las que poder iniciar sesión.
 - **Precondiciones:** El usuario no puede tener una sesión iniciada y no puede introducir un nombre de usuario o un email que ya exista en la base de datos.
 - **Postcondiciones:** Se almacenan los datos del usuario en la base de datos, lo que le permite iniciar sesión en la aplicación.
 - **Escenario principal:**
 - El usuario pincha en el botón “Iniciar sesión”.
 - Aparece la pantalla de inicio de sesión en la que hay un botón de registro.
 - El usuario, al no tener usuario, pincha en “Registrarse”.
 - El sistema valida que todos los campos estén correctamente rellenos.
 - Se almacenan los datos del usuario y base de datos.
 - Se redirige a la pantalla de inicio de sesión.
 - **Escenarios alternativos:**
 - El usuario puede cerrar la aplicación en cualquier momento.
 - Algún campo del formulario no pasa la validación y se muestra el mensaje correspondiente.
 - El nombre de usuario ya existe y aparece el mensaje correspondiente.
 - El email ya existe y aparece el mensaje correspondiente.
- **Caso de uso:** Iniciar sesión
 - **Descripción:** El usuario inicia sesión en la aplicación.
 - **Actores:** Usuario.
 - **Resumen:** El usuario inicia sesión en la aplicación usando las credenciales que ha obtenido previamente en el registro.
 - **Precondiciones:** El usuario debe estar registrado previamente.
 - **Postcondiciones:** El usuario inicia sesión y puede acceder a las secciones de la aplicación que su rol le permita.
 - **Escenario principal:**

- El usuario pulsa en “Iniciar sesión”.
 - Aparece la pantalla de inicio de sesión.
 - El usuario introduce sus credenciales.
 - El sistema valida las credenciales.
 - El sistema crea la sesión y redirige a la pantalla inicial.
- **Escenarios alternativos:**
 - El usuario puede cerrar la aplicación en cualquier momento.
 - Algún campo del formulario no pasa las validaciones y se muestra un mensaje.
 - Las credenciales introducidas no son correctas y se muestra un mensaje.
- **Caso de uso: He olvidado la contraseña**
 - **Descripción:** El usuario no puede acceder a la aplicación porque no recuerda su contraseña.
 - **Actores:** Usuario
 - **Resumen:** El usuario no recuerda la contraseña de acceso a la aplicación, por lo que pincha en “He olvidado la contraseña” e introduce su correo electrónico para poder cambiar la contraseña sin tener que haber iniciado sesión.
 - **Precondiciones:** El usuario debe estar registrado y no debe haber iniciado sesión en la aplicación.
 - **Postcondiciones:** El usuario modifica su contraseña y ya puede iniciar sesión con sus nuevas credenciales.
 - **Escenario principal:**
 - El usuario pincha en “Iniciar sesión”, pero no recuerda su contraseña.
 - El usuario pincha en “He olvidado mi contraseña”.
 - Aparece un formulario para introducir el correo electrónico.
 - El usuario introduce el correo electrónico con el que está registrado.
 - El usuario recibe un email con un enlace único a la aplicación.
 - El usuario accede a la aplicación mediante el enlace.
 - Aparece un formulario para cambiar la contraseña.
 - Se actualiza la contraseña en base de datos.
 - **Escenarios alternativos:**
 - El usuario puede cerrar la aplicación en cualquier momento.
 - El usuario no rellena el campo del correo electrónico y no puede pulsar en “Enviar”.
 - El correo electrónico introducido no existe.
 - Las contraseñas introducidas en el formulario no son iguales y se muestra un mensaje.
- **Caso de uso: Editar perfil**
 - **Descripción:** Editar los datos con los que el usuario se ha registrado en el sistema.
 - **Actores:** Usuario

- **Resumen:** El usuario accede a su perfil y edita los datos personales con los que se ha dado de alta en el sistema.
- **Precondiciones:** El usuario está registrado y con la sesión iniciada en el sistema.
- **Postcondiciones:** El usuario modifica sus datos personales almacenados en el sistema.
- **Escenario principal:**
 - El usuario ha iniciado sesión y pincha en su nombre.
 - Se despliega una serie de opciones entre las que está “Mi perfil”.
 - El usuario pincha en “Mi perfil”.
 - Aparece un formulario relleno con los datos del usuario en la aplicación.
 - El usuario modifica el campo que quiere editar y pincha en “Guardar”.
 - Se almacenan los datos editados del usuario.
- **Escenarios alternativos:**
 - El usuario puede cerrar la aplicación en cualquier momento.
 - Algún campo del formulario no pasa la validación y se muestra un mensaje.
 - El usuario introduce un email que ya existe en el sistema y se muestra un mensaje.
- **Caso de uso:** Introducir ingrediente.
 - **Descripción:** Introducir un nuevo ingrediente en la aplicación.
 - **Actores:** Usuario.
 - **Resumen:** El usuario inserta un nuevo ingrediente en la aplicación para que pueda ser añadido en una receta.
 - **Precondiciones:** El usuario debe tener una sesión activa.
 - **Postcondiciones:** El usuario introduce un nuevo ingrediente y aparece en la lista con los demás ingredientes.
 - **Escenario principal:**
 - El usuario, con la sesión activa, pincha en “Ingredientes”.
 - Aparece una pantalla con un listado de los ingredientes disponibles.
 - El usuario pincha en “Nuevo ingrediente”.
 - Aparece un formulario para introducir los datos del nuevo ingrediente.
 - El usuario introduce los datos del ingrediente y pincha en “Guardar”.
 - Se almacena el nuevo ingrediente.
 - **Segundo escenario principal:**
 - El usuario, con la sesión iniciada, pincha en “Mis Recetas”.
 - Aparece un listado con las recetas creadas por el usuario.
 - El usuario pincha en “Nueva receta”.
 - Aparece un formulario para introducir los datos de la receta.
 - El usuario pincha en “Nuevo ingrediente”.
 - Aparece un panel con un formulario para introducir los datos de un nuevo ingrediente.
 - El usuario rellena los datos del nuevo ingrediente y pulsa en “Guardar”.

- **Tercer escenario principal:**
 - El usuario, con la sesión iniciada, pincha en “Mis Recetas”.
 - Aparece un listado con las recetas creadas por el usuario.
 - El usuario pincha en “Editar receta”.
 - Aparece un formulario con los datos de la receta a modificar.
 - El usuario pincha en “Nuevo ingrediente”.
 - Aparece un panel con un formulario para introducir los datos de un nuevo ingrediente.
 - El usuario rellena los datos del nuevo ingrediente y pulsa en “Guardar”.
- **Escenarios alternativos:**
 - El usuario puede cerrar la aplicación en cualquier momento.
 - Algún campo del formulario no pasa la validación.
- **Caso de uso: Editar ingrediente**
 - **Descripción:** Editar los datos de un ingrediente.
 - **Actores:** Administrador.
 - **Resumen:** El administrador edita los datos de un ingrediente introducido en el sistema.
 - **Precondiciones:** El usuario debe estar registrado y tener rol de administrador.
 - **Postcondiciones:** El administrador edita los datos de un ingrediente existente en el sistema.
 - **Escenario principal:**
 - El administrador pincha en “Ingredientes”.
 - Aparece la lista de ingredientes que hay en el sistema.
 - El administrador pincha en “Editar ingrediente”.
 - Aparece un formulario relleno con los datos del ingrediente.
 - El administrador edita los datos y pincha en “Guardar”.
 - **Escenarios alternativos:**
 - El usuario puede cerrar la aplicación en cualquier momento.
 - El usuario no tiene el rol de administrador y no ve el botón de “Editar expediente”.
 - Algún campo del formulario no pasa la validación y se muestra un mensaje.
- **Caso de uso: Introducir receta.**
 - **Descripción:** Crear una nueva receta.
 - **Actores:** Usuario.
 - **Resumen:** El usuario crea una nueva receta en el sistema.
 - **Precondiciones:** El usuario debe estar registrado y con una sesión iniciada.
 - **Postcondiciones:** El usuario crea una receta que se almacena en el sistema y puede ser visualizada por los demás usuarios.
 - **Escenario principal:**
 - El usuario, con la sesión iniciada, pincha en su nombre.

- Se despliega una serie de opciones entre la que se encuentra “Mis recetas”.
 - El usuario pincha en “Mis recetas”.
 - Aparece una lista con las recetas que han sido creadas por ese usuario.
 - El usuario pincha en “Nueva receta”.
 - Aparece un formulario en el que deberá rellenar todos los datos que contiene una receta, incluidos los ingredientes.
 - El usuario introduce todos los datos y pincha en “Guardar”.
 - **Escenarios alternativos:**
 - El usuario puede cerrar la aplicación en cualquier momento.
 - El usuario no tiene iniciada la sesión.
 - Algún campo del formulario no pasa la validación y se muestra un mensaje.
- **Caso de uso: Editar receta**
 - **Descripción:** Editar los datos de una receta.
 - **Actores:** Usuario.
 - **Resumen:** El usuario edita los datos de una receta introducida en el sistema.
 - **Precondiciones:** El usuario debe estar registrado, tener una sesión iniciada y al menos haber creado una receta.
 - **Postcondiciones:** El usuario edita los datos de una de las recetas que ha creado anteriormente y los guarda en el sistema.
 - **Escenario principal:**
 - El usuario, con la sesión iniciada, pincha en su nombre.
 - Se despliega una serie de opciones entre la que se encuentra “Mis recetas”.
 - El usuario pincha en “Mis recetas”.
 - Aparece una lista con las recetas que han sido creadas por ese usuario.
 - El usuario pincha en “Editar receta” en una de las recetas de la lista.
 - Aparece un formulario relleno con los datos actuales de esa receta.
 - El usuario modifica los datos pertinentes y pincha en “Guardar”.
 - **Escenarios alternativos:**
 - El usuario puede cerrar la aplicación en cualquier momento.
 - El usuario no tiene iniciada la sesión y no puede acceder a sus recetas.
 - Algún campo del formulario no pasa la validación y se muestra un mensaje.
- **Caso de uso: Recomendar receta**
 - **Descripción:** Recomienda un listado de recetas que consumir.
 - **Actores:** Usuario, Sistema.
 - **Resumen:** El usuario accede a un listado de recetas recomendadas y filtra por los parámetros deseados para obtener una receta recomendada que preparar.
 - **Precondiciones:** El usuario debe estar registrado y con una sesión iniciada.

- **Postcondiciones:** El sistema recomienda al usuario una receta que consumir según los parámetros que el usuario ha introducido.
- **Escenario principal:**
 - El usuario, con la sesión iniciada, “¿Que como hoy?”.
 - Aparece una pantalla con un listado de recetas recomendadas, según los datos almacenados en la aplicación sobre ese usuario, y con un formulario en el que el usuario deberá introducir los parámetros para la recomendación.
 - El usuario rellena el formulario y pincha en “Filtrar”.
 - El sistema hace una búsqueda de recetas que cumplan con los parámetros introducidos y las muestra.
 - Aparece el listado de recetas recomendadas filtradas por los parámetros que ha introducido el usuario.
- **Escenarios alternativos:**
 - El usuario puede cerrar la aplicación en cualquier momento.
 - El usuario no tiene iniciada la sesión y no puede ejecutar esta acción.
 - Algún campo del formulario no pasa la validación y se muestra un mensaje.
 - No existe ninguna receta que cumpla con los parámetros establecidos y no se muestra.

7.2. Diagrama de casos de uso

A continuación se muestra el diagrama de casos de uso, en el que se puede ver los diferentes roles que puede tener un usuario y como puede interactuar con el sistema.

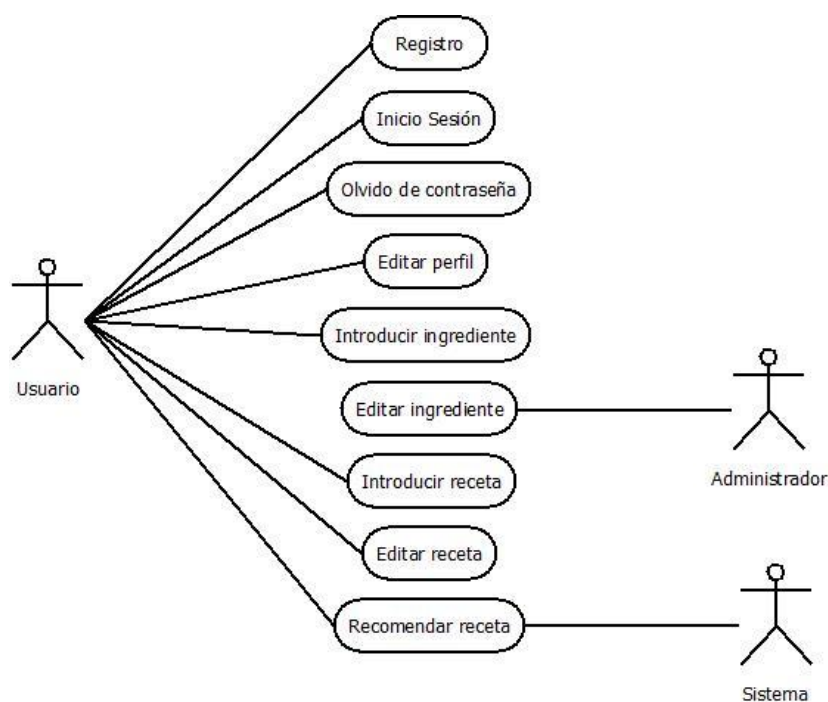


Figura 2: Diagrama de casos de uso

7.3. Modelo conceptual de datos

A continuación se muestra el modelo conceptual de datos del sistema. Es un modelo de datos relacional en el que se puede ver como se relacionan las distintas entidades entre si y los atributos que contiene cada una de ellas.

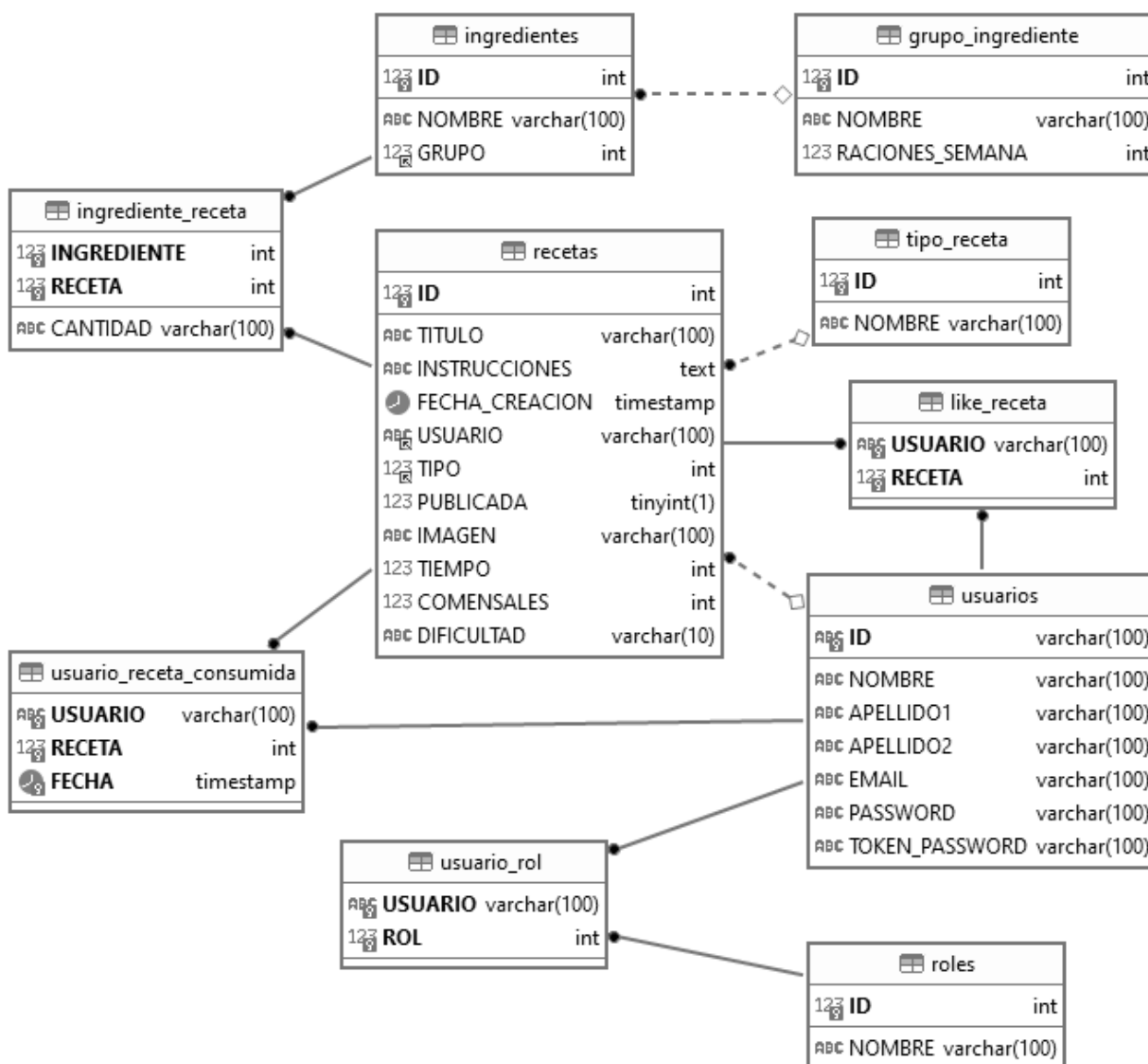


Figura 3: Diagrama conceptual de datos

Se pasa a detallar en qué consiste cada entidad y para qué se usan cada uno de sus atributos en el orden en el que se ha llevado a cabo el desarrollo:

