

Pirquet - Allergic Detected

Cristian Correa Román

Resumen — Este documento explica el proyecto Pirquet, que nace con la idea de ayudar a personas con alergias alimentarias a identificar qué alimentos pueden consumir, de una manera fácil e intuitiva. Pirquet se basa en 3 grandes apartados, dos de ellos enfocados al usuario final y otro destinado a posibles desarrolladores externos.

Los apartados destinados al usuario, son, una web principal desde la cual los usuarios, pueden gestionar sus alergias, ver sus búsquedas de productos y comprobar las alergias relacionadas con un producto en concreto. Los usuarios también tienen una aplicación Android destinada a la búsqueda de productos y sus alergias de una manera rápida y fácil a través del escaneo del código de barras de dicho producto con la cámara del dispositivo.

Para los desarrolladores, se ha creado una API con su respectiva documentación, destinada al desarrollo de más aplicaciones móviles o portales web por parte de personas ajenas al proyecto principal. Esto hace que Pirquet también sea en una plataforma para la identificación de las alergias en productos concretos.

Paraules clau — Alergias alimentarias, detección de alergias, código de barras, Android, página web, escaneo de productos, alimentos, productos, API, desarrolladores.

Abstract — This document explains the Pirquet project, which was created with the idea of helping people with food allergies to identify which foods can be consumed in an easy and intuitive way. Pirquet will be based on 3 main sections, two of them focused on the end user and another for possible external developers.

Paragraphs intended for the user, will be a main web site from which users can manage their allergies, view their product searches and check allergies associated with a particular product. Users will also have an Android application aimed at finding products and their allergies quickly and easily by scanning the barcode of the product with the device camera.

For developers, we will create an API with their respective documentation, aimed at developing more mobile applications or web portals by people outside the main project. This will make Pirquet also become a platform for identifying allergies in particular products.

Index Terms — Food Allergies, detecting allergies, barcode, Android, web page, scanning products, foods, products, API, developers.



1 INTRODUCCIÓN

CON el paso del tiempo, la sociedad actual está tomando conciencia de que las personas con alergias alimentarias tienen muchos problemas para poder saber qué productos pueden consumir y qué productos les pueden causar intolerancia, por este motivo, poco a poco muchos supermercados están comenzando a identificar qué productos pueden consumir las personas con dichas alergias, por ejemplo al gluten o al huevo. Aun así, hay muchos productos en los cuales no se etiqueta correctamente las diferentes posibles alergias o ni tan siquiera lo indica, por esta razón, el usuario se tiene que parar a comprobar que ingredientes contiene el producto y si puede consumir dicho producto.

Para poder ayudar a estas personas, se ha desarrollado Pirquet (Clemens Von Pirquet fue la primera persona en acuñar el término “alergia” como intolerancia a ciertos productos alimenticios, de ahí el nombre del proyecto), esta herramienta da solución al problema de los productos mal etiquetados o con información confusa. La

intención del proyecto es facilitar el escaneo del código de barras del producto, a través de la cámara de un dispositivo móvil con Android y con el código de dicho producto, y poder saber qué alergias contiene dicho producto o trazas que podría desembocar en alergias alimentarias concretas.

Las alergias alimentarias se categorizan en 8 grandes^[1] grupos:

- Alergia a los lácteos
- Alergia a los cereales
- Alergia a las legumbres
- Alergia al marisco
- Alergias a las frutas y verduras
- Alergia al huevo
- Alergia a los frutos secos
- Alergia al Pescado



- E-mail de contacte: correaroman@gmail.com
- Menció realitzada: Enginyeria del Software.
- Treball tutoritzat per: Yolanda Benítez (Ciències de la Computació)
- Curs 2015/16

Fig. 1 Representación de las diferentes Alergias a través de iconos claros.

Estas alergias pueden ser provocadas por distintos productos o por lo que se denomina trazas, que significa que dicho producto ha estado en contacto (partículas en el aire, maquinaria, tratamiento...) con otro producto que sí puede provocar dicha alergia.

2 ESTADO DEL ARTE

Antes del desarrollo de nuestro proyecto se realizó una búsqueda de información sobre otras posibles implementaciones o proyectos ya desarrollados que hicieran algo parecido a lo que nosotros planteábamos. A lo largo de las búsquedas hemos encontrado varias opciones interesantes con sus ventajas y desventajas respecto a nuestro enfoque.

5.1 IntolerApp

El primer proyecto que encontramos fue intolerapp, una aplicación para el sistema Android desarrollada por unos padres de un niño alérgico que decidieron desarrollar esta aplicación para ayudar a los padres.

Ventajas

Como ventaja principal es su amplio catálogo de productos, según ellos, tienen más de 9500 productos.

Desventajas

El mayor inconveniente de este proyecto es que está destinado a las personas alérgicas al huevo, gluten y lactosa, dejando de lado otras alergias como el pescado, el marisco o los frutos secos.

5.2 Gluten Free Food Finder

Otra alternativa encontrada fue una aplicación también para Android, con la que puedes escanear un código de barras y esta te da información sobre posibles alergias.

Ventajas

Se ayuda de la comunidad de usuarios para poder aumentar el número de productos disponibles en su base de datos

Desventajas

La comunidad en español es muy pequeña en usuarios y hay poca transparencia sobre el número de personas que votan o están de acuerdo con los datos.

5.2 Mi Intolerancia Alimentaria

Esta aplicación está desarrollada por Steffen Mauser, y la aplicación está disponible tanto en Android como iOS y tiene una página web con información.

Ventajas

Está disponible en los dos sistemas más utilizados en el mundo móvil.

Desventajas

La fiabilidad de la base de datos es muy cuestionable, y muchos usuarios se quejan de mala información o información confusa.

Aparte de las aplicaciones o servicios comentados anteriormente muchas marcas están incluyendo en su etiquetado la información referente a las alergias y a la ausencia de ingredientes que permitan el consumo de dichos alimentos por los usuarios, como por ejemplo, las etiquetas de "Sin lactosa" o la más famosa "Sin gluten".

Como se ha visto en esta sección, las alternativas que hay son escasas (nos ha costado encontrar alternativas reales), incluso los datos, en muchos casos no son totalmente veraces o confiables. Debido a esta falta de alternativas se decidió la continuación de nuestro proyecto incluyendo lo mejor de cada uno de ellos, la facilidad de uso de la aplicación a través de un escáner de código de barras y de información fiable a través de la implicación de empresas.

