
DESARROLLO DE UN SISTEMA DE PREDICCIÓN DE VENTAS PARA TIENDAS FÍSICAS



TRABAJO DE FIN DE GRADO

JORGE MEDINA CARRETERO

Director

Antonio Sarasa Cabezuelo

Grado en Ingeniería del Software

Facultad de Informática

Universidad Complutense de Madrid

Junio 2023

DEVELOPMENT OF A SALES PREDICTION SYSTEM FOR PHYSICAL STORES

Final Degree Project Report
Software Engineering Degree

Directed by
Antonio Sarasa Cabezuelo

Jorge Medina Carretero

Faculty of Computer Science
Universidad Complutense de Madrid

June 2023

A todos mis seres queridos, hacia quienes sólo puedo expresar mi más sincero agradecimiento por apoyarme durante toda esta etapa.

Agradecimientos

En primer lugar, me gustaría agradecer este trabajo a mi tutor, Antonio Sarasa Cabezuelo, por su guía y apoyo durante todo el proyecto. En todo momento ha estado disponible para brindarme sugerencias y motivarme a alcanzar mis metas.

Deseo también expresar mi gratitud hacia mis familiares y amigos cercanos. Han estado brindándome su apoyo emocional, alentándome en los momentos más desafiantes y mostrándose dispuestos a probar el proyecto cuando ha sido necesario.

Resumen

Este trabajo tiene como objetivo proporcionar al dueño de cualquier tienda física una herramienta donde poder obtener información sobre las intenciones de compra de los clientes, ofreciendo una aplicación capaz de realizar una predicción de ventas[1] en un futuro inmediato y mostrar al vendedor en qué productos se están interesando más los clientes.

La aplicación ofrece a los usuarios un lector de QR[2] que les ayudará a conocer más acerca de los productos que les han interesado de la tienda física pudiendo acceder a detalles más relevantes del producto como una descripción detallada, el precio, o las valoraciones que otros usuarios pusieron a ese producto en la tienda.

El sistema está desarrollado en Angular[3], teniendo aplicación tanto en Web[4] como en dispositivos iOS[5] y Android[6]. Todas comparten la información añadida en una base de datos común existiendo dos tipos de cuenta, administrador y cliente.

Palabras Clave

Predicción de ventas, aplicación Android, aplicación IOs, aplicación Web, Angular, tienda

Abstract

The objective of this work is to provide the owner of any physical store with a tool where they can obtain information about the purchase intentions of customers, offering an application capable of making a sales prediction and showing the administrator user which products they are most interested in. the clients.

Users have a QR reader that will help them learn more about the products that have interested them in the physical store, being able to access more relevant details of the product such as a detailed description, the price, or the evaluations that other users have posted. that product in the store.

The system is developed in Angular, having application both on the Web and on Android and IOs devices. They all share the information added in a common database, there being two types of account, administrator and client.

Keywords

Sales prediction, Android application, IOs application, Web application, Angular, store.

Índice

Dedicatoria	4
Agradecimientos	5
Resumen	6
Palabras Clave	6
Abstract	7
Keywords	7
1. Introducción	17
1.1. Motivación	17
1.2. Objetivos	18
1.3. Metodología	19
1.3.1. Fases del proyecto	19
1.3.2. Organización del trabajo	22
1.4. Estructura de la memoria	22
1. Introduction	25
1.1. Motivation	25
1.2. Objective	25
1.3. Methodology	26
1.3.1. Project phases	27
1.3.2. Work organization	28
1.4. Structure of the memory	29
2. Estado del arte	31
2.1 Tableau	31
2.2 Salesforce	31
2.3 Microsoft Dynamics 365	31
2.4 Oracle Sales Cloud	32
3. Tecnología empleada	33
3.1. Angular	33
3.2. Typescript	33
3.3. HTML5, CSS y Bootstrap	34
3.4. Firebase	34
3.5. Capacitor	35
3.6. Ngx-Scanner	35