- **roles**: Los roles establecen los permisos que va a tener el usuario para acceder a las distintas partes de la aplicación (usuario y administrador). Consta de los siguientes atributos:

- **ID:** Identificador único en base de datos.
- **NOMBRE:** Nombre del rol.
- **usuarios:** Gestiona los datos de los usuarios registrados en la aplicación, así como sus credenciales para iniciar sesión. Contiene los atributos:
 - **ID:** Identificador único en base de datos.
 - **NOMBRE:** Nombre del usuario.
 - **APELLIDO1:** Primer apellido del usuario.
 - **APELLIDO2:** Segundo apellido del usuario.
 - **EMAIL:** Email del usuario.
 - **PASSWORD:** Contraseña usada para acceder a la aplicación.
 - **TOKEN_PASSWORD:** Se usa para enviar un enlace al correo del usuario para posibilitarle el recuperar la contraseña sin tener que iniciar sesión.
- **usuario_rol:** Relaciona a los usuario con los roles. Sus atributos son:
 - **USUARIO:** Referencia al id de un usuario.
 - **ROL:** Referencia al id de un rol.
- **grupo_ingrediente:** Gestiona los distintos grupos de alimentos a los que puede pertenecer un ingrediente. Sus atributos son:
 - **ID:** Identificador único en base de datos.
 - **NOMBRE:** Nombre del grupo de alimentos (Ej.: Carne, Pescado...)
 - **RACIONES_SEMANA:** El número de veces que se recomienda consumir un alimento, que pertenece a un grupo, en una semana.
- **ingredientes:** Son los ingredientes que componen una receta. Sus atributos son:
 - **ID:** Identificador único en base de datos.
 - **NOMBRE:** Nombre del ingrediente.
 - **GRUPO:** Referencia al id del grupo al que pertenece el alimento.
- **tipo_receta:** Son los tipos en los que se puede clasificar una receta (Ej.: Plato, postre, bebida...). Sus atributos son los siguientes:
 - **ID:** Identificador único en base de datos.
 - **NOMBRE:** Nombre del tipo de receta.
- **receta:** Son las recetas, y todos los datos que puede contener una receta, introducidas por los usuarios de la aplicación. Sus atributos son:
 - **ID:** Identificador único en base de datos.
 - **TÍTULO:** El título describe en que consiste la receta.
 - **INSTRUCCIONES:** Instrucciones paso a paso para elaborar la receta.

- **FECHA_CREACION:** Fecha en la que se inserta la receta en base de datos.
 - **USUARIO:** Referencia al id de usuario que ha creado la receta.
 - **TIPO:** Referencia al id de tipo_receta al que pertenece la receta.
 - **PUBLICADA:** Si la receta esta publicada para que la vean los demás usuarios o no.
 - **IMAGEN:** Nombre de la imagen que contiene la receta, que se usará para construir una *url* para mostrar la imagen por pantalla.
 - **TIEMPO:** Tiempo, en minutos, que se tarda en elaborar la receta.
 - **COMENSALES:** Número de comensales para los que está pensada la receta.
 - **DIFICULTAD:** Nivel de dificultad de la elaboración de la receta.
- **ingrediente_receta:** Relaciona una receta con los ingredientes que contiene y un ingrediente con las recetas en las que se puede encontrar. Sus atributos son:
 - **INGREDIENTE:** Referencia al id de un ingrediente.
 - **RECETA:** Referencia al id de una receta.
 - **CANTIDAD:** Es la cantidad de ese ingrediente que se necesita para elaborar la receta referenciada.
- **like_receta:** Relaciona las recetas a las que le ha dado *like* y los usuarios que han dado *like* a una receta. Sus atributos son los siguientes:
 - **USUARIO:** Referencia al id de un usuario.
 - **RECETA:** Referencia al id de una receta.
- **usuario_receta_consumida:** Relaciona a un usuario con una receta y la fecha en la que ha consumido esa receta. Sus atributos son:
 - **USUARIO:** Referencia al id de un usuario.
 - **RECETA:** Referencia al id de una receta.
 - **FECHA:** Día y hora en la que el usuario ha consumido la receta.

7.4. Arquitectura de la aplicación

La aplicación consiste en una aplicación web. En concreto, se desarrolla una *Progressive Web App*, que tiene la particular característica, que puede ser consultada en un dispositivo móvil como si fuera una aplicación nativa. Además debe cumplir otras propiedades que son:

- **Progresiva:** Debe funcionar en todos los navegadores y sistemas operativos en los que se quiera ejecutar.
- **Independiente a la conectividad:** Debe proporcionar un buen rendimiento, independientemente de la calidad de conexión a internet de la que disponga el usuario.
- **Segura:** Solo funciona con *HTTPS*.
- **Estilo app:** Debe tener la apariencia de una aplicación móvil nativa.

- **Adaptable:** Se adapta a cualquier tamaño y resolución de pantalla.
- **Vinculable:** Se puede compartir fácilmente proporcionando su *url*.
- **Instalable:** Debe poder instalarse en los dispositivos de forma sencilla.
- **Actualizada:** Debe estar siempre actualizada al ser una aplicación web que no necesita instalación.

En cuanto a la estructura que forma la aplicación, hablamos de dos partes bien diferenciadas como son el **backend** y el **frontend**.

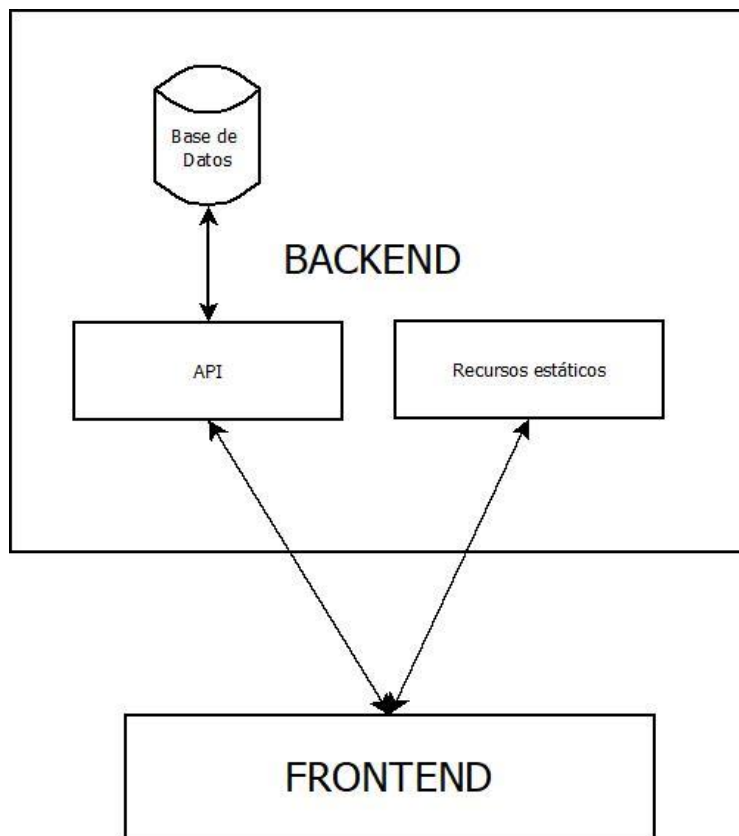


Figura 4: Arquitectura del sistema

Como se puede observar, se distinguen varios elementos en el esquema, pero sobretodo se ve la división entre *backend* y *frontend*. Las funciones de cada parte del esquema son las siguientes:

- **Backend:** Es la parte de la aplicación que se ejecuta en el lado servidor. Se encarga de persistir en base de datos los cambios realizados por los usuarios en la aplicación. También implementa una API de consulta desde donde proporciona los datos que son consumidos por el *frontend*. Además, aloja los recursos estáticos de la aplicación como las imágenes de las recetas.
- **Frontend:** Es la parte de la aplicación que se ejecuta en el lado cliente. Se comunica con la API implementada en el *backend* para obtener la información y mostrarla a los usuarios en el navegador donde estén ejecutando la aplicación, formando así la interfaz de usuario.

8. Plataforma de desarrollo

En esta sección se especifica las diferentes herramientas software y hardware usadas para realizar el desarrollo de este proyecto.

8.1. Entorno de desarrollo

8.1.1. Eclipse

Como entorno de desarrollo para el *backend* se ha usado **Eclipse**. Se ha escogido esta opción debido a la experiencia de uso que se ha obtenido a lo largo de los años usando este software en diferentes proyectos realizados anteriormente. Además **Eclipse** incluye numerosas herramientas y *plugins* que se integran perfectamente con el entorno de desarrollo y facilitan la gestión y realización del proyecto.

8.1.2. Visual Studio Code

Para el desarrollo del *frontend*, se ha optado por usar el editor de texto **Visual Studio Code**, debido a que ha sido el editor que se ha usado a lo largo de la realización del master y se ha obtenido una buena experiencia durante este tiempo.

8.1.3. Maven

Respecto a la obtención de librerías externas, se ha usado **Maven**. La causa es que también se tiene bastante experiencia en su uso y facilita y agiliza la inclusión de librerías en el proyecto

8.1.4. Git

Para el control de versiones, se ha usado **Git**. Se han instalado los plugins correspondientes en *Eclipse* y *Visual Studio Code* para facilitar la obtención y guardado de las distintas versiones del código fuente en **GitHub**.

8.1.5. Apache Tomcat

Como contenedor de *servlets* local, se ha usado **Apache Tomcat**. Se ha usado para desplegar el *backend* de forma local.

8.1.6. Base de datos

Para la base de datos, se ha usado una base de datos relacional **MySql** gestionando los esquemas gráficamente con el software **Dbeaver**.

Para la interacción entre el *backend* y la base de datos se ha usado el *framework* **MyBatis**. Que es un *framework* de persistencia que soporta *SQL* y permite realizar mapeos avanzados sin ser necesario código *JDBC*.

8.2. Lenguajes y librerías

A continuación se detallan los lenguajes, *frameworks* y librerías que se han usado para desarrollar el código fuente de la aplicación.

8.2.1. Backend

Para el desarrollo del código fuente que implementa el *backend* de la aplicación, se ha usado el lenguaje **Java** bajo el *framework* de **Spring**. Se ha optado por esta opción debido a que se tiene experiencia desarrollando software con esta tecnología y facilita mucho el proceso de desarrollo debido a su potencia y flexibilidad.

8.2.2. Frontend

Para el desarrollo del *frontend* se ha optado por usar un *framework* que se ha usado durante el curso como **Angular**. El lenguaje usado por *Angular* es **Typescript**. Un lenguaje de programación libre, mantenido por *Microsoft*, superconjunto de *JavaScript* que agrega tipado estático y que genera código *JavaScript*. Esto nos permite usar las últimas tecnologías sin poner en peligro la compatibilidad con la mayoría de los navegadores.

Para el maquetado se ha usado **HTML** y **CSS**, además de **Bootstrap**. También se ha usado la librería externa de **Fontawesome** para la obtención de iconos.

8.3. Hardware

El hardware que se ha usado para el desarrollo del proyecto ha sido un ordenador portátil, en el cual se han instalado todas las herramientas de desarrollo. Y para comprobar el correcto comportamiento de la aplicación en diferentes dispositivos, se ha usado una *Tablet* y un teléfono móvil.

9. Prototipos

En esta sección se mostrará un primer diseño de la apariencia que se quiere conseguir en el producto final de la aplicación. La aplicación, resumiendo de manera muy básica, consta de la gestión de **Usuarios**, **Recetas** e **Ingredientes**, como se ha especificado en apartados anteriores detallando los casos de uso que resuelve la aplicación. Teniendo los datos de estos tres modelos, se comienza a trabajar para establecer filtros de búsqueda o mostrar estadísticas dependiendo de la relación de todos los datos entre sí.

Como se ha descrito anteriormente, este diseño debe ser adaptable a dispositivos móviles además de que debe ser capaz de ejecutarse en grandes pantalla, por lo que se incluyen a continuación los prototipos en las dos versiones.

9.1. Diseños de escritorio

A continuación se muestra el diseño de las diferentes pantallas por las que pasara el usuario en la versión de escritorio:

La primera página será un resumen de las recetas que se han introducido en la aplicación por parte de los usuarios. Aquí se mostraran las **recetas que se recomiendan hoy**, las **recetas mejor valoradas**, y las últimas recetas introducidas en la aplicación.

Además, esta primera vista, cuenta con un buscador de recetas por título, en el que se ofrecerán sugerencias de títulos de recetas que cumplen con lo que se va escribiendo en el cuadro de texto.

Todas las vistas contarán con un **menú** para navegar por los diferentes apartados de la aplicación y un **footer** que contiene los típicos apartados de **Ayuda**, **Información legal** y **Sobre nosotros**.

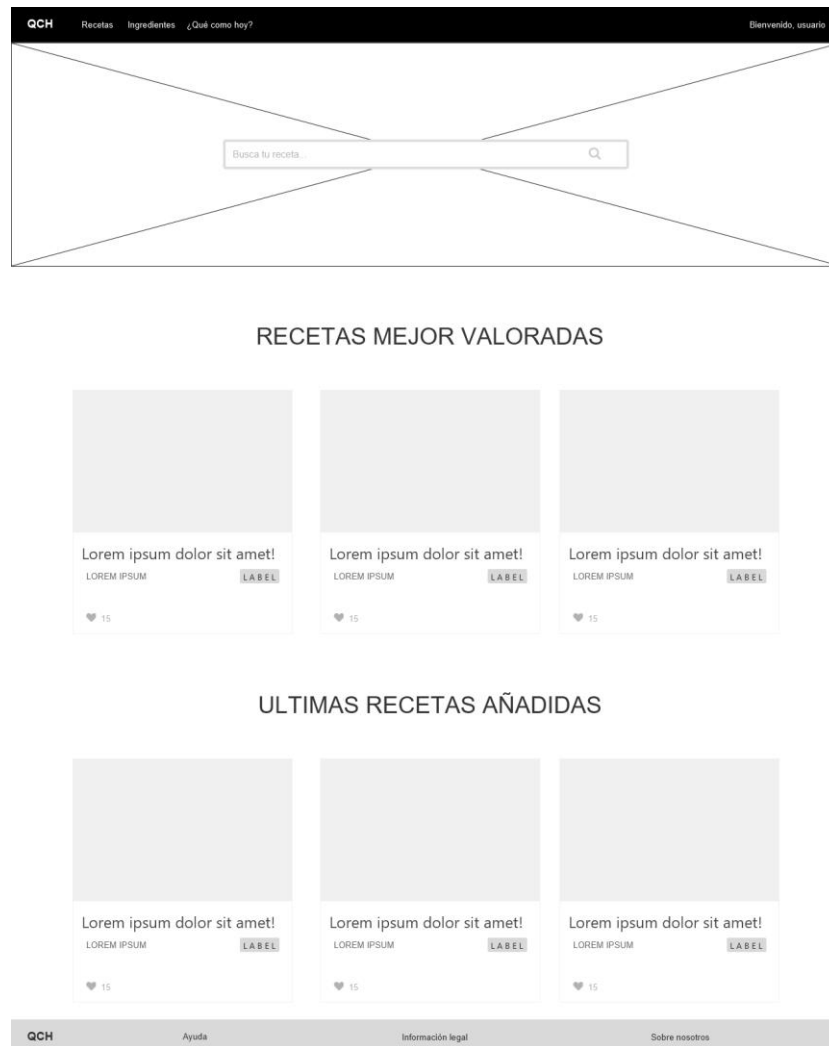


Figura 5: Página de inicio

La siguiente imagen muestra la lista de ingredientes que compondrán las recetas que se van almacenando en la aplicación por los usuarios.

Incluye una serie de campos que van filtrando esta lista de ingredientes para poder facilitar al usuario el encontrar la información de un ingrediente concreto.

Dependiendo del rol del usuario, aparecerán botones de editar o crear nuevos ingredientes.