3 OBJETIVOS

Este proyecto está dividido en 3 grandes objetivos (API, Web y Aplicación). La página web y la aplicación son independientes entre ellos, pero en cambio, estas dos secciones dependen de la creación de la API.

3.1 Objetivo API

Para la utilización de la API, se ha desarrollado paralelamente una web en la cual se documentan todas las funciones que permite realizar la API, como se deben de realizar las llamadas, que datos devuelve la API y cómo tratar dichos datos. Esta web está destinada a los desarrolladores externos al proyecto, para que puedan utilizar la API y poder crear una gran comunidad de usuarios.

API PIRQUET

En esta web se informa de que acciones permite realizar la api PIRQUET y como utilizarla. En este documento se especifica como realizar las llamadas o que JSON enviar, que codigos de error nos devuelve la api y porque. También como te devuelve la información que has solicitado.

Elementos implementados y no implementados:
 ✓ - Implementado
 ✗ - No implementado

Usuario

Usuario - Devuelve un usuario concreto

GET

/api/usuario/{identificador}

Permisos necesarios: Ninguno

Ejemplo

http://alergia.uab.cat/api/usuario/1014242424242424

Nombre	Tipo	Descripción
298	Successful Request	La petición se a realizado correctamente
494	Not Found	No existe el elemento deseado o la petición es incorrecta

Elemento recibido (Data)

Nombre	Tipo	Descripción
identificador	String	Identificador del usuario
nombre	String	Nombre del usuario
apellido	String	Apellido del usuario
mail	String	Correo electronico del usuario
alergias	Array	Alergias del usuario

Alergias de los usuarios		
Nombre	Tipo	Descripción
tipo	String	Tipo de alergia del usuario

Fig. 2 Inicio de la página de documentación de la API.

La API se ha desarrollado para que pueda aceptar las llamadas GET, POST y PUT de la manera siguiente y con los atributos especificados a continuación:

3.1.1 Petición GET

Elemento concreto de cualquier sección de la Base de Datos. Para que la petición se realice correctamente se le ha de enviar una dirección URL en la cual se le indicará la llamada que queremos realizar y parámetro necesario para realizar la búsqueda en la base de datos.

tfg-alergia.uab.cat/api/usuario/{{Identificador}}

Una vez la API recibe la llamada tiene diferentes códigos de respuesta, que se traducen en respuesta de si la petición se ha realizado correctamente y además, envía el JSON con los datos solicitados a la API.

3.1.2 Petición POST

Esta petición está pensada para la creación de nuevos elementos (usuarios, productos, marcas...). Esto se realiza a través de una dirección URL y de un JSON en el cual se envían los datos que deberá tener la nueva instancia.

tfg-alergia.uab.cat/api/usuario/

```
{
  nombre: "Usuario",
  apellido: "De Prueba",
  contraseña: "pruebaPass",
  mail: "prueba@gmail.com"
}
```

Una vez la API recibe la llamada tiene diferentes códigos de respuesta, que se traducen en respuesta de si la petición se ha realizado correctamente y además, envía el JSON con los datos de la nueva instancia creada por la API.

3.1.2 Petición PUT

Esta petición está pensada para la modificación de elementos (usuarios, productos, marcas...). Esto se realiza a través de una dirección URL con el identificador del elemento a modificar y del JSON en el cual se envían los nuevos datos que sustituirán al actual.

tfg-alergia.uab.cat/api/usuario/{{Token}}

```
{
  nombre: "Usuario",
  apellido: "De Prueba",
  contraseña: "pruebaPass",
  mail: "prueba@gmail.com"
}
```

Una vez la API recibe la llamada tiene diferentes códigos de respuesta, que se traducen en respuesta de si la petición se ha realizado correctamente y además, envía el JSON con los datos de la nueva instancia.

También hemos añadido un sistema de tokens^[7] para el acceso de usuarios y empresas. Este sistema de tokens funciona de la siguiente manera:

Sistema de Tokens

El usuario o empresa accede al servicio, en el primer acceso, dicho usuario o empresa envía sus datos de validación, de tal manera que la API puede autenticar al usuario, una vez lo autentica, la API genera una cadena única para la sesión con una validación de 7 días y lo devuelve junto a todos los datos del usuario.

Una vez el cliente recibe los datos del usuario junto con el token^[8], este último será lo que utiliza para autenticar al usuario y realizar ciertas llamadas que implican la autenticación del usuario como modificación del perfil o la consulta del historial.

3.2 Objetivo Aplicación Android

Una vez la API ya tenía ciertas funciones mínimas necesarias (gestión de usuario, Acceso, búsqueda de productos, etc.) se inició el desarrollo de la Aplicación Android, la aplicación se ha diseñado a partir de las líneas de diseño de Google (creador de la plataforma Android), Material Design^[2]. Este diseño va más allá de lo estético, sino que también afecta a las animaciones de la aplicación^[3], uso de colores en el sistema (en la StatusBar o en la NavigationBar).

Dicha aplicación tendrá las siguientes secciones:

3.2.1 Home

Esta sección incluye las últimas novedades del servicio, ya sea la incorporación de nuevas empresas, nuevos productos o novedades destacadas dentro de la propia aplicación, ya sean actualizaciones, mejoras o novedades.

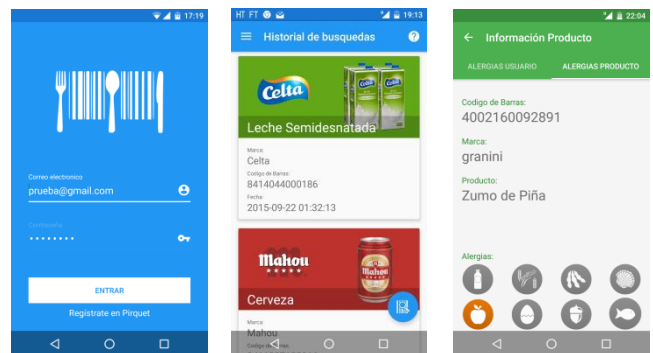


Fig. 3 Imágenes de la aplicación Android.

3.2.2 Productos

Esta sección nos aparecerá cuando escaneemos un código de barras con la cara del dispositivo. Esta sección incluye los datos básicos del producto (nombre, empresa, código de barras) así como las alergias que afectan al usuario o las propias alergias del producto.