3.7. Highcharts	35
3.8. Visual Studio Code	35
3.9. Github	36
3.10. Trello	36
4. Especificación de requisitos	37
4.1. Actores	37
4.2. Diagramas de casos de uso	38
4.3. Especificación de requisitos	39
4.3.1 Cuentas de usuario	40
4.3.2 Gestión de productos	45
4.3.3 Gestión de ventas	48
4.3.4 Informes de productos	52
4.3.5 Servicios orientados al usuario	56
5. Arquitectura de la aplicación	62
5.1. Estructura de la aplicación	62
5.2. Modelo de datos	64
5.2.1. Colección Usuarios	65
5.2.2. Colección Productos	66
5.2.3. Colección Ventas	66
5.2.4. Colección Encuestas	67
6. Implementación y diseño de la aplicación	69
6.1. Módulo Cuentas de Usuario	69
6.1.1. Registrar usuario	69
6.1.2. Login	71
6.1.3. Logout	73
6.1.4. Dar de baja usuario	74
6.1.5. Modificar datos de usuario	77
6.2. Módulo Gestión de Productos	78
6.2.1. Registrar un producto	78
6.2.2. Dar de baja un producto	80
6.2.3. Modificar un producto	82
6.2.4. Eliminar valoración de un producto o un comentario en el foro	84
6.3. Módulo Gestión de Ventas	86
6.3.1. Registrar ventas de un producto	86
6.3.2. Modificar número de ventas de un producto	88
6.3.3. Registrar devolución de un producto	90
6.3.4. Mostrar caja de tienda	92

6.4. Módulo Informes de Productos	92
6.4.1. Informe sobre los productos más escaneados y con relación a las ventas, edad o género	92
6.4.2. Generar/Modificar/Publicar encuesta para clientes	101
6.4.3. Buscar y ver resultados de la encuesta	106
6.5. Servicios Orientados al Usuario	108
6.5.1. Realizar lectura de QR de un producto	108
6.5.2. Añadir/Quitar producto de favoritos	109
6.5.3. Consultar productos comprados	111
6.5.4. Realizar encuesta	111
6.5.5. Añadir/Eliminar/Modificar valoración en foro	113
7. Evaluación de la aplicación	115
7.1. Diseño de la evaluación	115
7.2. Resultados de la encuesta	116
8. Conclusiones y trabajo futuro	123
8.1. Conclusiones	123
8.2. Trabajo futuro	124
9. Conclusions and future work	126
Conclusions	126
Future work	126
10. Bibliografía	129
A. Guía de uso	131
A.1. Vista principal	131
A.1.1 Usuario	131
A.1.2 Administrador	132
A.2. Catálogo	135
A.2.1. Usuario	135
A.2.2. Administrador	141
A.3. Ventas	143
A.3.1. Usuario	143
A.3.2. Administrador	143
A.4. Encuestas	145
A.4.1. Usuario	145
A.4.1. Administrador	147
A.5. Perfil	152
A.5.1. Usuario	152

Índice de figuras

Figura 1.1: Trello	18
Figura 4.1: Diagrama de casos de uso	35
Figura 5.1: Arquitectura de la aplicación	60
Figura 5.2: Modelo de datos	62
Figura 5.3: Colección Usuarios	62
Figura 5.4: Colección Productos	63
Figura 5.5: Colección Ventas	64
Figura 5.6: Colección Encuestas	65
Figura 6.1: Registrar usuario	67
Figura 6.2: newRegister()	68
Figura 6.3: dataService.saveUser()	68
Figura 6.4: Login	69
Figura 6.5: logIn()	70
Figura 6.6: Botón desplegable	70
Figura 6.7: Barra de navegación	71
Figura 6.8 : logOut()	71
Figura 6.9: Administrar perfiles	72
Figura 6.10: Modificar perfil	73
Figura 6.11: deleteUser()	73
Figura 6.11: dataService.deleteUser()	74
Figura 6.12: updateUser()	74
Figura 6.13: dataService.updateUser()	75
Figura 6.14: Catálogo de productos	76
Figura 6.15: Registrar producto	76
Figura 6.16: addNewProduct()	77
Figura 6.17: dataService.saveProduct()	77
Figura 6.18: Baja de producto	78