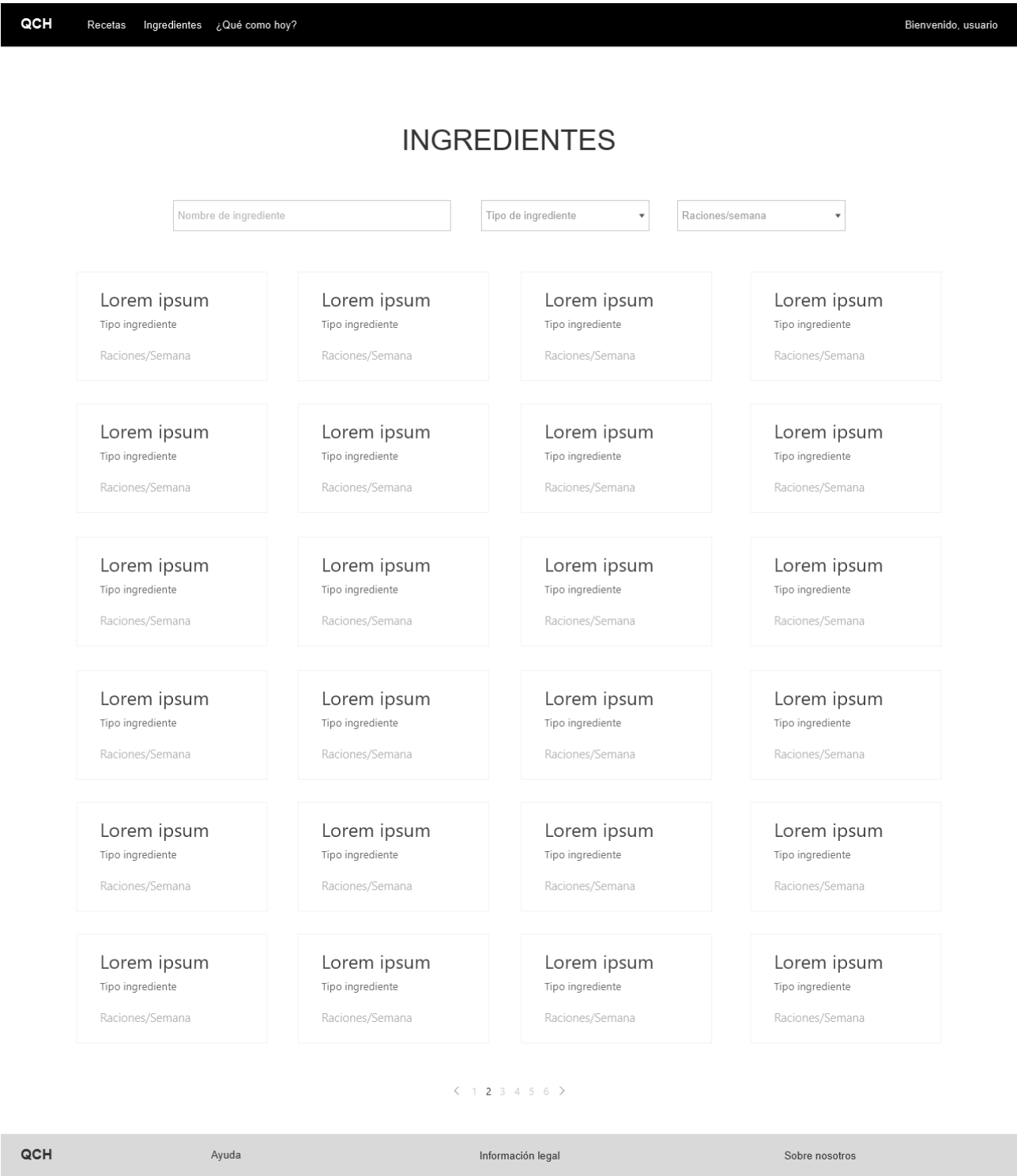


Figura 6: Lista de ingredientes

La siguiente imagen, muestra el listado de recetas que componen la aplicación, aquí se dispondrá de un buscador de recetas con una serie de filtros para obtener las recetas que más interesen. Además, se podrá dar *like* a cada receta, ver cuantos *likes* tiene cada una, y ver información como su imagen, el título, el tiempo de elaboración, los comensales, o la dificultad en la elaboración.

QCH

RecetasIngredientes¿Qué como hoy?

Bienvenido, usuario

RECETAS

Nombre de receta

Tipo de receta

Usuario

Dificultad

Ingredientes

Comensales

Tiempo

RESTABLECER

Ingrediente 1 X

Ingrediente 2 X

Ingrediente 3 X

FILTRAR

Lorem ipsum dolor sit amet!

LOREM IPSUM

LABEL

15

Lorem ipsum dolor sit amet!

LOREM IPSUM

LABEL

15

Lorem ipsum dolor sit amet!

LOREM IPSUM

LABEL

15

Lorem ipsum dolor sit amet!

LOREM IPSUM

LABEL

15

Lorem ipsum dolor sit amet!

LOREM IPSUM

LABEL

15

Lorem ipsum dolor sit amet!

LOREM IPSUM

LABEL

15

Lorem ipsum dolor sit amet!

LOREM IPSUM

LABEL

15

Lorem ipsum dolor sit amet!

LOREM IPSUM

LABEL

15

Lorem ipsum dolor sit amet!

LOREM IPSUM

LABEL

15

<123456>

QCH

Ayuda

Información legal

Sobre nosotros

Figura 7: Lista de recetas

A continuación se muestra el detalle que aparece al acceder a una receta del listado anteriormente expuesto. Aquí podemos ver, además de los detalles anteriormente mostrados en el listado, los ingredientes o las instrucciones de elaboración.

QCH

Recetas

Ingredientes

¿Qué como hoy?

Bienvenido, usuario

Nombre de receta

usuario

Tipo de plato

Ingredientes

- Lorem ipsum dolor sit amet.
- Consectetur adipiscing elit.
- Nunc eget sollicitudin arcu.
- Sed consectetur dolor diam.
- Volutpat est accumsan quis.

Contiene

- Lorem ipsum dolor sit amet.
- Consectetur adipiscing elit.
- Nunc eget sollicitudin arcu.
- Sed consectetur dolor diam.
- Volutpat est accumsan quis.

Recomendado

3 veces a la semana como máximo

✕

✕

✕

12/04/2021

Añadir al calendario

♥

27 me gusta

Instrucciones

Lorem ipsum dolor sit amet, sapien etiam, nunc amet dolor ac odio mauris justo. Luctus arcu, urna praesent at id quisque ac. Arcu es massa vestibulum malesuada, integer vivamus elit eu mauris eus, cum eros quis aliquam wisi. Nulla wisi laoreet suspendisse integer vivamus elit eu mauris hendrerit facilisi, mi mattis pariatur aliquam pharetra eget.

Lorem ipsum dolor sit amet, sapien etiam, nunc amet dolor ac odio mauris justo. Luctus arcu, urna praesent at id quisque ac. Arcu es massa vestibulum malesuada, integer vivamus elit eu mauris eus, cum eros quis aliquam wisi. Nulla wisi laoreet suspendisse integer vivamus elit eu mauris hendrerit facilisi, mi mattis pariatur aliquam pharetra eget.

Lorem ipsum dolor sit amet, sapien etiam, nunc amet dolor ac odio mauris justo. Luctus arcu, urna praesent at id quisque ac. Arcu es massa vestibulum malesuada, integer vivamus elit eu mauris eus, cum eros quis aliquam wisi. Nulla wisi laoreet suspendisse integer vivamus elit eu mauris hendrerit facilisi, mi mattis pariatur aliquam pharetra eget.

Lorem ipsum dolor sit amet, sapien etiam, nunc amet dolor ac odio mauris justo. Luctus arcu, urna praesent at id quisque ac. Arcu es massa vestibulum malesuada, integer vivamus elit eu mauris eus, cum eros quis aliquam wisi. Nulla wisi laoreet suspendisse integer vivamus elit eu mauris hendrerit facilisi, mi mattis pariatur aliquam pharetra eget.

QCH

Ayuda

Información legal

Sobre nosotros

Figura 8: Detalle de una receta

43

La siguiente imagen, muestra el formulario que hay que cumplimentar para introducir una nueva receta o para editar una receta ya existente. Los campos de los formularios tienen sus respectivas validaciones mostrando los mensajes correspondientes en caso de error.

QCH

Recetas

Ingredientes

¿Qué como hoy?

Bienvenido, usuario

NUEVA RECETA

Título

Ingredientes

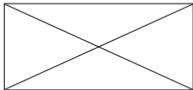
Añadir

Crear nuevo ingrediente

Instrucciones

Imagen

Seleccionar archivo



Tipo de receta

-- Seleccione --

Tiempo de preparación

Comensales

Dificultad

-- Seleccione --

Guardar

Cancelar

QCH

Ayuda

Información legal

Sobre nosotros

Figura 9: Formulario de receta

Una de las funciones de esta aplicación, es recomendar una receta que consumir en un momento determinado, atendiendo a diferentes parámetros que introduciría el usuario como pueden ser los ingredientes o los comensales para los que se quiere preparar la receta.

En la siguiente imagen, se muestra la lista de recetas recomendadas según las recetas que haya consumido el usuario durante un periodo de tiempo determinado. En la pantalla aparece un formulario que permite introducir parámetros para poder filtrar estas recetas y encontrar la receta recomendada adecuada.

QCH

Recetas

Ingredientes

¿Qué como hoy?

Bienvenido, usuario

RECETAS RECOMENDADAS

Nombre de receta

Tipo de receta

Usuario

Dificultad

Ingredientes

Comensales

Tiempo

RESTABLECER

Ingrediente 1

Ingrediente 2

Ingrediente 3

FILTRAR

Lorem ipsum dolor sit amet!
LOREM IPSUM
LABEL

15

Lorem ipsum dolor sit amet!
LOREM IPSUM
LABEL

15

Lorem ipsum dolor sit amet!
LOREM IPSUM
LABEL

15

Lorem ipsum dolor sit amet!
LOREM IPSUM
LABEL

15

Lorem ipsum dolor sit amet!
LOREM IPSUM
LABEL

15

Lorem ipsum dolor sit amet!
LOREM IPSUM
LABEL

15

Lorem ipsum dolor sit amet!
LOREM IPSUM
LABEL

15

Lorem ipsum dolor sit amet!
LOREM IPSUM
LABEL

15

Lorem ipsum dolor sit amet!
LOREM IPSUM
LABEL

15

<

1

2

3

4

5

6

>

QCH

Ayuda

Información legal

Sobre nosotros

Figura 10: Recomendar receta

45

Por último, la aplicación será capaz de elaborar estadísticas, dependiendo de la alimentación que haya registrado el usuario sobre las recetas que ha consumido en el tiempo.

En la siguiente imagen, se muestra como se visualizarían estos datos en la aplicación.

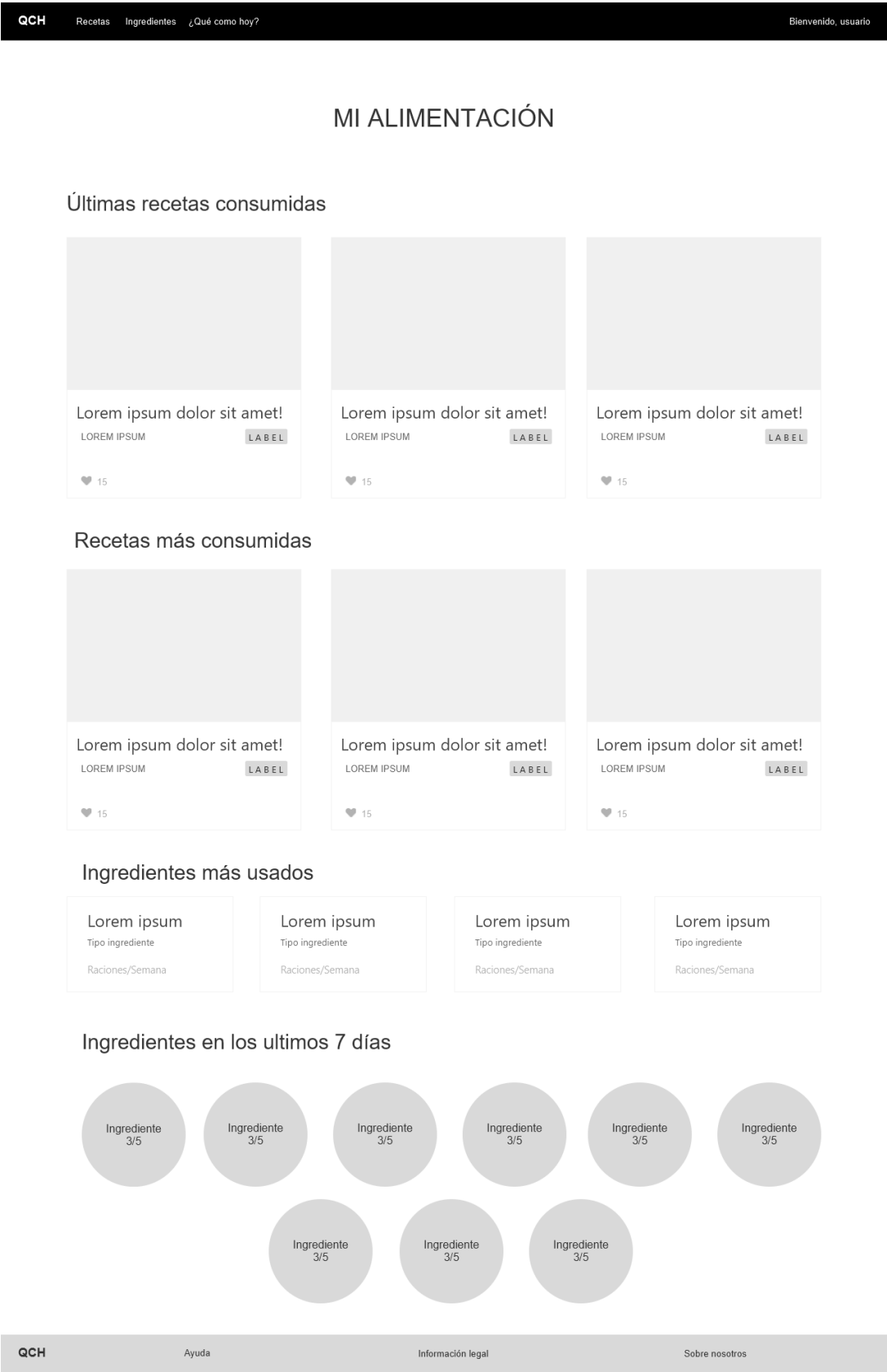


Figura 11: Mi alimentación

Estos son los prototipos de las principales pantallas que formarían la aplicación en la versión de escritorio con pantallas en formato apaisado.

9.2. Diseños en dispositivos móviles

En esta sección se mostrará los diseños *responsive* de las mismas pantallas que se han mostrado en el apartado anterior.

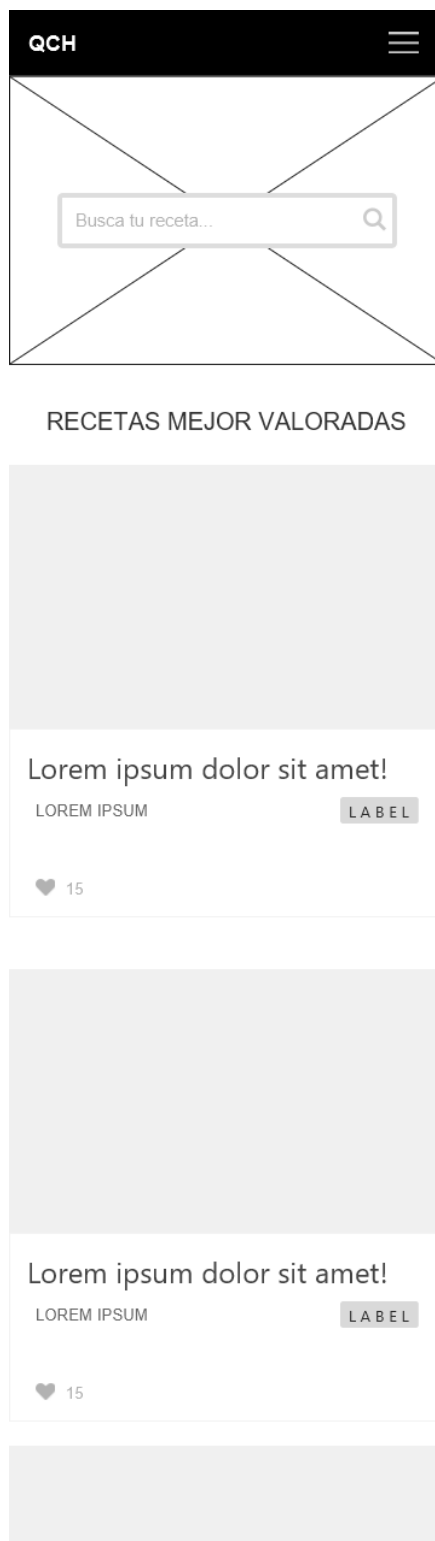


Figura 12: Inicio *responsive*

QCH

INGREDIENTES

Nombre de ingrediente

Tipo de ingrediente

Raciones/semana

Lorem ipsum

Tipo ingrediente

Raciones/Semana

Lorem ipsum

Tipo ingrediente

Raciones/Semana

Lorem ipsum

Tipo ingrediente

Raciones/Semana

Lorem ipsum

Tipo ingrediente

Raciones/Semana

Lorem ipsum

Tipo ingrediente

Raciones/Semana

Lorem ipsum

Tipo ingrediente

Raciones/Semana

Lorem ipsum

Tipo ingrediente

Raciones/Semana

Lorem ipsum

Tipo ingrediente

Raciones/Semana

<

123456

>

QCH

Ayuda

Información legal

Sobre nosotros

Figura 13: Ingredientes responsive

QCH

RECETAS

Nombre de receta

Usuario

Tipo de receta

Dificultad

Tiempo

Comensales

Ingredientes

Ingrediente 1 X

Ingrediente 2 X

Ingrediente 3 X

RESTABLECER

FILTRAR

Lorem ipsum dolor sit amet!