3.2.3 Historial

Esta sección nos mostrará los productos que hemos ido buscando con anterioridad, así como cuanto hace que hemos buscado dichos productos.

3.2.4 Perfil

Desde nuestro perfil podremos visualizar ciertos elementos como nuestro nombre, apellido, correo electrónico o nuestras alergias.

La aplicación ha sido diseñada^{[4][5]}, para ser compatible con dispositivos con Android 4.0.3 (versión del SDK +15), ya que esta presenta en el 96% de los dispositivos^[6], aunque el uso recomendable es Android 5.0, ya que dicha versión es la que incluía por primera vez las líneas de diseño Material Design con la que se ha desarrollado nuestra aplicación, dicha cuota de mercado se reduce hasta el 26% de los dispositivos Android. Estos datos están tomados del último informe sobre dispositivos Android a fecha de noviembre de 2015.

Version	Codename	API	Distribution
2.2	Froyo	8	0.2%
2.3.3 - 2.3.7	Gingerbread	10	3.8%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	3.3%
4.1.x	Jelly Bean	16	11.0%
4.2.x		17	13.9%
4.3		18	4.1%
4.4	KitKat	19	37.8%
5.0	Lollipop	21	15.5%
5.1		22	10.1%
6.0	Marshmallow	23	0.3%

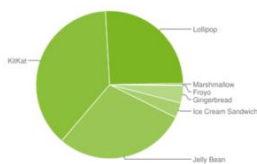


Fig. 4 Distribución de versiones Android a fecha de Noviembre de 2015^[6]

Como se puede observar en la Fig. 3, desde la aplicación Android se puede acceder al historial de búsquedas realizadas anteriormente, se pueden buscar productos, gestionar el perfil de usuario y como es evidente, acceder al escaneo de código de barras desde el botón azul situado en la zona inferior derecha de la aplicación (botón FAB) y ver la información del producto escaneado.

3.3 Objetivo Página Web

Al igual que con la aplicación, una vez finalizada las partes esenciales de la API se desarrolló la página web. Dicha web dispone de las siguientes secciones con las siguientes características y funciones:

3.3.1 Home

Esta sección incluye las últimas novedades del servicio, ya sea la incorporación de nuevas empresas, nuevos productos o novedades destacadas. También se podrá utilizar dicha función para anunciar novedades o nuevas funciones de dicha web.

3.3.2 Productos

Desde esta sección podremos acceder al buscador de marcas y productos. Este buscador nos permite introducir el nombre de la marca o producto o su identificador para una búsqueda más inmediata.

3.3.2 Historial

En el apartado podremos ver de una manera rápida las últimas búsquedas de productos que hayamos realizado tanto desde la web como desde la aplicación móvil.

3.3.4 Info

Esta sección incluirá toda la información relevante respecto al proyecto, así como aspectos legales o términos y condiciones.

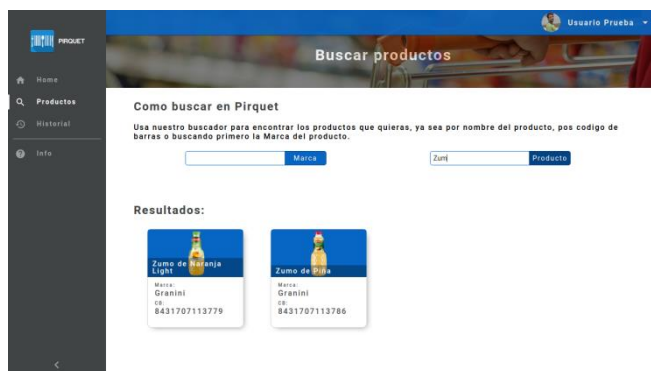


Fig. 5 Imagen de la sección de búsquedas de la web.3.3.3 Historial

3.3.5 Perfil

Desde aquí el usuario podrá modificar todos los aspectos de su cuenta, desde su nombre y apellido hasta su contraseña o alergias pasando por la imagen de perfil.

Para la página se ha utilizado web components, una tecnología en crecimiento y que, pese a tener una pequeña curva de aprendizaje, una vez adquirida esta, desarrollar elementos y secciones de la web es mucho más fácil y dinámico.

Un ejemplo de web components son las cartas utilizadas para mostrar los productos, estas cartas están desarrolladas en un fichero html aparte, y desde la sección de la web en la que queremos que aparezca dicha carta debemos referenciarlo y llamarla con el tag especialmente creado para ello "<carta-producto>" y los elementos que permite dicha carta (marca, identificador, producto, imagen o alergias).



Fig. 6 Representación de un producto en la página web.

La web se ha desarrollado pensando en crear un servicio multiplataforma y adaptable (se adapta a todas las pantallas i resoluciones), la web se adapta a todas las pantallas y/o resoluciones. Por ejemplo, en pantallas pequeñas como móviles o tablets pequeños la barra lateral se minimiza y elimina los nombres dejando solo los

iconos.

4 METODOLOGÍA

Para poder realizar los objetivos se ha utilizado una mezcla de dos metodologías. Se ha seguido el diagrama de Gantt (Apéndice 1), para establecer las fechas más significativas del proyecto, combinado con Scrum^[9], con en el cual se evaluaban los progresos en cortos periodos de tiempo para poder saber si había algún elemento que tenía que ser modificado o replanteado a raíz del desarrollo inicial.

Antes del inicio del proyecto se realizaron varios documentos básicos para poder establecer las bases de nuestro proyecto y conocer todo lo que se debía tener en cuenta antes de ponerse a desarrollar código.

A continuación, se explican los documentos previos antes del desarrollo:

4.1 Requisitos del Software

Con este documento queremos especificar todas las funcionalidades, restricciones o la calidad de los elementos a desarrollar para poder tener una base sobre la que poder trabajar. A continuación, se especifican los 3 requisitos que se especificaron (Apéndice 2) y la definición de cada una de las secciones:

4.1.1 Requisitos Funcionales

Aquí se especificaron las funcionalidades básicas que debe tener nuestro proyecto, así como especificar todo lo que debe permitir realizar nuestro sistema.

4.1.2 Requisitos de Calidad

En esta sección se especificaron cuando podríamos establecer que una funcionalidad o un elemento lo podemos considerar como concluido, esto nos marcara los objetivos finales de nuestras funcionalidades.