Figura 6.19: deleteProduct()	79
Figura 6.20: dataService.deleteProduct()	79
Figura 6.21: Modificar Producto	80
Figura 6.22: updateProduct()	81
Figura 6.23: Valoraciones	81
Figura 6.24: Eliminar Valoración	82
Figura 6.25: deleteReview()	82
Figura 6.26: dataService.updateProduct()	82
Figura 6.27: Registro de Ventas	83
Figura 6.28: Registro de nueva venta	84
Figura 6.29: registerNewSale()	84
Figura 6.30: dataService.saveSale()	85
Figura 6.31: Modificar venta	86
Figura 6.32: modifySale()	87
Figura 6.33: dataService.updateSale()	87
Figura 6.34: Eliminar venta	88
Figura 6.35: deleteSale()	88
Figura 6.36: dataService.deleteSale()	88
Figura 6.37: Filtro de Ventas	89
Figura 6.38: Predicción de ventas	91
Figura 6.39: Predicción de ventas()	91
Figura 6.40: Gráfico de visitas	92
Figura 6.41: linechart	93
Figura 6.42: loadChart()	93
Figura 6.43: yearSales()	93
Figura 6.44: yearUnitsSold()	94
Figura 6.45: runRateMethod()	95
Figura 6.46: seasonalMethod()	96
Figura 6.47: lastYearSeasonalMethod()	97
Figura 6.48: Amazon Seller	98

Figura 6.49: Encuestas	99
Figura 6.50: Añadir título de la encuesta	99
Figura 6.51: Nueva encuesta	100
Figura 6.52: Añadir Campo	100
Figura 6.53: Publicar Encuesta	101
Figura 6.54: publishFeedback()	102
Figura 6.55: updateUser()	102
Figura 6.56: Respuestas	103
Figura 6.57: Respuestas de usuarios encuestados	104
Figura 6.58: Lector QR	105
Figura 6.59: Escanear Producto	106
Figura 6.60: onCodeDecode()	106
Figura 6.61: Favoritos	107
Figura 6.62: addFavouriteProduct()	107
Figura 6.63: Productos Comprados	108
Figura 6.64: Encuestas	108
Figura 6.65: Responder encuesta	109
Figura 6.66: formResponse()	109
Figura 6.67: Valoraciones	110
Figura 6.68: Añadir valoración	111
Figura 7.1: ¿Cuál es tu edad?	113
Figura 7.2: ¿Crees que una herramienta de predicción de ventas es útil para un negocio?	114
Figura 7.3: Nivel de satisfacción con la forma de mostrar el catálogo	114
Figura 7.4: Nivel de satisfacción con la página de detalles del producto	115
Figura 7.5: Nivel de satisfacción con el scanner de código QR de la aplicación	116
Figura 7.6: Nivel de satisfacción con la pantalla perfil	117
Figura 7.7: Nivel de satisfacción con la pantalla de ventas	117
Figura 7.8: Nivel de satisfacción con la realización de encuestas	118
Figura 7.9: Nivel de satisfacción general con la aplicación	119

Figura A.1: Lector QR	129
Figura A.2: Escanear producto	129
Figura A.3: Predicción de ventas	130
Figura A.4: Gráfico de ventas	130
Figura A.5: Modo de gráfico	131
Figura A.6: Producto	132
Figura A.7: Mando FireTv	133
Figura A.8: Detalles del producto	134
Figura A.9: Código QR de producto	135
Figura A.10: Registrar Compra	135
Figura A.11: Valoraciones	136
Figura A.12: Registrar Valoración	136
Figura A.13: Modificar Valoración	137
Figura A.14: Eliminar Valoración	137
Figura A.15: Producto con cuenta de administrador	138
Figura A.16: Registrar venta de producto con cuenta de administrador	139
Figura A.17: Valoraciones con cuenta de administrador	139
Figura A.18: Productos comprados	140
Figura A.19: Ventas como usuario administrador	141
Figura A.20: Registrar nueva venta como usuario administrador	141
Figura A.21: Aviso de Encuesta	142
Figura A.22: Encuestas pendientes	143
Figura A.23: Responder encuesta	144
Figura A.24: Encuestas	145
Figura A.25: Nueva encuesta	146
Figura A.26: Publicar Encuesta	147
Figura A.27: Respuestas	147
Figura A.28: Respuestas de usuario	148
Figura A.29: Perfil	149
Figura A.30: Modificar Perfil	150

Figura A.31: Borrar cuenta	150
Figura A.32: Administrar Cuentas	151
Figura A.33: Administrar Perfiles	152
Figura A.34: Modificar Perfil	152

Capítulo 1

Introducción

A continuación se explican los motivos que llevaron a la realización de este trabajo así como las necesidades que se pretenden cubrir para los usuarios. Posteriormente, se descubrirán los objetivos específicos que abarca el proyecto, la metodología de trabajo que se ha seguido junto con las fases en las que se ha dividido el proyecto y la forma en la que se ha organizado.