LOREM IPSUM

LABEL

♥

15

Lorem ipsum dolor sit amet!

LOREM IPSUM

LABEL

♥

15

Figura 14: Lista de recetas *responsive*

50

QCH

Nombre de receta

usuario

Tipo de plato

Ingredientes

Lorem ipsum dolor sit amet.

Consectetur adipiscing elit.

Nunc eget sollicitudin arcu.

Sed consectetur dolor diam.

Volutpat est accumsan quis.

Contiene

Lorem ipsum dolor sit amet.

Consectetur adipiscing elit.

Nunc eget sollicitudin arcu.

Sed consectetur dolor diam.

Volutpat est accumsan quis.

Recomendado

3 veces a la semana como máximo

12/04/2021

Añadir al calendario

27 me gusta

Instrucciones

Lorem ipsum dolor sit amet, sapien etiam, nunc amet dolor ac odio mauris justo. Luctus arcu, urna praesent at id quisque ac. Arcu es massa vestibulum malesuada, integer vivamus elit eu mauris eus, cum eros quis aliquam wisi. Nulla wisi laoreet suspendisse integer vivamus elit eu mauris hendrerit facilisi, mi mattis pariatur aliquam pharetra eget.

Lorem ipsum dolor sit amet, sapien etiam, nunc amet dolor ac odio mauris justo. Luctus arcu, urna praesent at id quisque ac. Arcu es massa vestibulum malesuada, integer vivamus elit eu mauris eus, cum eros quis aliquam wisi. Nulla wisi laoreet suspendisse integer vivamus elit eu mauris hendrerit facilisi, mi mattis pariatur aliquam pharetra eget.

Lorem ipsum dolor sit amet, sapien etiam, nunc amet dolor ac odio mauris justo. Luctus arcu, urna praesent at id quisque ac. Arcu es massa vestibulum malesuada, integer vivamus elit eu mauris eus, cum eros quis aliquam wisi. Nulla wisi laoreet suspendisse integer vivamus elit eu mauris hendrerit facilisi, mi mattis pariatur aliquam pharetra eget.

Lorem ipsum dolor sit amet, sapien etiam, nunc amet dolor ac odio mauris justo. Luctus arcu, urna praesent at id quisque ac. Arcu es massa vestibulum malesuada, integer vivamus elit eu mauris eus, cum eros quis aliquam wisi. Nulla wisi laoreet suspendisse integer vivamus elit eu mauris hendrerit facilisi, mi mattis pariatur aliquam pharetra eget.

Lorem ipsum dolor sit amet, sapien etiam, nunc amet dolor ac odio mauris justo. Luctus arcu, urna praesent at id quisque ac. Arcu es massa vestibulum malesuada, integer vivamus elit eu mauris eus, cum eros quis aliquam wisi. Nulla wisi laoreet suspendisse integer vivamus elit eu mauris hendrerit facilisi, mi mattis pariatur aliquam pharetra eget.

QCH

Ayuda

Información legal

Sobre nosotros

Figura 15: Detalle de una receta *responsive*

QCH

NUEVA RECETA

Título

Ingredientes

Añadir

Crear nuevo ingrediente

Instrucciones

Imagen

Seleccionar archivo

Tipo de receta

-- Seleccione --

Tiempo de preparación

Comensales

Dificultad

-- Seleccione --

Guardar

Cancelar

QCH

Ayuda

Información legal

Sobre nosotros

Figura 16: Formulario receta responsive

QCH

RECETAS RECOMENDADAS

Nombre de receta

Usuario

Tipo de receta

Dificultad

Tiempo

Comensales

Ingredientes

Ingrediente 1 X

Ingrediente 2 X

Ingrediente 3 X

RESTABLECER

FILTRAR

Lorem ipsum dolor sit amet!

LOREM IPSUM

LABEL

♥

15

Lorem ipsum dolor sit amet!

LOREM IPSUM

LABEL

♥

15

Figura 17: Recomendar receta *responsive*

53



Figura 18: Mi alimentación *responsive*

10. Usabilidad/UX

A la hora de realizar el diseño de la aplicación, se han tenido en cuenta diversos patrones de diseño que ayudan a que la aplicación tenga una interfaz intuitiva y amigable con el usuario.

Al estar dirigida al mayor número de usuarios distintos, la interfaz ha de contar con un diseño sencillo y que explique claramente lo que se muestra y lo que se puede hacer en cada pantalla.

Así, la estructura de la aplicación también ha de ser una estructura lo más simple posible que permita en todo momento saber en qué lugar de la aplicación está situado el usuario en ese momento, y como puede acceder a la parte de la aplicación en la que puede ejecutar la acción que desea.

Por ello, se han usado diferentes patrones de diseño que facilitan la obtención de estas características que facilitan el uso y proporcionan una experiencia agradable al usuario:

- **HomeLink:** Se muestra en todo momento, en un lugar fijo (arriba a la izquierda) un enlace que lleva al usuario a la página principal de la aplicación.
- **Morphing Controls:** Se muestra al usuario si un botón está disponible para pulsarse o no, mediante cambios en la apariencia.
- **Paginación:** La información a mostrar se muestra en subconjuntos pequeños que hace que el usuario no quede abrumado con toda la información de golpe.
- **Input Feedback:** Se muestra información al usuario de si esta rellenando los datos correctamente o no.
- **Table Filter:** Se ofrece un mecanismo sencillo al usuario de filtrado de los datos que se muestran en pantalla.
- **WYSIWYG:** Se proporciona al usuario la posibilidad de introducir texto enriquecido y formateado.

Estos patrones, también se han tenido en cuenta a la hora de adaptar el diseño a diferentes tamaños y formatos de pantallas, ya que se pretende que esta aplicación sea usada en el mayor número de dispositivos posibles.

Para ellos se ha seguido una estrategia de diseño llamada **Desktop First**. Esto consiste en realizar el diseño para la aplicación ejecutada en pantallas de gran formato. Una vez obtenido el diseño, se ha

adaptado ese resultado obtenido para que se vea de una forma correcta conforme el tamaño de la pantalla va disminuyendo hasta el tamaño de los dispositivos móviles.

11. Instrucciones de instalación

A continuación se especifica el proceso que hay que llevar a cabo para poder ejecutar la aplicación, ya sea en un entorno local o en un servidor web.

La aplicación está dividida en dos partes, cada una desarrollada con dos tecnologías distintas, por lo que habrá que construir ambos proyectos y unirlos en un solo paquete que se desplegará en el servidor correspondiente.

En concreto, una de las partes en las que se divide el proyecto es el **backend**, desarrollado en **java**. Y la otra parte es el **frontend**, que está desarrollado en angular.

11.1. Construir paquete

Para construir el paquete de la aplicación, en primer lugar hay que descargarse todo el código del repositorio **github**: <https://github.com/ajrodasr/TFM-Que-como-hoy>.

Aquí se obtendrán dos carpetas en las que están las dos partes de la aplicación (*backend* y *frontend*). Obtenidos los archivos, en primer lugar construiremos el *frontend*.

11.1.1. Construir *frontend*

Para construir el *frontend*, necesitamos tener instalado **Node.js** con el gestor de paquetes **npm** descargándolo de su web oficial. Una vez instalados, ejecutamos el comando `npm install`, en la carpeta raíz del proyecto, para descargar todas las dependencias que contiene el proyecto.

Por último, también en la carpeta raíz del proyecto, ejecutamos el comando `npm run build -- --prod`, que construirá el proyecto dentro de una carpeta situada dentro del *backend* que ya está configurada.

Con esto, tendríamos todo lo necesario para construir el *backend* y con ello el paquete final.

11.1.2. Construir *backend*

Una vez construido el *frontend*, pasamos a construir la segunda parte del proyecto, el *backend*. Si se ha realizado el apartado anterior, tendremos las dos partes del proyecto en el mismo sistema de carpetas. Por lo que la carpeta del *backend* contendrá el proyecto construido del *frontend* en la carpeta *static/frontend*.

Antes de construir, es necesario configurar el *application.properties* con los parámetros correspondientes a la máquina en la que vayamos a lanzar la aplicación. En este caso, debemos **configurar** la **base de datos** a la que se conectará la aplicación junto con sus credenciales, y el **email** y la **clave** necesarios para enviar correos desde la aplicación a los usuarios.

Hecho esto, para construir esta parte del proyecto, necesitamos tener instalado **maven** en el equipo. Lo descargaremos yendo a la web oficial y siguiendo las instrucciones de instalación.

Una vez descargado, ya podremos construir el paquete del proyecto completo, para ello nos situaremos en la carpeta raíz del *backend* y ejecutaremos el comando *mvn clean package*.

11.1.3. Base de datos

Para que la aplicación pueda ejecutarse, se necesita crear la base de datos en la que almacenará toda la información que se vaya generando.

Para ello, tendremos que crear una base de datos *MySQL*. Esto lo haremos con un *script* que hay creado dentro del sistema de archivos del *backend*. Concretamente se encuentra en *resources/db/migrations/create_database.sql*.

Una vez ejecutado este *script* se creará la base de datos con todas las tablas y datos necesarios para que pueda funcionar la aplicación.

11.1.4. Despliegue local

Una vez construido el paquete, y creada la base de datos, desplegaremos la aplicación en nuestra máquina del entorno local.

Para ello, iremos a la carpeta donde se ha creado el paquete, que normalmente se crea en la carpeta *target*.

Una vez ahí, ejecutaremos el comando *java -jar nombreDelPaquete.jar*, y la aplicación quedará desplegada.

11.1.5. Despliegue en *heroku*

Para que la aplicación tenga un dominio público y poder enfrentarnos a problemas reales en cuanto a poner una aplicación en un entorno de “producción” se refiere, se ha realizado la subida del proyecto a la plataforma **heroku**, que proporciona un servidor limitado que funciona con *Apache Tomcat* y dispone de numerosos *add-ons*, también limitados en cuanto a capacidad debido a que la cuenta creada es gratuita, de distintas tecnologías de bases de datos con los que se puede integrar el proyecto. En nuestro caso **MySQL**.

En primer lugar, ha sido necesario crear una cuenta gratuita en la plataforma, para poder hacer uso de ella. Para ello, simplemente nos registramos y rellenamos los datos correspondientes.

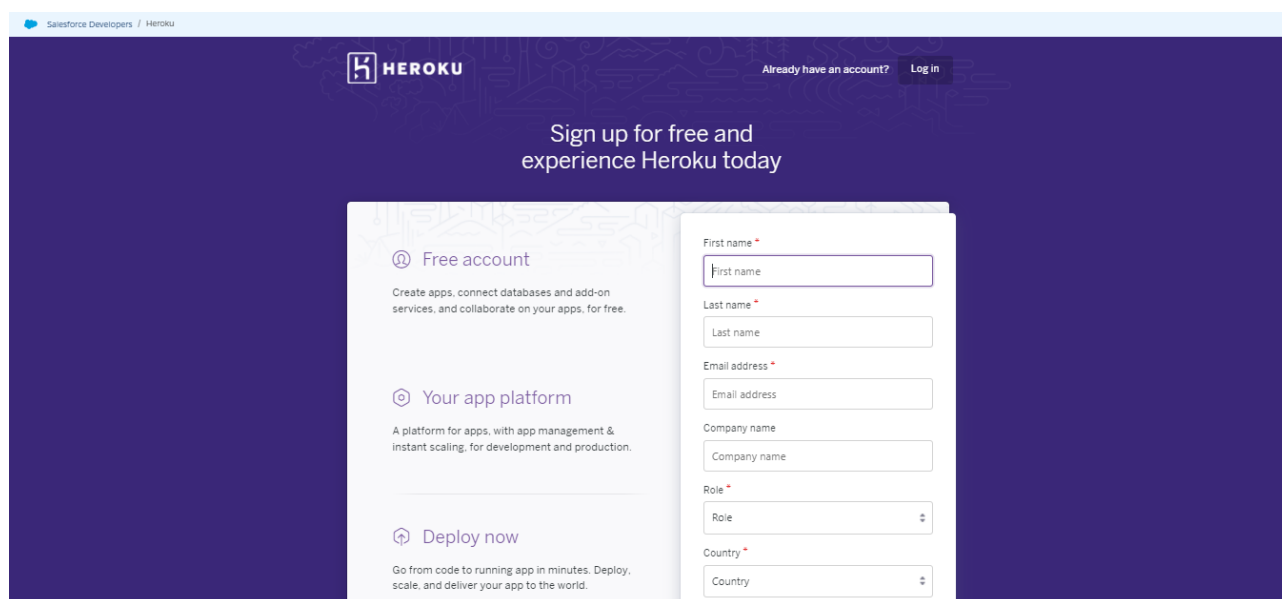


Figura 19: Registro *heroku*

Una vez registrado, accederemos a nuestro *dashboard* en el que veremos las aplicaciones que tenemos funcionando, en este caso ninguna. Aquí crearemos una nueva aplicación que será nuestro proyecto.

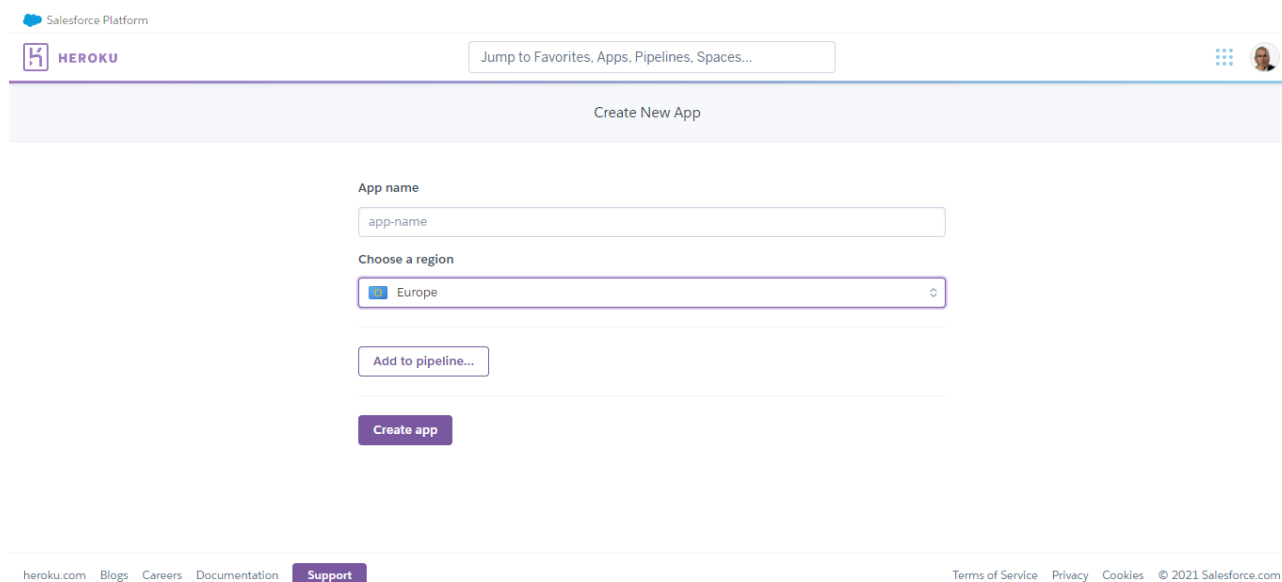


Figura 20: Nueva aplicación

Una vez creada, la aplicación aparecerá en el listado de nuestro *dashboard* pudiendo establecer algunas configuraciones que necesitemos o añadir *add-ons* necesarios, por ejemplo en nuestro caso, para añadir la base de datos con la que interactúa nuestra aplicación.

Para añadir un *add-on*, accederemos a la aplicación, en la pestaña de *resources*, y en la sección *Add-ons* buscamos el que necesitamos que se instale en nuestra aplicación.

En este caso solo necesitamos una base de datos *MySQL*. Para nuestra aplicación, se ha elegido el *add-on* *JawsDB MySQL*, que nos proporcionará una base de datos en la que volcar nuestro contenido.