4.1.3 Restricciones de Desarrollo

Aquí se marcaron las limitaciones propias del proyecto, que es lo que nos limita (fecha final de entrega o horas de trabajo) y los recursos disponibles para poder desarrollar nuestro proyecto.

4.2 Diseño de la Base de Datos

Una vez establecidos los requisitos, se desarrolló del documento del diseño de nuestra base de datos, este diseño nos debía permitir la relación entre productos, empresas y alergias, así como entre usuarios, alergias y productos (Apéndice 3).

Primero se establecieron los requisitos funcionales que debía tener esta base de datos para poder saber que debe permitir la base de datos y de este modo saber las tablas que tendremos o las relaciones de la base de datos. Estos requisitos están estrechamente relacionados con el documento hecho de Requisitos del Software.

Una vez establecida los requisitos funcionales de la base de datos, se desarrolló el diagrama Entidad-Relación que nos permitirá, ya de una manera gráfica, que tablas deberemos crear y como se relacionaran entre ellas.

Finalmente, y basándonos en el diagrama E-R se desarrolló el diseño definitivo de la base de datos, este diseño ya debe de contener todas la tablas, los atributos de estas tablas, las claves primarias y las foráneas así como las relaciones.

4.3 Diagramas de Casos de Uso y de Secuencia

El último documento que se desarrolló antes de iniciar el desarrollo del producto fueron los diagramas de casos de uso y secuencia:

4.3.1 Diagramas de Casos de Uso

Con los diagramas de Casos de Uso (Apéndice 4) podemos establecer que funcionalidades son las que tenemos que desarrollar en nuestro sistema, así como que funcionalidades son las que pertenecen a cada actor (cliente o empresa) que esta relacionado con nuestro sistema.

Estos diagramas nos permiten ver las funcionalidades básicas, así como ver que funcionalidades dependen de otras.

4.3.2 Diagramas de Secuencia

Los diagramas de Secuencia (Apéndice 5) nos permiten ver la secuencia que se realizaría al realizar varias acciones en nuestro sistema, en nuestro caso hemos realizado los diagramas más representativos del sistema.

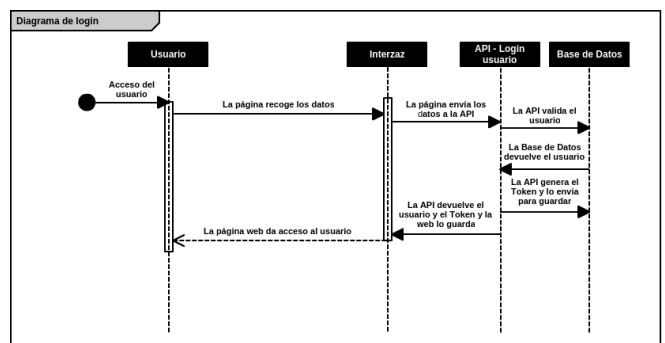


Fig. 7 Diagrama de Secuencia "Autenticación de un usuario".

Una vez se desarrollaron los documentos previos al desarrollo del proyecto se comenzó a crear la estructura de control y seguimiento del proyecto.

Para llevar a cabo un seguimiento de lo que se estaban realizando, la prioridad de las funciones que se desarrollaban o se tenían que desarrollar así como las ya desarrolladas que están pendientes de una validación final se usó la herramienta Trello. Dicha herramienta se organizó creado 5 boards (tablero en los que se organizan las tareas

que incluye el proyecto).

Estas son las 5 boards:

- **Tareas pendientes:** En esta board se almacenan todas las tareas que se tienen que crear, ya sean de la API, de la aplicación Android o de la página web.
- **Tareas en proceso:** En esta board se almacenan las tareas que se están desarrollando en ese instante.
- **Tareas en espera:** Aquí se almacenan las tareas que se han realizado y están esperando para la validación o a ser aceptadas como terminadas.
- **Tareas realizadas:** En esta board se depositan todas las tareas que ya están terminadas y se han dado por concluidas.
- **IDEAS:** En esta board se almacenan las ideas que están relacionadas con el proyecto pero no están incluidas en el desarrollo del mismo.



Fig. 8 Imagen de Trello a medio proyecto.

Como se puede observar en la figura 6, en las tareas de Trello se pueden añadir enlaces relacionados con dicha tarea, enlaces a drive para adjuntar documentos, enlaces a GitLab para enlazarlo con el código desarrollado o imágenes para mostrar el desarrollo.

Dichas tareas también están organizadas con un nivel de prioridad, que tiene 6 opciones (si hay tiempo, poco prioritario, prioritario, muy prioritario, extremadamente prioritario y pasado de tiempo), estas etiquetas se utilizaban para saber qué tareas tenían que ser priorizadas.

Por último, estas tareas también podían incluir “Checklist” para añadir subtareas e ir tachando aquellas que ya se hubieran realizado.

Aparte de Trello para la organización de las tareas que se van realizando, se ha utilizado el control de versiones de Git, para ello se aloja el código en 3 repositorios privados de GitLab. Para la organización de las versiones se usa una ramificación estándar, esto lo que hace es que podamos editar todo lo necesario sin tener que estar guardando versiones funcionales del código, todo se realizaba de una manera metódica a partir de ramas Git, lo que permite un versionado estable y la opción de recupe-

rar cualquier estado de nuestro código, ya sea en la parte estable o en la parte experimental de nuestro proyecto.

Con lo dicho anteriormente, se estableció el siguiente esquema para poder tener una organización de la ramificación producida del proyecto:

- **Master:** Es la rama de la cual saldrán el resto, en esta rama “master” tendremos todo el código general que hayamos desarrollado hasta ahora y que sabemos que funciona todo el desarrollo correctamente.
- **Feature/NombreDeLaRama:** Este será el nombre que utilizemos para crear una nueva funcionalidad, cuando queramos crear una, la denominaremos “feature/” y a continuación el nombre de la funcionalidad a implementar.
- **HotFix/NombreDeLaRama:** Este será el nombre que utilizaremos cuando queramos hacer un cambio rápido desde el master, ya sea corregir un error, bug de seguridad o correcciones menores.