1.1. Motivación

El mundo del comercio es cada vez más competitivo y los vendedores están constantemente buscando formas de mejorar sus ventas y la eficiencia de su negocio. Una herramienta de predicción de ventas puede ser una solución valiosa para abordar los desafíos que supone llevar un negocio basado en la venta de productos.

Con la ayuda de una herramienta de predicción de ventas, los vendedores pueden planificar mejor su inventario, gestionar sus costes y ajustar sus precios para maximizar sus ventas. Esto puede mejorar la eficiencia general de la tienda y ayudar a los vendedores a reducir los costes o descartar aquellos productos que resultan defectuosos, de baja calidad o poco interesantes para los compradores. Los vendedores pueden ofrecer una experiencia de compra más personalizada para mejorar la satisfacción del cliente y aumentar la fidelidad.

Esta herramienta le permite al vendedor conocer las opiniones de los clientes en función de los productos que han escaneado, saber qué productos se han escaneado más veces, conocer la opinión de los clientes a través de encuestas, escanear productos para conocer más información acerca de estos, permite a los clientes valorar los productos que compran y ver otras valoraciones y finalmente, crear gráficos explicativos basados en la actividad de los clientes en la tienda para que el vendedor pueda sacar sus propias conclusiones.

1.2. Objetivos

El objetivo principal es el desarrollo de un sistema que permita al vendedor conocer la actividad del cliente con respecto a sus productos para poder decidir cuáles van a ser los productos que debe mantener en su tienda y cuáles no, a parte de poder anticipar las compras de inventario de cada producto para obtener una rentabilidad máxima.

Este objetivo se particulariza en los siguientes puntos:

- Gestionar productos: La herramienta contiene un inventario de productos “Catálogo” donde pueden añadirse, modificarse o eliminarse del catálogo. En el catálogo aparecen todos los productos y es posible buscar productos con la barra buscador.
- Leer QRs de productos: Existe un lector de QR que detecta los QR de los productos registrados en el catálogo, lo que le permite al usuario interactuar con los productos de la tienda física y poder acceder a información detallada sobre cada producto. A su vez, el sistema recogerá cada visita realizada por un usuario a un nuevo producto.
- Añadir producto a favoritos: El usuario puede añadir los productos que considere a su lista de favoritos o sacarlos de la lista.
- Valorar productos: Los usuarios pueden añadir, modificar o eliminar comentarios con una valoración en cada producto y ver los comentarios de otros usuarios.
- Gestionar ventas: El administrador puede registrar las ventas que han realizado los clientes.
- Realizar informes con gráficos dinámicos que muestran datos relevantes sobre los clientes como una predicción de ventas, una predicción de stocks y un gráfico de visualizaciones de cada producto
- Realizar encuestas y publicarlas para que los usuarios respondan lo que les ha parecido la experiencia en la tienda o su opinión acerca de un producto determinado

1.3. Metodología

El desarrollo del proyecto se ha realizado de acuerdo a un plan de proyecto y a una organización del trabajo que se explica en los siguientes apartados.

1.3.1. Fases del proyecto

Para desarrollar el proyecto, se hizo una planificación que incluyó la toma de requisitos, desarrollo de las aplicaciones, pruebas y el desarrollo de la memoria. La tabla 1.1 muestra las actividades realizadas en cada periodo, así como la representación visual mediante trello en la [figura 1.1](#), también se puede acceder al propio trello haciendo en este enlace: [Trello](#).