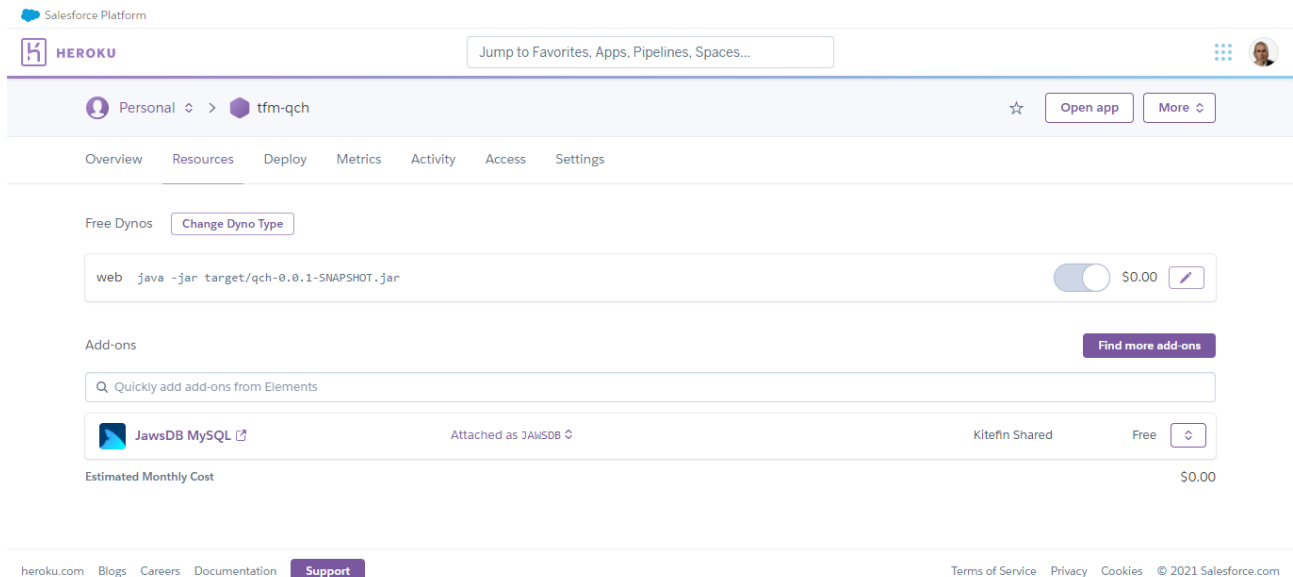


Figura 21: Add-on de base de datos

Hecho esto, ya tendríamos el entorno creado, ahora habría que subir el código fuente y migrar los datos de la base de datos que tenemos en local.

Para ello, *heroku* hace un despliegue automático del código que se encuentre en un repositorio que le indiquemos. El repositorio puede ser un repositorio de *GitHub* con el que estemos trabajando, o podemos subir el proyecto a un repositorio que nos indica *heroku* al crear la aplicación.

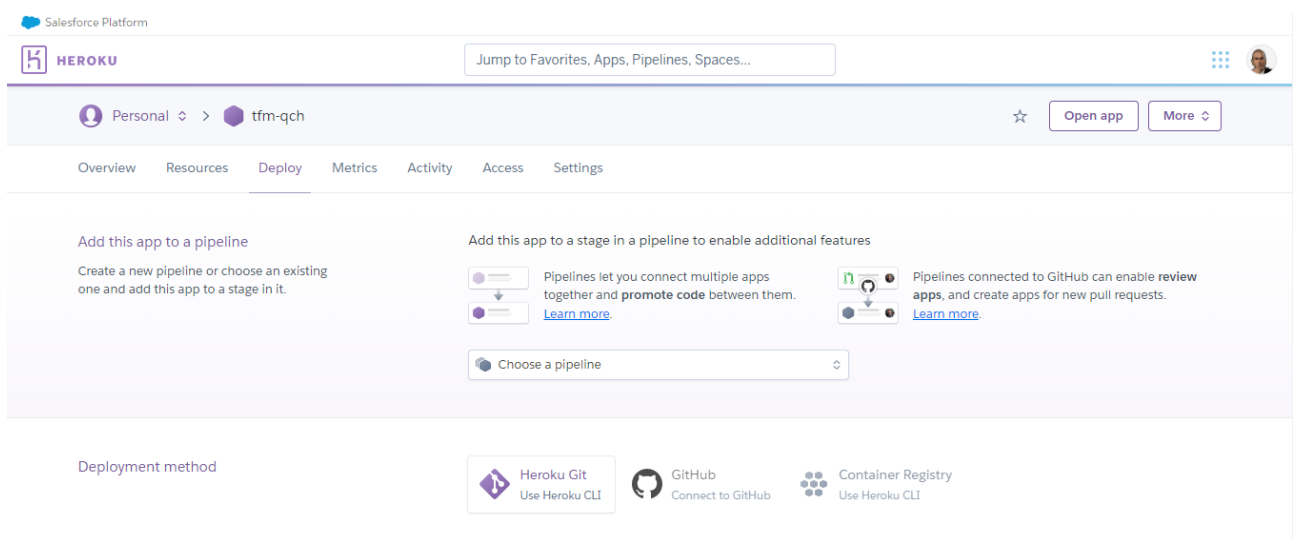


Figura 22: Elección de repositorio *heroku*

Una vez añadido el repositorio deseado, *heroku* se encargara de desplegar la aplicación cada vez que se produzca un *commit*.

Para realizar los *commit* y gestionar la maquina en la que se ejecuta la aplicación, es necesario descargarse el cliente de *heroku*, como viene indicado en la sección *deploy* del menú de la aplicación.

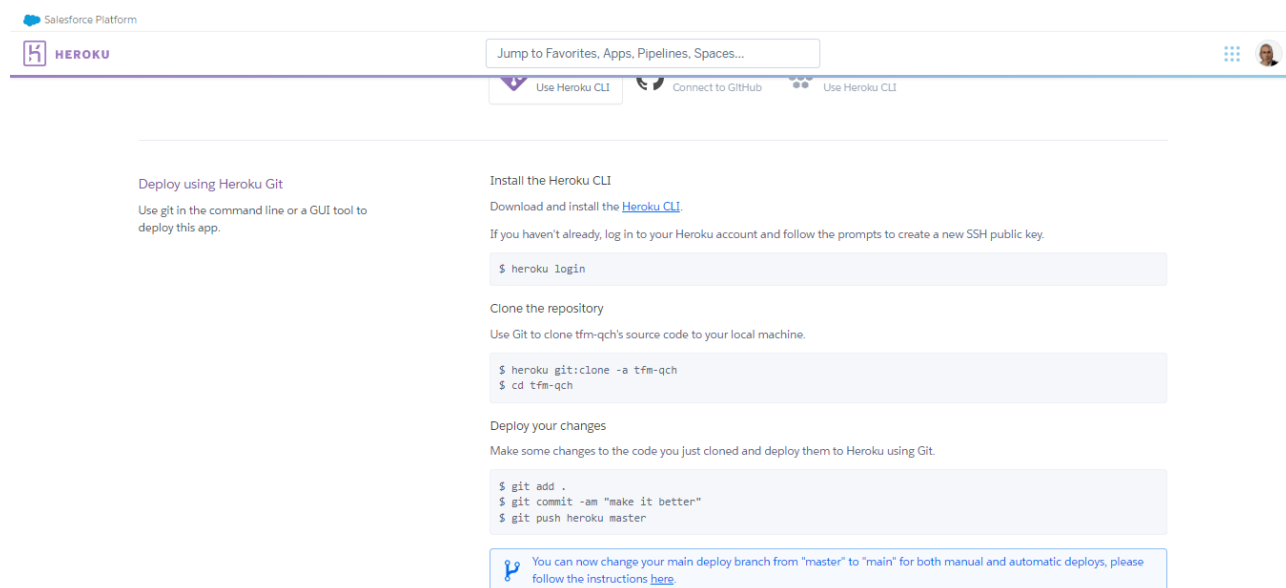


Figura 23: Cliente *heroku*

Hecho esto, solo quedaría crear las tablas de la base de datos que nos proporciona *heroku*, esta podemos verla en el apartado *settings* de la aplicación, iremos a la sección *Config Vars* y ahí se habrá creado una variable llamada *JAWSDB_URL*, creada por el *add-on* que hemos instalado previamente. *Heroku* se encarga de sustituir los datos de acceso de la base de datos en nuestro *application.properties*, por lo que no será necesario configurar nada más. En esta sección, también se pueden añadir variables que se sustituirán en nuestro *application.properties* si la indicamos con el formato *\${VARIABLE}*.

Finalmente, *heroku* nos proporcionará un dominio público con el que poder acceder a nuestra aplicación. En este caso: <https://tfm-qch.herokuapp.com>

12. Proyección a futuro

Antes de empezar con el desarrollo del proyecto, se estudian las funcionalidades que se desea tener en la aplicación resultante, pero su implementación, depende del tiempo disponible y las dificultades que vayan apareciendo al ir avanzando fases. Al final se ha conseguido desarrollar un producto totalmente funcional y con bastantes funcionalidades, pero además, han ido surgiendo bastantes ideas que pueden ser interesantes de implementar en un futuro para terminar de hacer un proyecto más competente y completo.

A continuación enumero algunas de estas funcionalidades que se han quedado en el tintero.

12.1. Registro e inicio de sesión con redes sociales

Una de las funciones que más cuesta a los usuarios es el tema del registro y el *login*, ya que tienen que introducir una serie de datos que ya han introducido en otras aplicaciones y puede decantar la balanza acerca de si se empieza a usar la aplicación o se da de lado por “pereza” a rellenar esos datos de registro.

Para solucionar eso, se puede implantar un sistema de registro/*login* que se establezca mediante la obtención de la sesión usando redes sociales como las cuentas de *Google* o *Facebook*. Esto simplificaría el alta de los datos del usuario y el inicio de sesión en la aplicación.

12.2. Notificaciones *push*

Al ser la aplicación una *PWA*, esta está preparada para que sea usada en todos los dispositivos posibles, incluidos los dispositivos móviles. Esto da la posibilidad de implementar un sistema de suscripciones que permita enviar notificaciones *push* a los usuarios, y no tener que usar, por ejemplo, un correo electrónico para ello.

Esto se podría usar para notificar si un usuario concreto ha creado una receta nueva, o si alguien ha reaccionado a una receta que haya creado ese usuario. También se podría advertir si con la consumición de una receta determinada, ha llegado al tope de veces que puede consumir un ingrediente concreto en un periodo de tiempo, o incluso hacer recomendaciones de recetas personalizadas.

12.3. Añadir videos o galería de imágenes a las recetas

En las recetas, actualmente se incluye una imagen del resultado final que se obtiene y una lista con los pasos a seguir para prepararla, pero sería una mejora, poder añadir un video de esos pasos que hay que

seguir o incluso una serie de imágenes que represente cada paso y muestre más gráficamente como se puede completar la receta correctamente.

12.4. Exportación de recetas a pdf

Otra funcionalidad interesante, puede ser el permitir que se pueda exportar una receta a formato PDF, para que un usuario pueda imprimirlo y tenerlo a mano en su cocina sin tener que consultar el dispositivo en un entorno que no es muy propicio para ello como suele serlo una cocina.

12.5. Sistema más complejo de valoración de recetas

Podría ser interesante que cada receta tuviera un sistema de valoración en el que cada usuario pudiera dejar una valoración en forma de nota y además añadir un comentario sobre que le ha parecido el proceso de elaboración de la receta y el resultado final obtenido. Así, los demás usuarios tendrían una referencia de alguien que ha puesto en marcha esa receta concreta antes.

12.6. Estudio en profundidad de la alimentación

Actualmente se han estudiado los tipos de ingredientes y las veces que están recomendados consumirlos a la semana. Sería interesante afinar mucho más estos datos con una persona especializada en el campo de la nutrición y que pueda aportar más datos y más exactos acerca de las recomendaciones de recetas y de ingredientes para poder informar más detalladamente al usuario de en qué estado está la nutrición que está llevando a cabo.

12.7. Compra de ingredientes

Por último, una mejor muy ambiciosa, sería el poder elegir una receta y marcar los ingredientes que tiene disponibles el usuario para poder prepararla. Esto serviría para que los ingredientes de los que no se dispone, se pudiera añadir a una lista de la compra y hacer un pedido directamente en un supermercado que colabore con la aplicación. Esto simplificaría el proceso de pensar en que hacer de comer, puesto que se elegiría la receta y se haría la compra en un simple proceso.

13. Conclusiones

El objetivo principal de realizar este proyecto ha sido el desarrollar una aplicación completamente funcional usando tecnologías que no conocía antes de cursar este master.

Se ha implementado una aplicación para gestionar recetas culinarias, analizando los ingredientes que la componen, el tipo de ingrediente y las veces que este ingrediente está recomendado consumir a la semana. Esto viene motivado a la dificultad o falta de imaginación que habitualmente se tiene al plantear un menú que cocinar diariamente.

Realizando el proyecto se ha aprendido además de conceptos informáticos, de conceptos básicos de nutrición, ilustrándome un poco acerca de que tipos de alimentos son más sanos y otros hay que consumirlos contadas veces a la semana.

En cuanto a las tecnologías usadas, nunca había realizado un proyecto que usara **Angular** para implementar el *frontend* de una aplicación web. Con lo aprendido en las asignaturas cursadas, se ha podido investigar acerca de más funcionalidades y componentes externos nuevos, tampoco usados en estas asignaturas y que se han incluido en este proyecto con un resultado bastante satisfactorio.

Tampoco había implementado nunca, aparte de en las prácticas de una de las asignaturas cursadas, una *PWA*, que también me llamó bastante la atención por su versatilidad, cosa que he confirmado durante el desarrollo de la aplicación.

Además, ha sido un proceso bastante costoso el subir la aplicación a un servidor para obtener un dominio público, ya que ha habido problemas de permisos de archivos, de versión de base de datos o de características que tienen un comportamiento correcto en el entorno local, pero sin embargo en el entorno de producción no funcionaban bien, por lo que se ha tenido que ir corrigiendo sobre la marcha y se ha conseguido solucionar todo. Se han obtenido bastantes conocimientos realizando este proceso.

Como conclusión, este proyecto me ha ayudado a conocer un poco más **Angular** y terminar de convencerme de que es una tecnología muy atractiva y que podría usar más veces en proyectos reales futuros, me ha ayudado a ampliar conocimientos acerca del uso de *bootstrap*, he realizado un proceso de subida a un servidor con un dominio público, con todos los problemas que ello conlleva si es de las primera veces que se hace y sobre todo a gestionar mejor el tiempo y análisis de cada tarea que se ha implementado.

Anexo 1. Entregables del proyecto

A lo largo del desarrollo se han ido realizando varios entregables tanto de código fuente como de la memoria generada, además de los recursos incluidos en ellos como imágenes, diagramas y esquemas.

Durante el todo el proceso, también se ha usado un repositorio *Github* para ir guardando todas las versiones del código fuente que se han ido desarrollado con el tiempo, siempre teniendo en cuenta que en el repositorio debe haber una versión totalmente funcional en cada subida.

Se puede acceder al repositorio con todo el código subido desde el principio al fin del desarrollo, pulsando en el siguiente enlace: <https://github.com/ajrodasr/TFM-Que-como-hoy>

Anexo 2. Librerías externas

A continuación se detallan algunas librerías y componentes externos que se han usado en la aplicación.

Bootstrap

Bootstrap es un componente que proporciona una serie de estilos definidos, lo que hace que el maquetado se haga bastante más sencillo si se conoce un poco las distintas clases que se pueden usar para resolver una situación determinada.

Además, incluye una serie de componentes como barras de navegación o acordeones (usados en el proyecto) que incluyen *JavaScript* y que dota a las vistas de un diseño fluido y atractivo.



Figura 24: *Bootstrap*

Animate css

Es un repositorio externo que también proporciona diferentes hojas de estilo que permiten añadir animaciones a los distintos elementos que conforman la vista.



Figura 25: *Animate css*

@kolkov/angular-editor

Se trata de un componente externo, que se puede instalar en *Angular* y que permite insertar un editor de texto enriquecido. Este componente se ha usado para la inserción o edición de las instrucciones de elaboración de una receta, permitiendo formatear el texto resaltando palabras importantes o haciendo una lista enumerada de los pasos que hay que seguir para que la receta salga a la perfección.

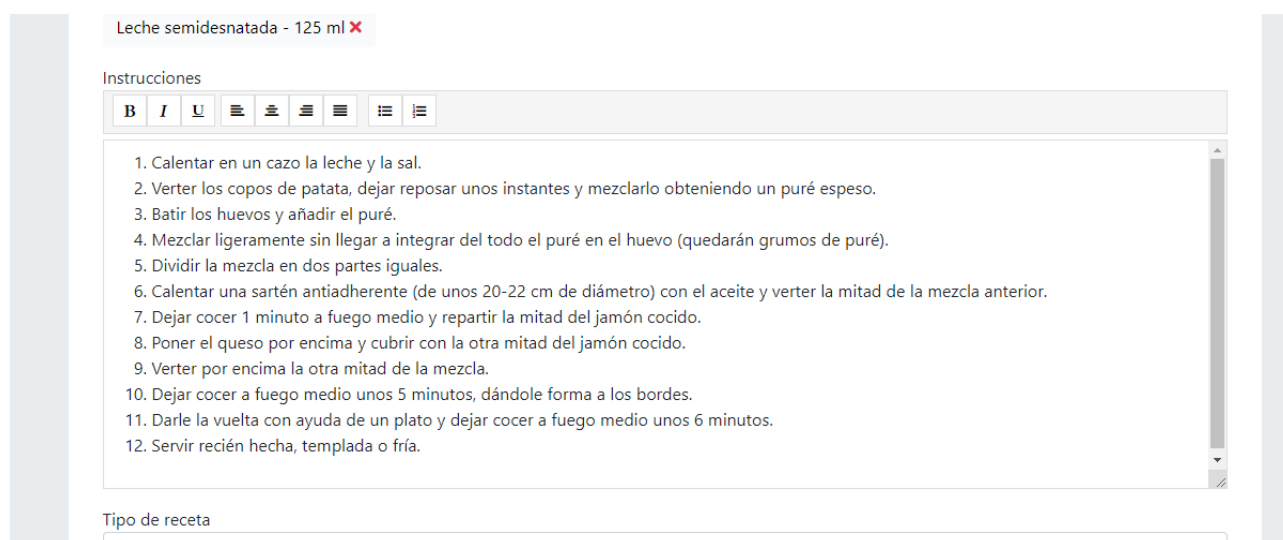


Figura 26: Editor texto enriquecido

ng-circle-progress

Es un componente externo que permite pintar en la pantalla una barra de progreso, pero al estilo circular. Este componente se ha usado para mostrar el número de veces que se ha consumido un tipo de ingrediente en un tiempo determinado, respecto al número de veces que está recomendado, indicando con colores y con ese progreso, si está recomendado consumir más veces ese alimento o es mejor optar por otra receta que no lo contenga.