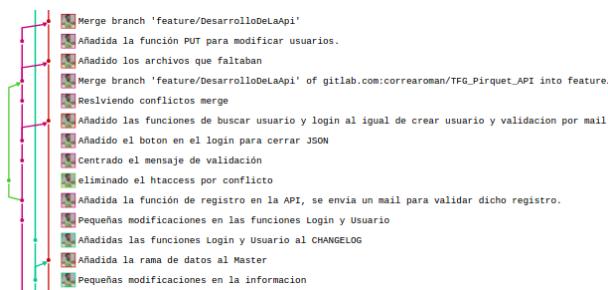


Fig. 9 Ejemplo de ramificación en uno de los repositorios.

Además del anterior sistema de ramificación mencionado, también se utiliza un sistema de versionado del código, este versionado utilizará el formato de 3 dígitos (X.Z.Y) que significa lo siguiente:

- **X:** Representa la versión estable del proyecto.
- **Z:** Representa el añadido de una funcionalidad.
- **Y:** Representa la corrección de un error rápido.

Para poder realizar un trabajo más fluido con el servidor (API y web) se decidió crear dos servidores:

Uno proporcionado por la Universidad Autónoma de Barcelona (tfg-alergia.uab.cat), en el que se alojaría todos los datos estables del proyecto y se modificarían elementos mediante Telnet y Git en las ramas master.

El otro servidor está destinado al desarrollo de pruebas (tfg.correaroman.com) en el que se podían realizar subidas en “caliente” mediante la tecnología FTP, de esta manera no afectar al funcionamiento estable del sistema.

Con estos dos servidores conseguimos que el proyecto nunca se vea afectado por las modificaciones propias del

desarrollo y la página web o la Android no fallen nunca por culpa de errores de desarrollo.

5 RESULTADOS

Al final del desarrollo de este proyecto se han podido alcanzar los objetivos marcados al inicio del mismo. Se ha desarrollado un sistema multiplataforma con la idea de ayudar a gente con alergias alimentarias a poder realizar compras o poder consultar que productos puedes causarles alergias, así como que alergias contiene dichos productos de una manera fácil, intuitiva y visual.

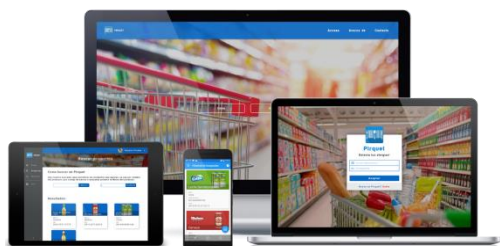


Fig. 10 Imagen multidispositivo de Pirquet.

Se ha desarrollado una API totalmente funcional y documentada (con ejemplos de uso, muestras de la devolución de resultados y códigos de estado que devuelve), se ha creado la aplicación móvil Android a partir de las líneas de diseño Material Design y con la opción de escanear el código de barras de un producto y devolver el resultado obtenido así como ver el historial de búsquedas (tanto web como aplicación) y por último se ha desarrollado la página web, desde la cual, a parte de poder realizar las acciones que permite la aplicación, también podemos modificar cualquier elemento de nuestro perfil (imagen, datos personales, contraseña o alergias), buscar productos por marca o obtener la información del proyecto (términos de uso, información básica...).

Se ha desarrollado una OpenAPI totalmente documentada que permite a cualquier desarrollador externo implementar sus propias soluciones alrededor de nuestro sistema, esto nos permite, que, aunque nuestro sistema no sea usado a través de nuestras aplicaciones o nuestras aplicaciones sean limitadas en plataformas, la comunidad pueda dar soluciones mejor diseñadas o con distintas funcionalidades. Esto hará que no solo nosotros tengamos que soportar el peso del desarrollo, sino que la comunidad aporte soluciones imaginativas o distintas a nuestras propias aplicaciones.

6 FUTURO DEL PROYECTO

Creemos que este proyecto puede ser ampliado y mejorado de muchas maneras, una idea que teníamos inicialmente, pero que no pudimos incluir en el proyecto por falta de tiempo, fue el añadir una comunidad para poder reforzar la base de datos, que sin la ayuda de la gente con alergias puede ser bastante complicada de conseguir.

Sabemos que la implicación de las empresas para la inclusión de sus productos es complicada y esto será una tarea que llevara tiempo, por lo que una manera de poder obtener datos de muchos productos en implicar a una comunidad de usuarios y que sean estos los que puedan votar las distintas alergias que puede contener un producto. A través de las votaciones que van haciendo estos usuarios, podremos saber la calidad de las votaciones de los usuarios y se podrían aplicar algoritmos de corrección para poder obtener unos datos los más confiables posibles. A raíz de la calidad de estas votaciones (comparando el número de las votaciones y la calidad de estos) los usuarios obtendrían una especie de "Karma" que influiría en el peso del voto del usuario en la alergia de ese producto.

Estos usuarios obtendrían una serie de privilegios a raíz de sus votaciones y de la calidad de estas, lo que les permitiría, por ejemplo, incluir productos a la base de datos para que la comunidad los puntuara, o por ejemplo, que los productos incluidos por estos usuarios tuvieran más visibilidad.

La idea de crear una comunidad alrededor de la aplicación es un complemento a la inclusión de los productos por parte de las empresas y ambos podrían convivir en un mismo sistema, de tal manera que, si un producto ha sido incluido por una empresa, dicha información se consideraría más fiable que la de la propia comunidad, eso sí, sin eliminar la información de la comunidad.

El tema de la comunidad haría que el proyecto involucrara más a la gente y que el uso de la aplicación o de la web por parte de las personas fuera más grande debido al uso intensivo.

Otra opción de futuro en el proyecto, es no solo limitar el uso de nuestro sistema a las alergias, sino permitir a los usuarios poder especificar qué tipo de comida deben vigilar debido a problemas alimenticios como el control de la diabetes o también, para el control de dietas.

Para la implementación de la anterior funcionalidad se añadiría los ingredientes a los productos y los usuarios podrán saber que ingredientes tiene un producto y les podrá advertir si dichos productos son convenientes a su enfermedad o a su dieta.

7 CONCLUSIÓN

Como hemos ido viendo en este informe, se ha conseguido desarrollar un servicio que nos permite saber si un producto produce alergia y poder detectar rápidamente si esa alergia es una de las nuestras. Se ha conseguido desarrollar una OpenAPI para poder fomentar el desarrollo de aplicaciones de terceros que permitan al proyecto ofrecer más aplicaciones al rededor del servicio aliviando a los desarrolladores la creación de multitud de aplicaciones en

multitud de servicios.