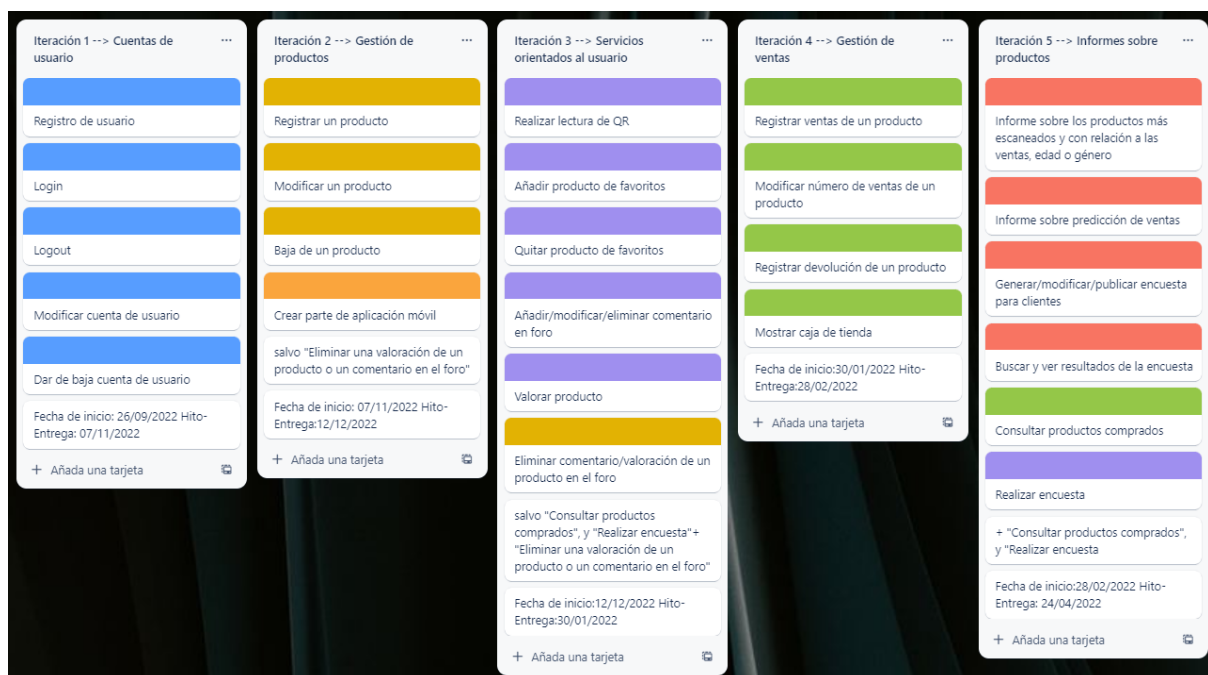


Figura 1.1: Trello

Etapa	Duración
Cuentas de usuario	26 Septiembre - 7 Noviembre
Gestión de productos	7 Noviembre - 12 Diciembre
Servicios orientados al usuario	12 Diciembre - 30 Enero
Gestión de ventas	30 Enero - 28 Febrero
Informes sobre productos	28 Febrero - 24 Abril
Desarrollo de la memoria	24 Abril - 16 Mayo

Tabla 1.1: Etapas en el desarrollo del proyecto

A continuación se procederá a explicar cada una de las fases en las que se ha segmentado el desarrollo del proyecto:

Iteración 1: Cuentas de usuario

Esta primera fase consistió en la realización de las pantallas de registro y de login de usuario. Además de la funcionalidad de Logout y modificar la cuenta de usuario o dar de baja otras cuentas de usuario como usuario administrador.

Iteración 2: Gestión de productos

Esta fase se basó en la creación de un catálogo con productos para que el comprador pudiese conocer todos los productos de la tienda física. Aparte de las funcionalidades de registrar un nuevo producto, modificarlo o darlo de baja para el usuario administrador. Por último se acordó añadir que la aplicación fuese no solo Web si no también móvil.

Iteración 3: Servicios orientados al usuario

Esta iteración añade funcionalidades de cara al usuario, poder realizar lecturas de QR de los productos para poder acceder a los detalles de estos, añadir un producto a la lista de favoritos o realizar valoraciones sobre los productos.

Iteración 4: Gestión de ventas

Registrar ventas de un producto, poder modificar las ventas registradas, poder ver la caja de tienda o poder eliminar una venta en caso de devolución son las funcionalidades que engloba este hito.

Iteración 5: Informes sobre productos

En esta iteración se añadieron las funcionalidades más importantes de la aplicación, las cuales permiten al administrador crear encuestas y publicarlas para que los clientes puedan responderlas y acceder a sus respuestas. Y también una predicción de ventas, stocks y los productos más populares de la tienda física, en forma de gráficos de barras y gráficos por sectores.