Figura 27: ng-circle-progress

Anexo 3. Guía de uso

La aplicación consta de dos *roles* diferenciados que son *USER* (usuarios) y *ADMIN* (administradores). Estos roles se diferencian que el *ADMIN* puede hacer todo lo que hace el *USER* pero además tiene ciertas funcionalidades añadidas que permiten de alguna forma el moderar la aplicación para que no se haga un uso indebido de ella.

A continuación se explica las funciones que puede realizar cada tipo de usuario.

Guía de usuarios

Para acceder a la aplicación, es necesario estar registrado, por lo que la primera pantalla que veremos será la pantalla de **Inicio de sesión**.

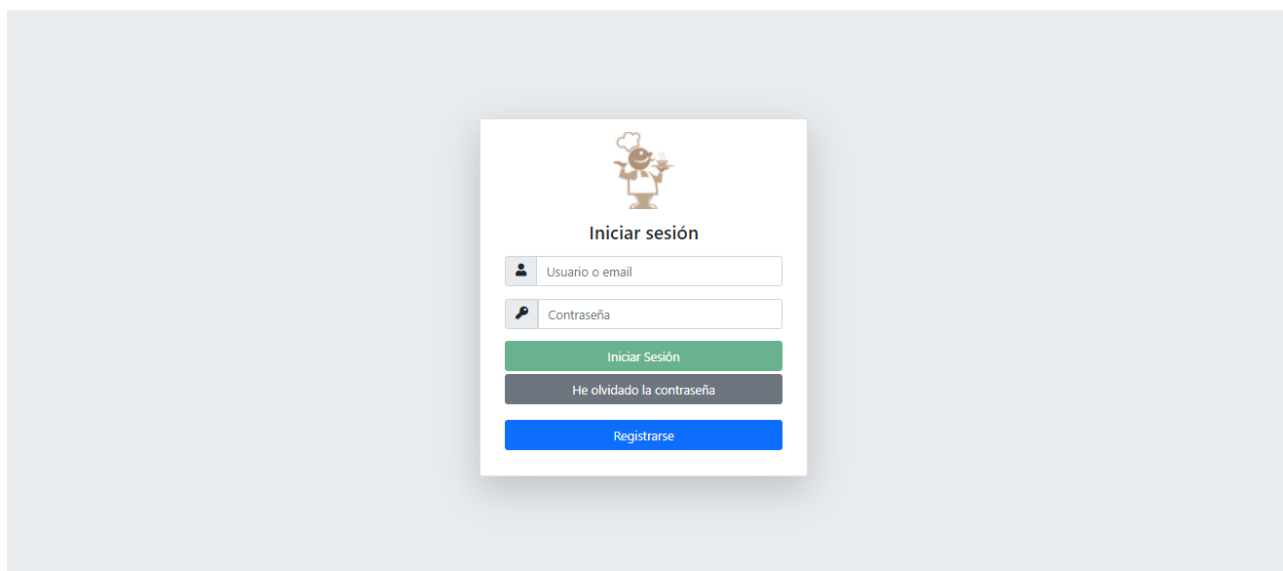


Figura 28: Inicio de sesión

Se podrá iniciar sesión introduciendo bien el nombre de usuario que hayamos elegido o el correo electrónico que rellenamos al registrarnos en el sistema. Si no tuviéramos los datos de acceso, entonces lo primero que habría que hacer sería registrarse.

Pulsando en **Registrarse** accederemos a la pantalla de registro.

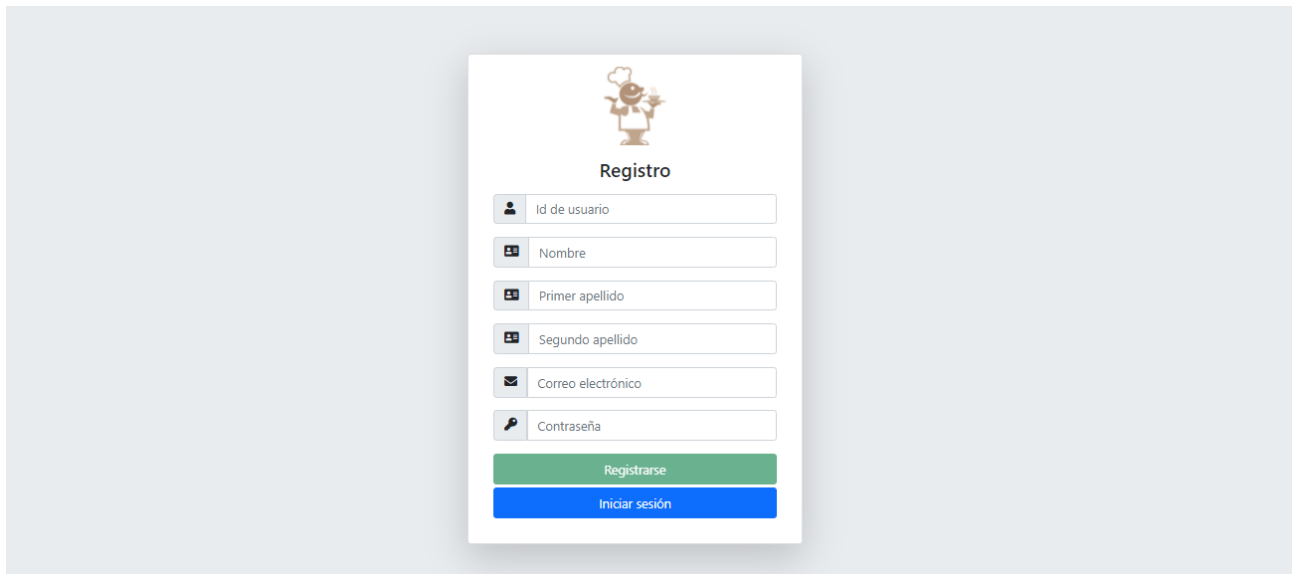
La imagen muestra la pantalla de registro de una aplicación web. En el centro hay un formulario con el título "Registro" y un icono de un chef. El formulario contiene los siguientes campos de entrada: "Id de usuario", "Nombre", "Primer apellido", "Segundo apellido", "Correo electrónico" y "Contraseña". Debajo de los campos hay dos botones: "Registrarse" (verde) y "Iniciar sesión" (azul).

Figura 29: Pantalla de registro

En esta pantalla, introduciremos los datos correspondientes y pulsaremos en **Registrarse**. Hecho esto se enviara un correo de bienvenida a la dirección que se haya introducido en el formulario y ya dispondremos de usuario y contraseña para poder acceder a la aplicación.

Cuando accedemos a la aplicación, veremos el menú principal, y en la esquina superior derecha, nuestro nombre de usuario, en el que si pinchamos, se despliega una serie de opciones para gestionar los datos de nuestro usuario entre otras cosas que veremos más adelante.

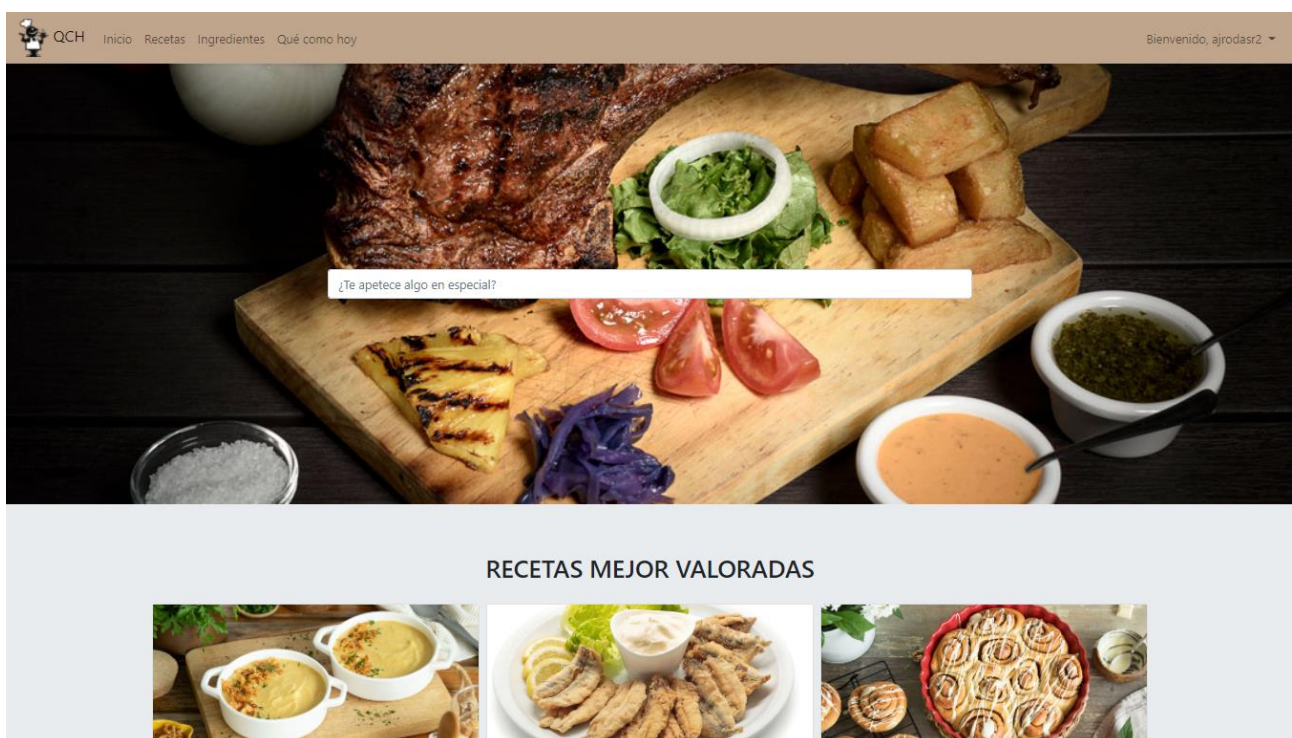


Figura 30: Pantalla de inicio

En esta pantalla, lo primero que vemos será un buscador, en el que podemos escribir cualquier palabra y se mostrará una lista de recetas que coinciden con esa palabra, en la cual podremos pulsar para acceder directamente a esa receta.

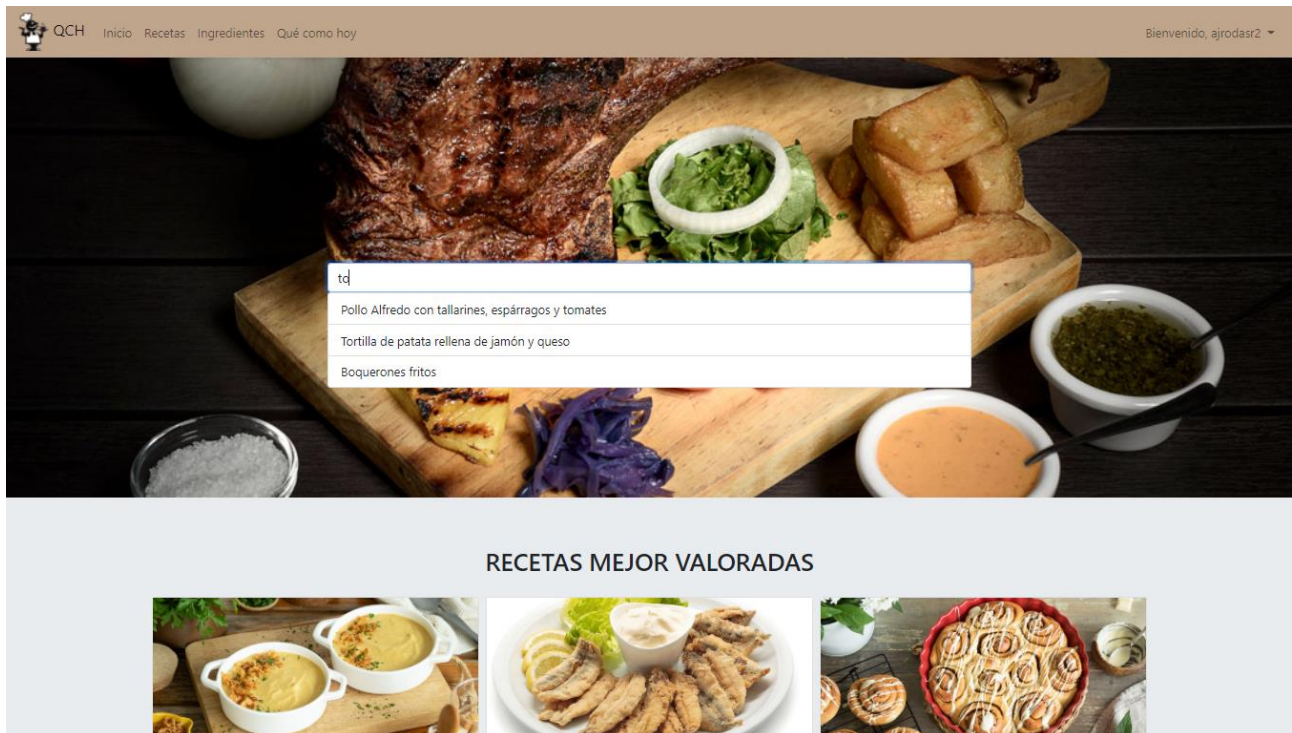


Figura 31: Buscador inicio

Más abajo veremos una sección con las recetas que están mejor valoradas con mayor cantidad de *likes* y otra sección con las ultimas recetas que se han añadido a la plataforma.

Mirando al menú de navegación, vemos la primera opción **Inicio**, que nos lleva a esta misma página, la página de inicio.

En la segunda opción de menú, tenemos **Recetas**, que nos llevara a una página con todas las recetas **publicadas** que se encuentran en la plataforma. Además, esta página contiene un formulario que permite filtrar las recetas mostradas según los parámetros introducido. Las recetas se muestran de forma paginada en un máximo de 9 recetas por página.

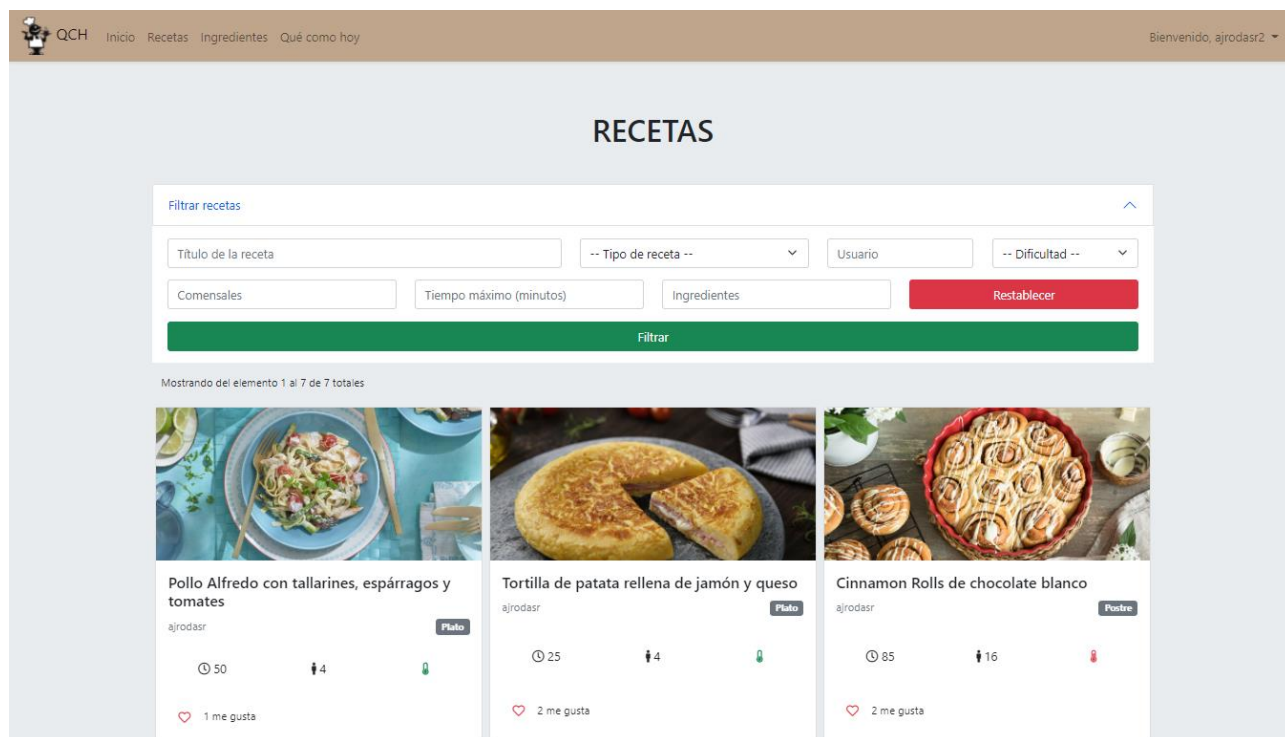


Figura 32: Página de recetas

Aquí tenemos la posibilidad, además de filtrar, de valorar cada receta dándole *like* y de picar en cada receta para ver su detalle, los ingredientes que contiene, y como son las instrucciones para su elaboración.