También se ha desarrollado una aplicación Android en la que se pueden buscar productos a través del código de barras del mismo, ver el historial de búsquedas o gestionar tu perfil de usuario. Todo esto hace que el uso del servicio sea muy ágil y atractivo, ya que el momento en el que quieres usar este tipo de servicios es en la compra, y con la aplicación facilitamos el uso y la búsqueda de alergias.

Como culminación del proyecto, se ha realizado la página web, un lugar en el que centralizar toda la información de Pirquet, donde poder buscar productos tanto por marcas como por nombres, ver tú historial de búsquedas tanto móvil como del ordenador, gestionar el perfil de usuario o que las empresas puedan añadir o modificar sus productos.

Como reflexión personal, este proyecto me ha permitido comprobar la dificultad de las personas con una alergia alimentaria a realizar una simple compra de supermercado. Es una acción en la que la gente sin ningún tipo de alergias no repara, pero una vez profundizado en el tema, puedes percartarte de las dificultades que tienen estas personas para poder simplemente cambiar de producto o marca por motivos, por ejemplo, económicos.

Este proyecto también me ha ofrecido la oportunidad de investigar y utilizar diferentes tecnologías innovadoras que a lo largo de mi vida no había podido utilizar, entre ellas los web components, una manera mucho más rápida y eficaz de programar para web. También he tenido que programar una aplicación Android, cosa que, al iniciar este proyecto, me causaba ciertas reticencias y además, la ayuda de herramientas como Android Studio también me ha facilitado la tarea crear dicha aplicación. Por último, la creación y uso de la API me ha permitido investigar sobre el uso de estas, la manera de autenticarse a través de tokens o la utilización de las llamadas GET, POST y PUT.

Aparte del uso de las nuevas tecnologías o lenguajes mencionados anteriormente, este proyecto también me ha permitido profundizar en las distintas tecnologías utilizadas a lo largo de la carrera, por ejemplo, el uso de repositorios Git, la configuración de un servidor, el desarrollo en lenguajes como Java, HTML, CSS, JavaScript, PHP o MySQL, todos lenguajes incluidos en el proyecto.

Este proyecto tiene una gran capacidad de ayudar a las personas con alergias, pero limitar a las empresas la inclusión de productos en la base de datos es algo que perjudicaría al proyecto, como he comentado en la sección "Futuro del proyecto" una gran idea para incrementar la base de datos y la participación de la gente es la de crear una gran comunidad que aporten productos y alergias, esto haría que la gente se involucrara con el servicio y lo

utilizara con más frecuencia.

8 AGRADECIMIENTOS

Para finalizar este informe, en primer lugar, quiero agradecer a Yolanda Benítez, tutora de este proyecto, por su orientación, paciencia y dedicación a lo largo de 4 meses de desarrollo.

También quiero agradecer a mi familia y amigos por el apoyo moral y por ultimo agradecer a toda la gente que ha mostrado interés a lo largo del desarrollo de este proyecto.

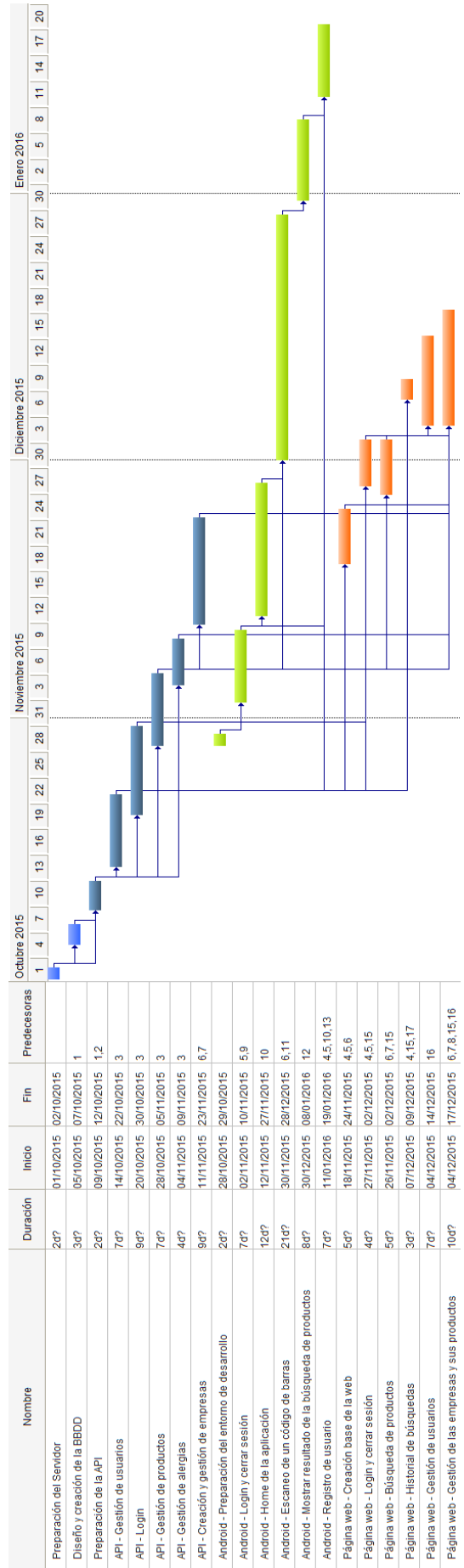
9 BIBLIOGRAFÍA

- [1] Asociación Española de personas con alergias a alimentos y látex. [Online]. Available: <http://aepnaa.org/>
- [2] Manual Material Design, Google [Online]. Available: <http://www.google.com/design/spec/material-design/>
- [3] Juhani Lehtimaki. Smashing Android UI: Responsive User Interfaces and Design Patterns for Android Phones and Tablets. Smashing Magazine, 2013.
- [4] E. Gemma, R. Helm, R. Johnson, J. Vlissides. Desing patterns: elements of reusable object oriented software. Addison Wesley, 1994 and Pearson Educacion, 2003.
- [5] E. Freeman, K. Sierra, B. Bates. Head first design patterns, O'Reilly, 2004.
- [6] Dashboards | Android Developer [Online]. Available: <http://developer.android.com/intl/es/about/dashboards/>
- [7] Carlos Azaustre. ¿Qué es la autenticación basada en Token?, 2015 [Online] Available: <https://carlosazaustre.es/blog/que-es-la-autenticacion-con-token/>
- [8] Definición de Token [Online] Available: https://es.wikipedia.org/wiki/Token_de_seguridad
- [9] Definición de Scrum [Online] Available: <https://es.wikipedia.org/wiki/Scrum>

APÉNDICE

A1. DIAGRAMA DE GANTT

Planificación realizada al inicio del proyecto, como podemos apreciar, la primera fase de desarrollo ha sido la creación de la API, una vez iniciado este apartado se comenzó con la creación de la aplicación Android y seguidamente, al tener ya comenzada dicha aplicación se inició el desarrollo de la página web.