1.3.2. Organización del trabajo

Durante la realización del proyecto se ha mantenido una metodología de trabajo que ha mezclado cascada con iteraciones e incrementos y relativa flexibilidad. Estas iteraciones han variado en su duración yendo desde un mes hasta dos aproximadamente, con una posterior presentación del trabajo realizado. Se estimaron al principio los tiempos de entrega de cada iteración y se fue realizando acorde a estas pautas, días antes de la presentación decidíamos si había trabajo suficiente como para llevarse a cabo, pudiéndose posponer las presentaciones la cantidad de tiempo necesaria hasta que estuviesen totalmente presentables.

Debido a que el trabajo se estimó de esta manera, se hizo uso de una herramienta de Trello en la que poder plasmar todas las iteraciones, que permitió tener un seguimiento tanto para mí como para el director del proyecto del estado de cada una de las incidencias y funcionalidades que se estaban implementando. Por otro lado, se empleó GitHub[7] como sistema de control de versiones para poder trabajar con un repositorio y ser capaz de llevar un control de versiones para llevar un desarrollo del proyecto de una forma más segura.

1.4. Estructura de la memoria

La presente memoria se estructura en diversas secciones con el objetivo de brindar una visión completa del trabajo realizado. A continuación, se describe la organización de cada sección:

Introducción: En esta sección se introduce el trabajo, proporcionando una descripción general del proyecto, su contexto y su importancia. Se abordan la motivación del proyecto, explicando las razones que llevaron a su desarrollo, y los objetivos, que definen los resultados esperados. Además, se presenta la metodología utilizada, explicando el enfoque y las técnicas empleadas en la realización del trabajo. Se detallan también las fases del proyecto, que muestran la secuencia de actividades realizadas, y la organización del trabajo, que describe cómo se estructuró y distribuyó el equipo de trabajo.

Estado del arte: En esta sección se realiza un estudio del estado actual de la tecnología y las herramientas relevantes para el proyecto, buscando aplicaciones similares que ofrezcan funcionalidades parecidas a las de nuestra aplicación. Se incluyen descripciones de Tableau, Salesforce, Microsoft Dynamics 365 y Oracle Sales Cloud.

Tecnología empleada: En esta sección se presenta y explica la tecnología utilizada en el desarrollo del proyecto. Se describen y se detallan las herramientas, lenguajes de programación y frameworks empleados. Se incluyen elementos como Angular, Typescript, HTML5, CSS y Bootstrap, que son utilizados para la construcción de la interfaz de usuario y la maquetación. También se mencionan tecnologías backend como Firebase, que se utiliza para el almacenamiento y gestión de datos, y Capacitor, que permite la compilación y distribución de la aplicación en diferentes plataformas. Además, se mencionan bibliotecas y herramientas complementarias como Ngx-Scanner, Highcharts, Visual Studio Code, Github y Trello, que contribuyen al desarrollo, gestión y seguimiento del proyecto.

Especificación de requisitos: En esta sección se detallan los requisitos del sistema. Se identifican y describen los actores involucrados en el proyecto, especificando sus roles y responsabilidades. Luego, se presentan los diagramas de casos de uso, que muestran las interacciones entre los actores y el sistema. Finalmente, se proporciona una especificación detallada de los requisitos, dividiéndolos en diferentes secciones. Esto incluye la gestión de cuentas de usuario, que describe las funcionalidades relacionadas con el registro, inicio de sesión y gestión de perfiles de usuario. También se especifica la gestión de productos, que detalla las funcionalidades relacionadas con la creación, edición y eliminación de productos. La gestión de ventas describe las funcionalidades relacionadas con la realización y seguimiento de ventas, mientras que los informes de productos muestran las funcionalidades relacionadas con la generación y visualización de informes estadísticos. Por último, se mencionan los servicios orientados al usuario, que describen funcionalidades adicionales que mejoran la experiencia del usuario.

Arquitectura de la aplicación: En esta sección se describe la arquitectura general de la aplicación. Se presenta la estructura de la aplicación, detallando la organización de los componentes y la navegación entre las diferentes secciones. Además, se explica el modelo de datos utilizado, que define la estructura y las relaciones entre las diferentes entidades del

sistema. Se describen las colecciones de Usuarios, Productos, Ventas y Encuestas, indicando los campos y atributos relevantes para cada entidad.