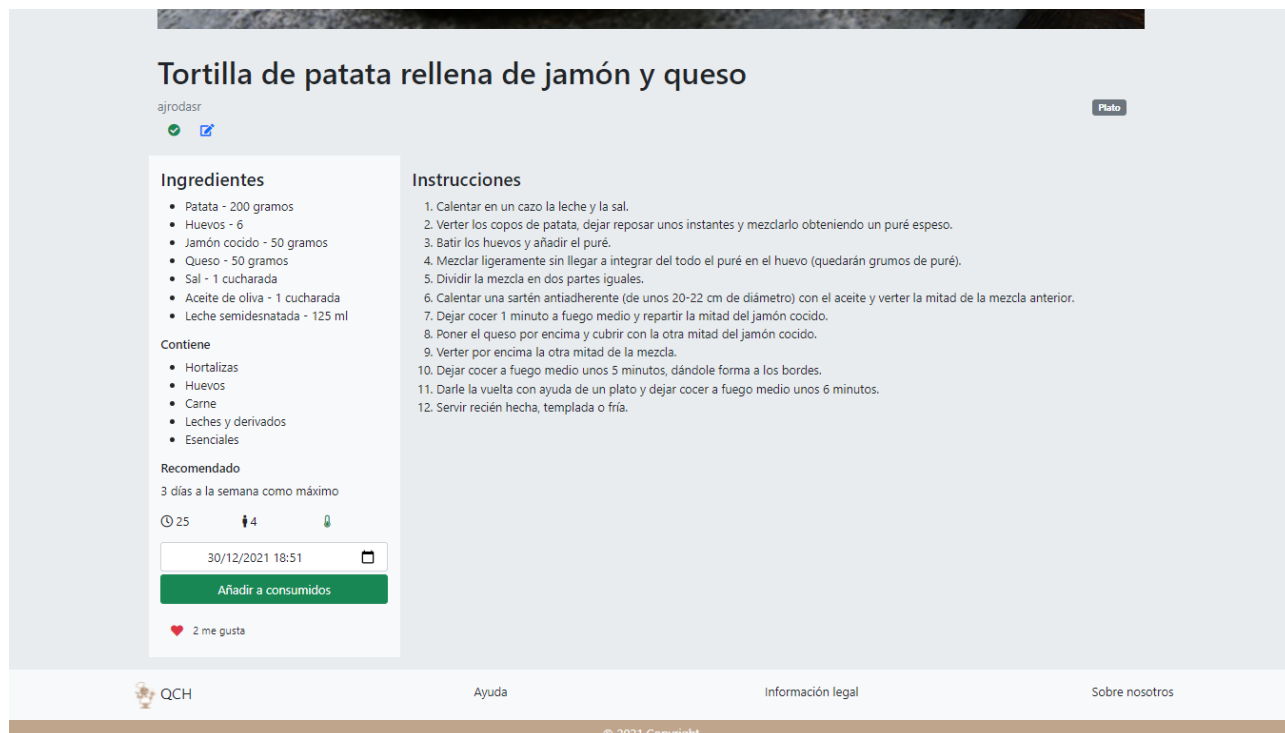


Figura 33: Página de detalle de receta

En la página de detalle, podemos ver la imagen a mayor tamaño de la receta, las instrucciones de elaboración, el tipo de receta del que se trata, para cuantos comensales está pensada, el tiempo de elaboración, y la dificultad que conlleva el prepararla.

Además, se puede ver los ingredientes que la componen y los grupos a los que pertenece cada ingrediente, lo cual es importante para calcular la información que se puede ver justo debajo de los ingredientes, que sería cuantas veces por semana está recomendado, como máximo, consumir esa receta en una semana.

Otra función que contiene esta página es que puedes guardar la fecha y hora en la que has consumido esta receta. Este dato se usa en una pantalla que veremos más adelante en la que se muestra un análisis de los alimentos consumidos.

Por último, podemos observar dos botones justo debajo del nombre del usuario que ha creado la receta. Estos dos botones solo aparecerán, si el usuario que ha creado la receta es el que está *logado*. Son dos acciones que, la primera, permite publicar o despublicar la receta para que pueda o no, ser encontrada por los demás usuarios; y la segunda, permite acceder al formulario de edición de esa misma receta, para que se pueda modificar alguno de sus datos.

Si avanzamos en el menú de navegación, tenemos la sección **Ingredientes**. Aquí podremos ver una lista paginada de todos los ingredientes que se han dado de alta en la aplicación. También dispondremos de un formulario que permitirá filtrar los ingredientes por nombre y por grupo al que pertenecen.

INGREDIENTES

Nombre del ingrediente: -- Grupo de ingrediente --

Mostrando del elemento 1 al 16 de 61 totales

Aceite de girasol Esenciales (7 raciones/semana)	Aceite de oliva Esenciales (7 raciones/semana)	Agua Esenciales (7 raciones/semana)	Atún Pescado (4 raciones/semana)
Azúcar Esenciales (7 raciones/semana)	Azúcar moreno Esenciales (7 raciones/semana)	Bacon Carne grasa (2 raciones/semana)	Berenjenas Verduras (7 raciones/semana)
Boquerón Pescado (4 raciones/semana)	Cacahuete Frutos secos (7 raciones/semana)	Café de sal Esenciales (7 raciones/semana)	Calabaza Verduras (7 raciones/semana)
Canela Especias (7 raciones/semana)	Canela en polvo Especias (7 raciones/semana)	Canela en rama Especias (7 raciones/semana)	Carrillada Carne (3 raciones/semana)

<< < 1 2 3 4 > >>

QCH Ayuda Información legal Sobre nosotros

Figura 34: Página de ingredientes

Además, si se pulsa sobre el botón de **Nuevo ingrediente**, se abrirá un formulario en el que podremos crear un nuevo ingrediente que se añadirá a la lista.

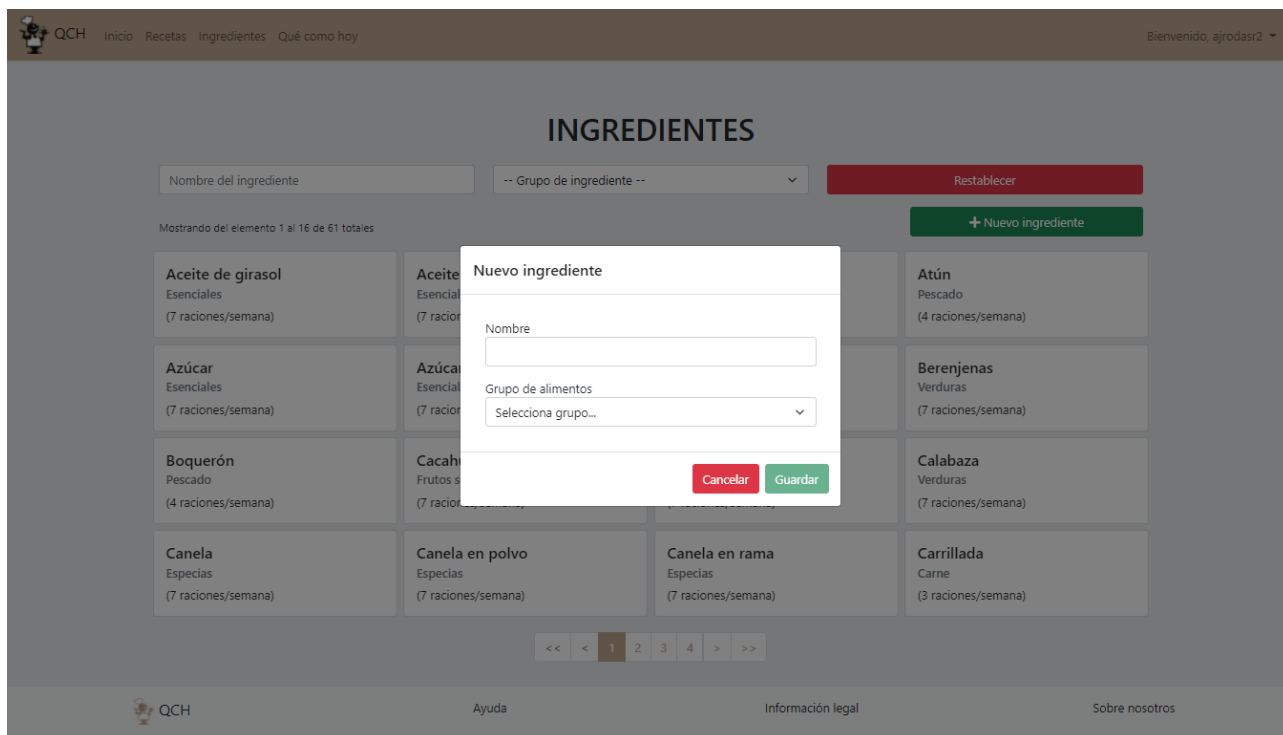


Figura 35: Crear nuevo ingrediente

En la siguiente sección, **Que comemos hoy**, se obtiene un listado de recetas como en la sección de recetas, con la diferencia de que aquí solo saldrán las recetas que se recomiendan según lo que hayamos marcado como consumido durante los últimos 7 días. Así, si por ejemplo, se ha consumido una receta que contiene dulce, y el dulce está recomendado, como máximo, una vez por semana, en ese listado no saldría ninguna receta que contenga un ingrediente que este en el grupo de dulces.

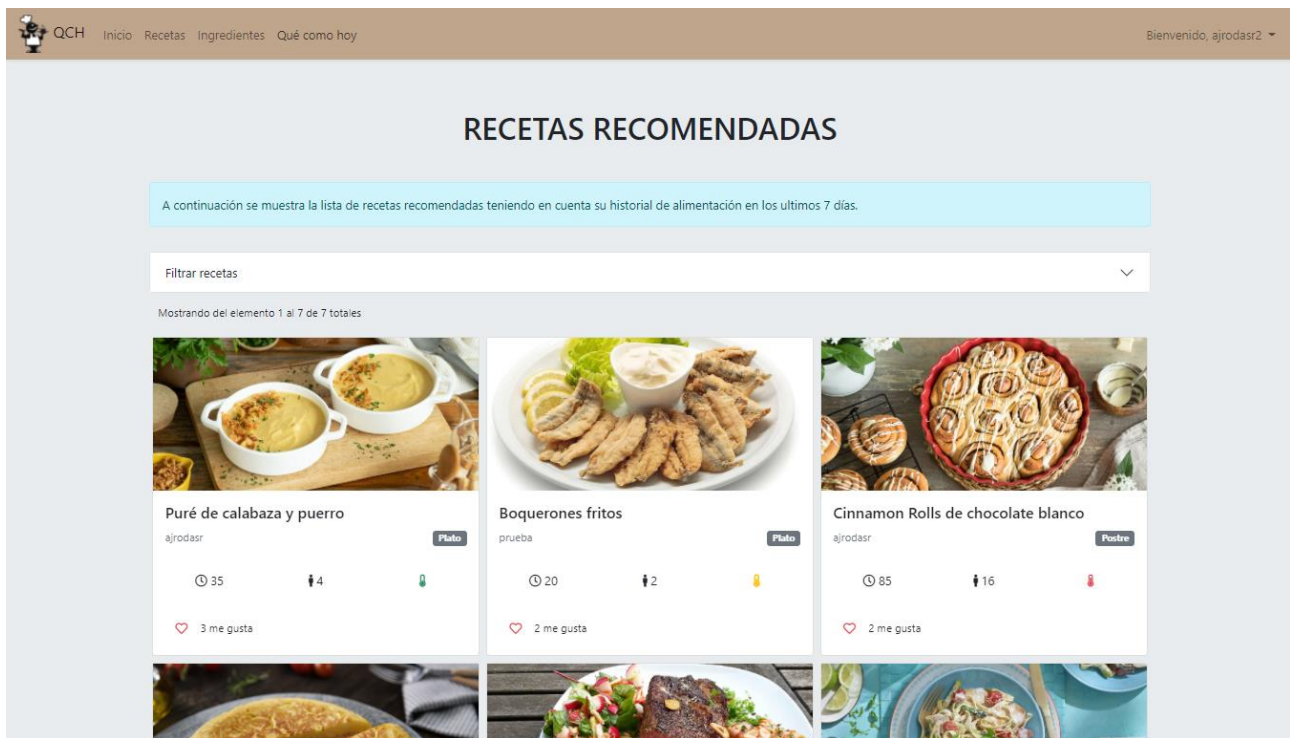


Figura 36: Página de recetas recomendadas

Si continuamos, y pulsamos sobre nuestro nombre de usuario, obtenemos 3 opciones, **Mi perfil**, **Mis recetas** y **Mi alimentación**.

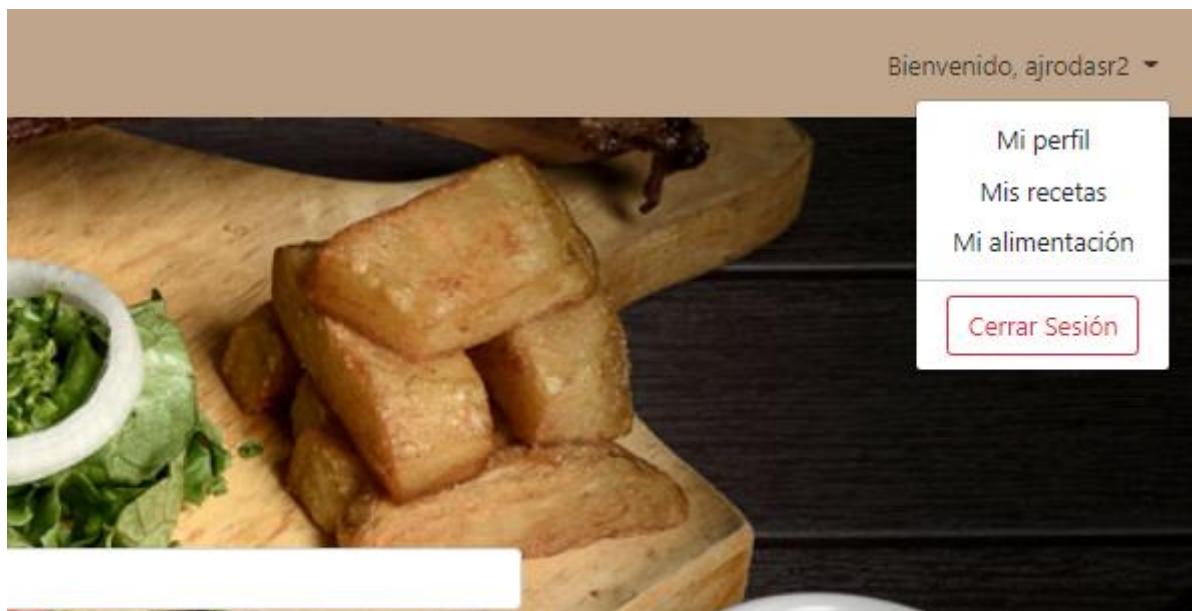


Figura 37: Desplegable usuario

Si accedemos a **Mi perfil**, podremos cambiar los datos de nuestro perfil o modificar la contraseña.

Si pinchamos en **Mis recetas**, veremos un listado de recetas igual que el de la sección **Recetas**, pero en el que solo aparecen las recetas que hayamos creado nosotros, tanto si están publicadas como si no, en el cual podremos igualmente filtrar, publicar, despublicar y pinchar para editar la receta deseada.