A2. REQUISITOS DEL SOFTWARE

A.2.1 Requisitos funcionales

En este apartado se especifican las funcionalidades que debe tener nuestro sistema. Es decir, todo aquello que nuestro sistema debe realizar.

API

Nuestra API debe realizar lo siguiente:

Usuario

- Debemos poder consultar la información de un usuario concreto del sistema.
- Un usuario debe poder editar su propia información del sistema.
- Debemos poder añadir un nuevo usuario en el sistema.

Producto

- Debemos poder consultar la información de un producto en concreto del sistema.
- La empresa del producto debe poder editar la información de un producto concreto del sistema.
- Debemos poder añadir un nuevo usuario en el sistema.

Búsquedas

- Debemos poder realizar diferentes búsquedas con diferentes parámetros.
- Debemos poder realizar búsquedas de productos concretos del sistema a partir de:
 - Código de Barras.
 - Nombre del producto.
 - Identificador del producto.
- Debemos poder realizar búsquedas de empresas del sistema concretando:
 - Nombre de la empresa.
 - Identificador de la empresa.
- También tenemos que poder buscar en el sistema todos los productos de una empresa, todas las empresas o todos los productos.

Historial

- Tenemos que poder consultar todo el historial de búsquedas de un usuario del sistema.

Acceso de Usuario

- Debemos poder acceder al sistema con el mail y la contraseña del usuario.
- Debemos poder consultar si un usuario está autenticado en el sistema.

Acceso de Empresa

- Debemos poder acceder al sistema con el mail y la contraseña de la empresa.
- Debemos poder consultar si una empresa está autenticada en el sistema.

Aplicación Android

La aplicación Android debe permitir lo siguiente:

- Un usuario se tiene que poder autenticar en el sistema para acceder a la aplicación.
- Debemos proporcionar una manera de acceder a la web para registrarse en el sistema.
- Debemos poder escanear un código de barras del sistema con la cámara del móvil.

- La aplicación nos debe dar información del producto escaneado.

- En la información, debemos ver las alergias que nos afectan y las del producto.

- Tenemos que poder consultar la información del historial de los productos escaneados o vistos desde la web.

- Tenemos que poder ver la información de perfil desde la aplicación.

- Tenemos que poder cerrar la sesión del usuario autenticado.

Página Web

La página web nos debe permitir lo siguiente:

- Información básica del proyecto.
- Información sobre las condiciones y la privacidad.
- Registro de nuevos usuarios en el sistema.
- Acceso del usuario a la web.
- Página principal sobre actividad reciente del servicio.
- Página de búsqueda de información sobre productos y empresas del sistema.
- Página de información de un producto del sistema.
- Página de información de una empresa con búsqueda de productos de la misma empresa.
- Página de información sobre el historial de búsquedas de productos tanto web como aplicación.
- Página de gestión de perfil de usuario con posibilidad de modificación de:
 - Nombre
 - Apellido
 - Mail
 - Contraseña
 - Alergias
- Página de información dentro del sistema sobre privacidad y condiciones.
- Posibilidad de cerrar sesión del usuario.

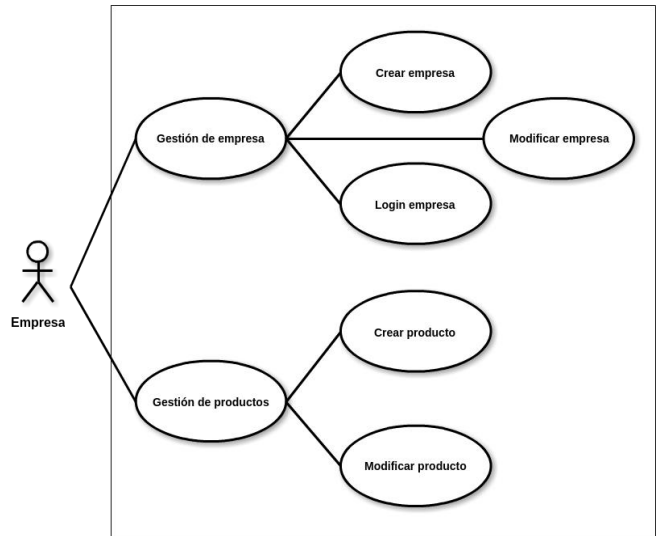
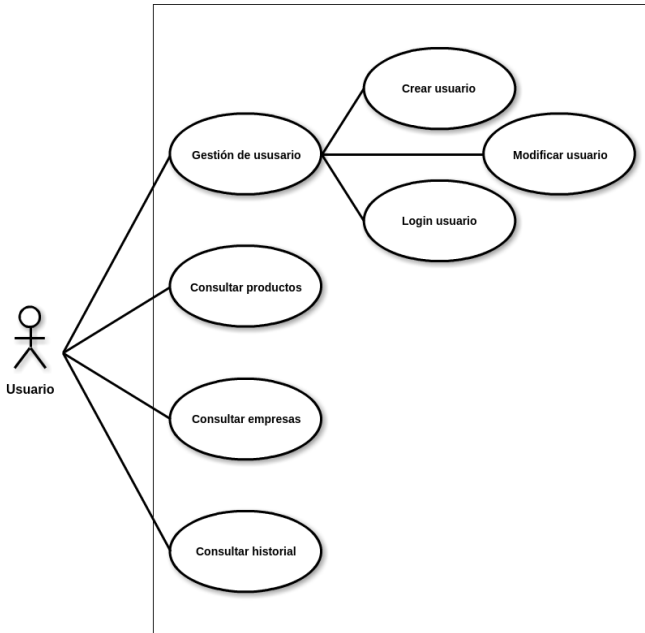
A.2.2 Restricciones del desarrollo

Restricciones del desarrollo

- Se disponen de 300 horas de desarrollo repartidas en 4 meses.
- La fecha límite de la entrega del proyecto es el 9 de febrero.
- Los recursos humanos disponibles son:
 - Estudiante que dedicará un mínimo de 300 horas
 - Profesora que se encarga de la supervisión y tutorización del proyecto
 - Los recursos técnicos son un ordenador con conexión a internet, 3 repositorios GIT donde se almacena el código, 2 servidores (uno de la autónoma como servidor estable y otro del estudiante como servidor de pruebas).

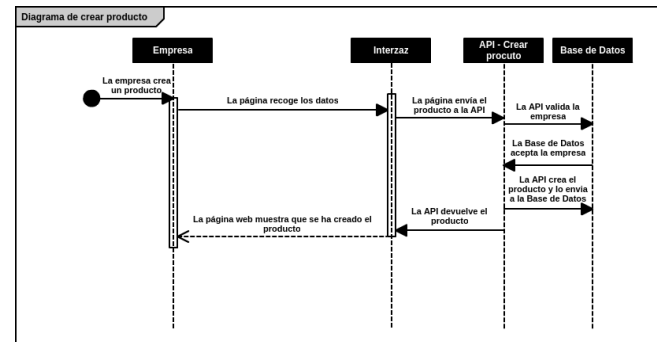
A4. DIAGRAMAS CASOS DE USO Y SECUENCIA

A.4.1 Diagrama Casos de Uso

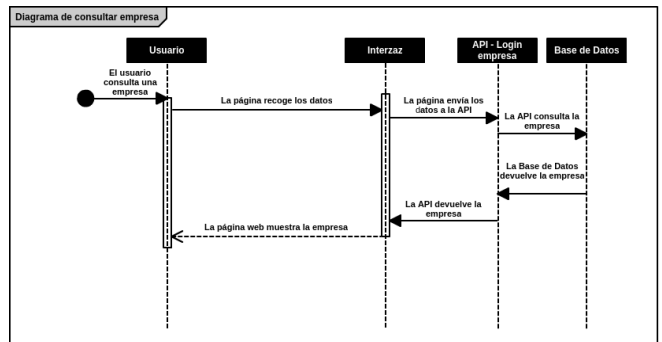


A.4.2 Diagrama de Secuencia

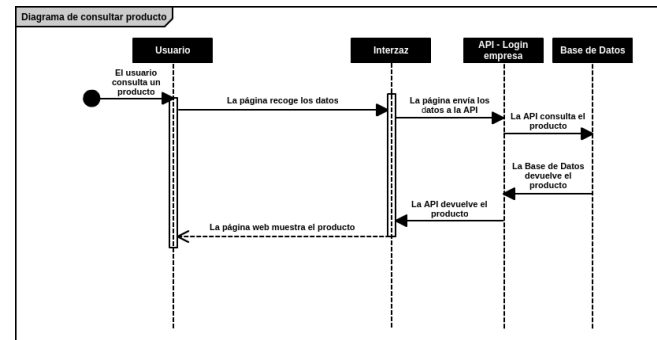
Creación de un producto



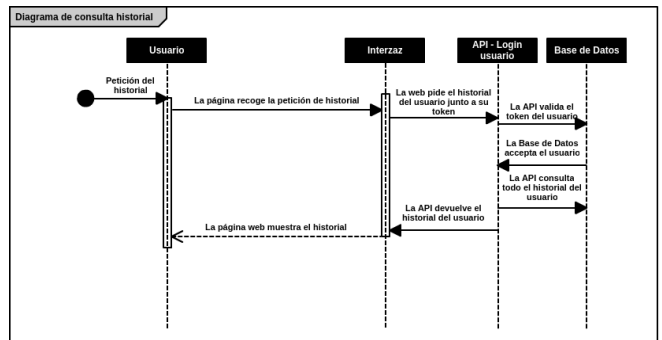
Consulta de una empresa



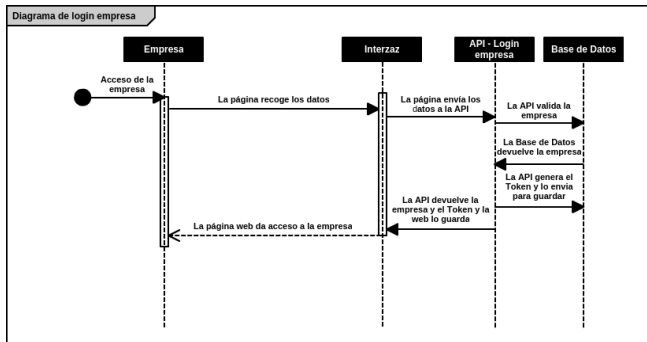
Consulta de un producto



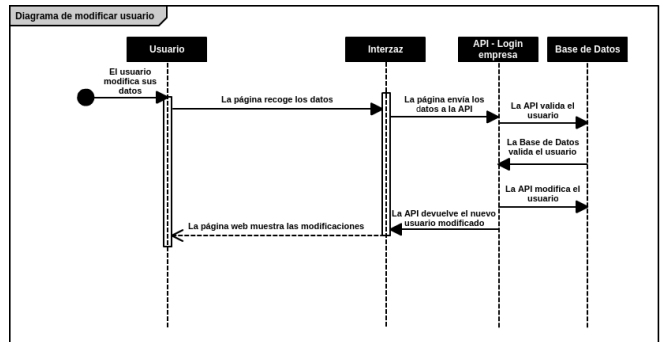
Consulta del historial de un usuario



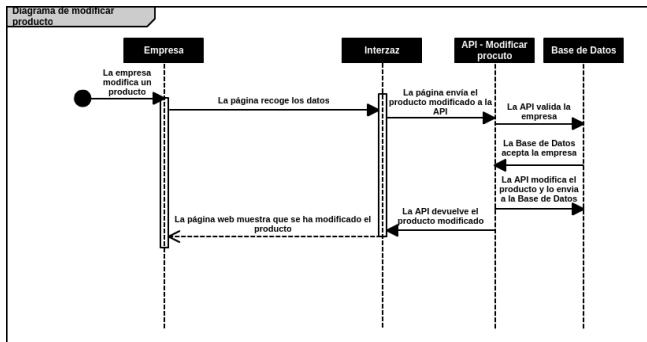
Autenticación de una empresa



Modificar un usuario



Modificar un producto



Autenticación de un usuario