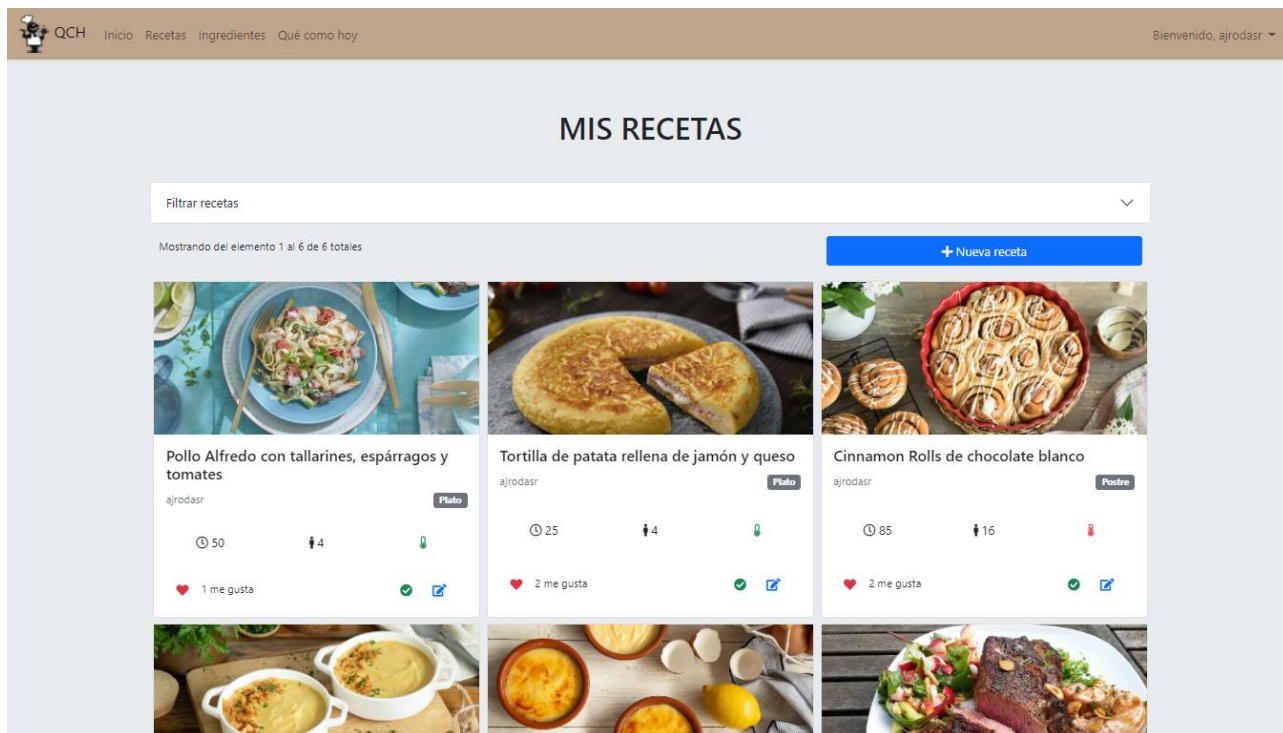


Figura 38: Página mis recetas

Además, tenemos un botón de **Nueva receta**, que no llevara a un formulario en el que crear una nueva receta que añadir a la plataforma.

The screenshot shows the 'Nueva receta' form. At the top, there's a header with 'QCH' and navigation links: 'Inicio', 'Recetas', 'Ingredientes', and 'Qué como hoy'. A user greeting 'Bienvenido, ajrodras' is on the right. The form itself has a title 'Nueva receta' and several sections:

- Titulo:** A single-line text input field.
- Ingredientes:** A section with a 'Nombre' input field, a 'Cantidad' input field, an 'Añadir' button, and a 'Crear ingrediente' button.
- Instrucciones:** A section with a rich text editor toolbar (bold, italic, underline, link, unlink, list, ul) and a large text area with the placeholder 'Escriba aquí las instrucciones...'.
- Tipo de receta:** A dropdown menu with the placeholder 'Selecciona tipo de receta...'.
- Imagen:** A section with a 'Seleccionar archivo' button and the text 'No se eligió archivo'.
- Tiempo de preparación:** A single-line text input field.
- Comensales:** A single-line text input field.

Figura 39: Formulario de receta

En este formulario, se introducen los datos de la receta. Como particularidad, para introducir un ingrediente en la receta, se debe seleccionar de una lista que aparece al buscar en el campo nombre. Si este no existiese, tiene la posibilidad de crear un ingrediente nuevo sin salir de esta misma pantalla pulsando en **Crear ingrediente**.

This screenshot shows the same 'Nueva receta' form as Figure 39, but with a modal dialog box open in the center. The modal is titled 'Nuevo ingrediente' and contains:

- A 'Nombre' input field.
- A 'Grupo de alimentos' dropdown menu with the placeholder 'Selecciona grupo...'.
- Two buttons at the bottom: 'Cancelar' (red) and 'Guardar' (green).

 The background form is dimmed, showing that the 'Ingredientes' section now contains two items: 'Sardina - Medio kilo' and 'Puerro - 250 gramos', each with a close button (X).

Figura 40: Crear ingrediente desde receta

Una vez introducidos los datos, pincharíamos en **Crear receta** y esta se guardaría como **No publicada**. Para que aparezca en el listado de recetas, habría que darle a **Publicar**.

Por últimos, la última de las secciones sería la sección de **Mi alimentación**, dentro del desplegable que aparece al pulsar en nuestro nombre de usuario.

En esta pantalla, de muestra un análisis de nuestra alimentación en función de las recetas que hemos marcado como consumidas en una fecha en concreto.

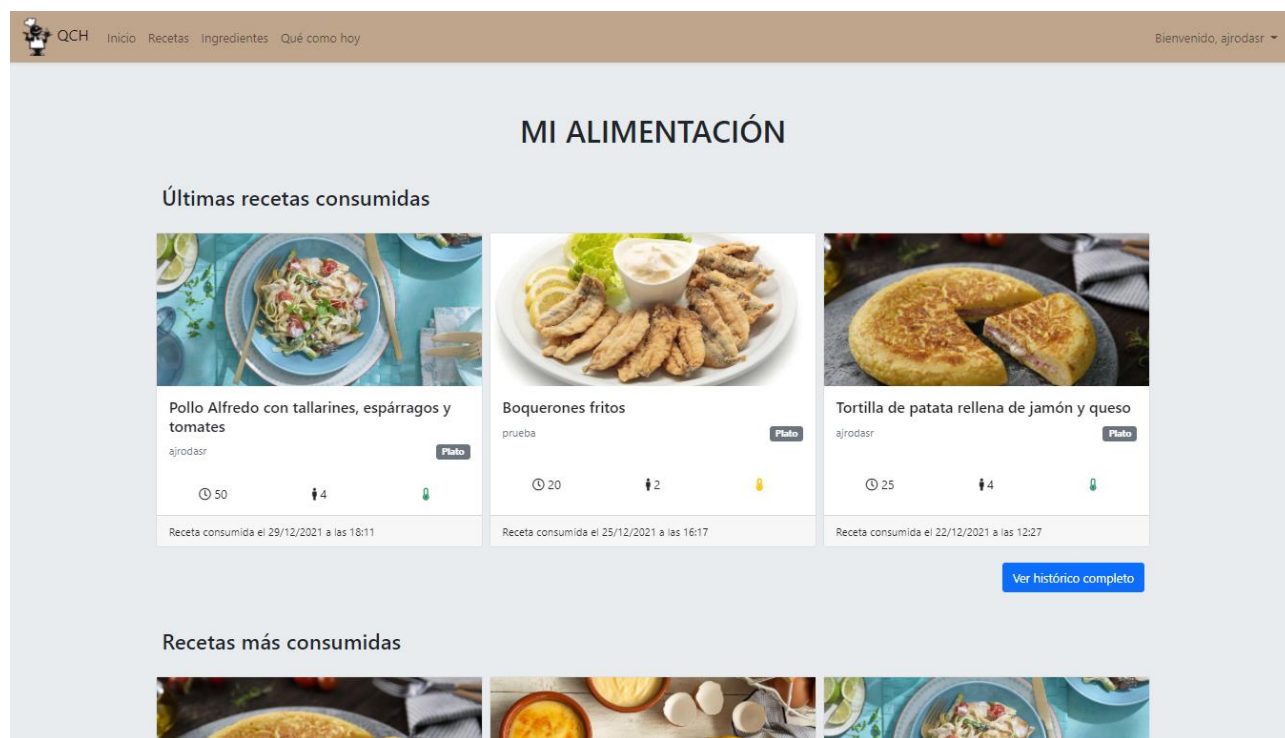


Figura 41: Página de mi alimentación

Aquí podemos ver un resumen de las últimas recetas que hemos consumido, las recetas que hemos consumido, los ingredientes que más hemos usado, y por ultimo un gráfico del estado en que se encuentra nuestra alimentación en los últimos siete días.



Figura 42: Ingredientes últimos siete días

Aquí podremos cuantas veces hemos consumido un tipo de ingrediente en la última semana y cuanto nos queda para llegar a la cantidad que está recomendada. Se marcarán con la barra de progreso y con distintos colores según nos vayamos aproximando al tope de consumiciones semanales. Hay ingredientes que se pueden consumir infinitas veces a la semana, por lo que su progreso siempre estará completo y en color verde.

En esta sección, también tendremos un botón denominado **Ver histórico completo**. Si pinchamos en él, se nos dirigirá a una página que muestra todas las recetas que hemos marcado como consumidas y la fecha en las que las hemos marcado.

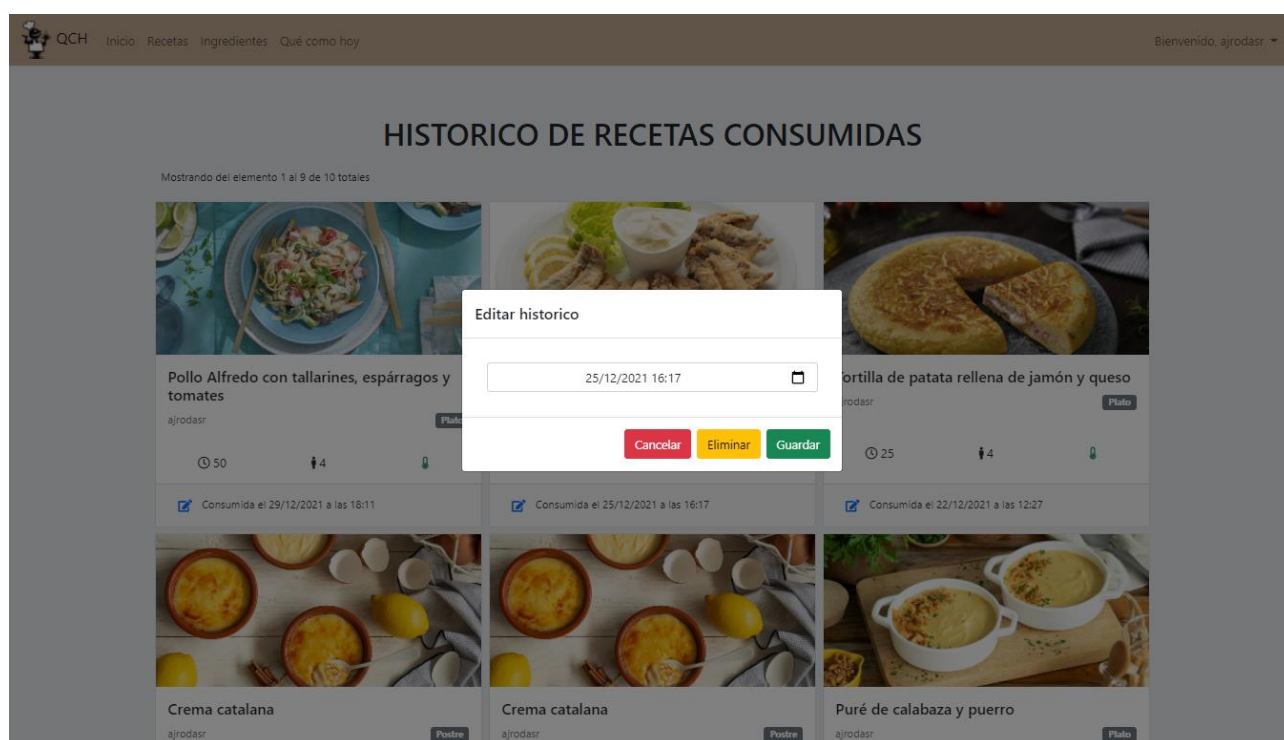


Figura 43: Histórico de consumiciones

En esta página, podremos modificar la fecha en la que se consumió una receta o incluso eliminar esa receta del historial, lo que afectaría y modificaría los datos de las pantallas de **Mi alimentación** y de **Qué como hoy**.

Guía de administradores

Un administrador de la aplicación, puede hacer las mismas cosas que un usuario, pero además, tiene ciertas opciones añadidas que permiten moderar el estado de la aplicación.

En primer lugar, vera modificado el menú principal con una opción más, llamada **Administrador**. Si pincha en esa opción, vera desplegada una sección llamada **Usuarios**.

QCH Inicio Recetas Ingredientes Qué como hoy Administrador Bienvenido, ajrodasr

LISTA DE USUARIOS

Buscar

ID	Nombre	Email	Administrador
ajrodasr	Antonio Jesús Rodas Rueda	prueba@gmail.com	✓
ajrodasr2	Antonio Rodas Rueda	prueba2@gmail.com	✗
alesapae	Alejandro Gómez	prueba3@gmail.com	✗
prueba	Prueba Prueba Prueba	prueba4@gmail.com	✗

Mostrando del elemento 1 al 4 de 4 totales

<< < 1 > >>

QCH Ayuda Información legal Sobre nosotros

© 2021 Copyright

Figura 44: Gestión de usuarios

Pinchando en esa opción, aparece una pantalla que muestra un listado de usuarios en la cual se pueden filtrar por **ID**, **Nombre** o **Email**. Aquí el administrador puede establecer o quitar a un usuario el rol de administrador.

Y por último, el administrador también tendrá la opción de editar los ingredientes que se hayan creado en la aplicación.

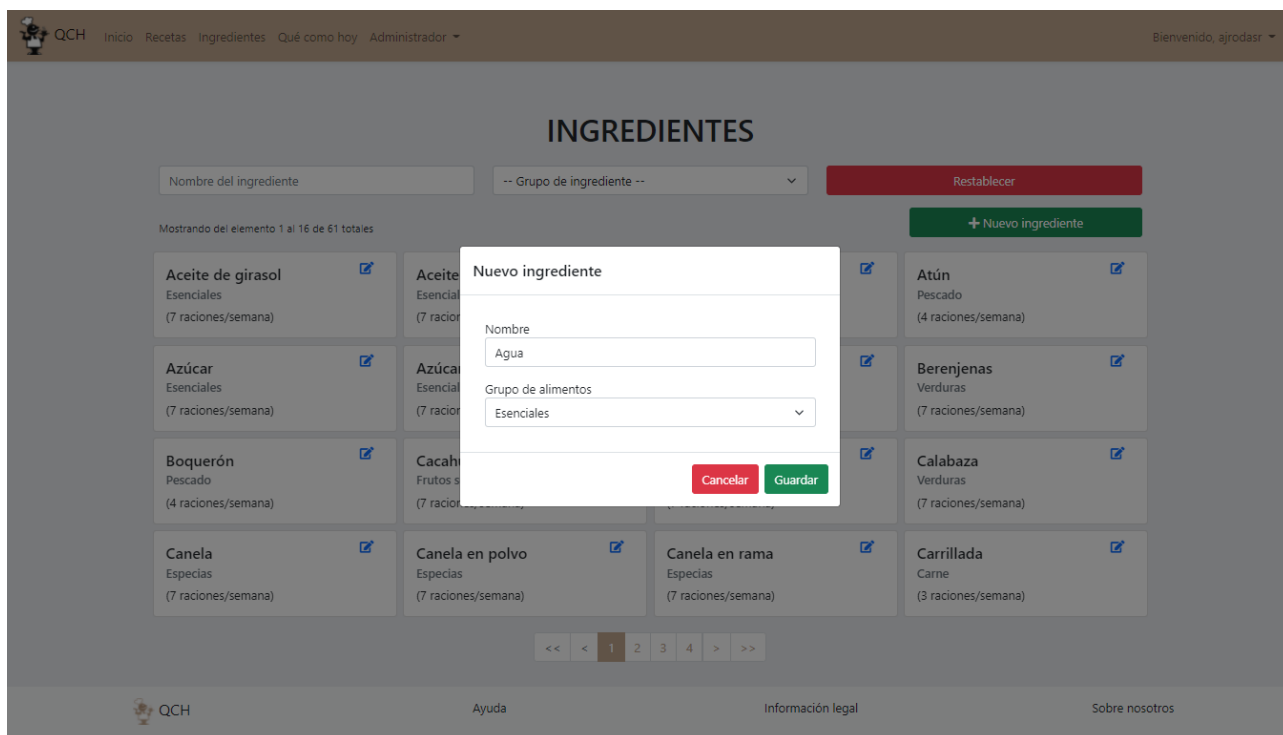


Figura 45: Editar ingredientes

Accediendo a la pantalla **Ingredientes**, si el usuario es **Administrador**, le aparecerá un nuevo icono en cada ingrediente, lo que le da la opción de editarlo por si ha habido algún error ortográfico, o se ha incluido un ingrediente en un grupo al que no pertenece realmente.

Anexo 4. Bibliografía

A continuación se muestran los principales recursos bibliográficos que se han consultado durante el desarrollo de este proyecto:

1. Realización del diagrama de *Gantt*: *GanttProject* - <http://www.ganttproject.biz>
2. Realización de diagramas de casos de uso y arquitectura: *DIA* - <http://dia-installer.de>
3. Realización de prototipos de la interfaz: *Justinmind* - <https://www.justinmind.com>
4. Consulta de conceptos *Spring*: *Baeldung* - <https://www.baeldung.com>
5. Consulta de conceptos *Angular*: *Angular doc* - <https://angular.io>
6. Consulta de conceptos *MySQL*: *MySQL doc* - <https://dev.mysql.com/doc>
7. Consulta de conceptos *HTML* y *CSS* - <https://www.w3schools.com>
8. Maquetación de vistas: *Bootstrap* - <https://getbootstrap.com>
9. Iconos : *Fontawesome* - <https://fontawesome.com>
10. Graficas: *ng-circle-progress* - <https://www.npmjs.com/package/ng-circle-progress>
11. Seguridad: *Json Web Token* - <https://jwt.io>
12. Recetas de ejemplo para la aplicación: *Nestle cocina* - <https://www.nestlecocina.es>