Implementación y diseño de la aplicación: En esta sección se aborda la implementación y el diseño de la aplicación en detalle. Se describe el módulo de Cuentas de Usuario, explicando las funcionalidades relacionadas con el registro de usuarios, el inicio de sesión y la salida de sesión. Se detallan los procesos, las interfaces y las interacciones relevantes para cada funcionalidad, ofreciendo una visión completa de la implementación y el diseño de dicho módulo.

Introduction

The reasons that led to the realization of this work are explained below, as well as the needs that are intended to be covered by the users. Subsequently, the specific objectives covered by the project will be discovered, the work methodology that has been followed along with the phases into which the project has been divided and the way in which it has been organized.

Motivation

The world of commerce is increasingly competitive and sellers are constantly looking for ways to improve their sales and the efficiency of their business. A sales forecasting tool can be a valuable solution to address the challenges of running a product-based business.

With the help of a sales forecasting tool, sellers can better plan their inventory, manage their costs, and adjust their prices to maximize their sales. This can improve overall store efficiency and help retailers cut costs or weed out products that are faulty, low-quality, or uninteresting to shoppers. Sellers can offer a more personalized shopping experience to improve customer satisfaction and increase loyalty.

This tool allows the seller to create accounts to know the opinions of customers based on the products they have scanned, know which products have been scanned the most times, scan products to know more information about them, and allows customers to rate the products. products they buy and see other evaluations and finally, create explanatory graphics based on the activity of customers in the store in relation to the products so that the seller can draw their own conclusions.

Objective

The main objective is the development of a system that allows the seller to know the activity of the client with respect to his products to be able to decide which are going to be the

products that he must keep in his store and which not, apart from being able to anticipate the purchases of inventory of each product for maximum profitability.

This objective is specified in the following points:

- Manage products: The tool contains an inventory of products "Catalog" where they can be added, modified or removed from the catalog. All the products appear in the catalog and it is possible to search for products with the search bar.
- Read product QRs: There is a QR reader that detects the QRs of the products registered in the catalogue, which allows the user to interact with the products in the physical store and access detailed information about each product.
- Add product to favorites: The user can add the products he considers to his favorites list or remove them from the list.
- Evaluate products: Users can add, modify or delete comments with an evaluation on each product and see the comments of other users.
- Manage Sales: The admin can record the sales that the customers have made.
- Create reports with dynamic graphs that show relevant data about customers such as a sales prediction, a stock prediction and a graph of views of each product.
- Carry out surveys and publish them so that users answer what they thought of the experience in the store or their opinion about a specific product.

Methodology

El desarrollo del proyecto se ha realizado de acuerdo a un plan de proyecto y a una organización del trabajo que se explica en los siguientes apartados.

Project phases

To develop the project, a planning was made that included the gathering of requirements, development of the applications, tests and the development of the memory. Table 1.1 shows the activities carried out in each period, as well as the visual representation through trello in figure 1.1 Trello, which can also be accessed by clicking [here](#).

The following is an explanation of the phases into which the development of the project has been segmented.

Iteration 1: User accounts

This first phase consisted of carrying out the registration and user login screens. In addition to the functionality of Logout and modify the user account or cancel other user accounts as an administrator user.

Iteration 2: Product Management

This phase was based on the creation of a catalog with products so that the buyer could see all the products in the physical store. Apart from the functionalities of registering a new product, modifying it or unsubscribing it for the administrator user. Finally it was agreed to add that the application was not only Web but also mobile.

Iteration 3: User-oriented services

This iteration adds functionalities for the user, being able to read the QR of the products to be able to access their details, add a product to the list of favorites or make evaluations of the products.

Iteration 4: Sales Management

Registering sales of a product, being able to modify registered sales, being able to see the store checkout or being able to delete a sale in the event of a return are the functionalities included in this milestone.

Iteration 5: Product reports

In this iteration, the most important functionalities of the application were added, which allow the administrator to create surveys and publish them so that customers can answer them and access their responses. And also a prediction of sales, stocks and the most popular products in the physical store, in the form of bar charts and pie charts.

Work organization

During the realization of the project, a work methodology has been maintained that has mixed cascade with iterations and increments and relative flexibility. These iterations have varied in duration, ranging from one month to approximately two, with a subsequent presentation of the work done. The delivery times of each iteration were estimated at the beginning of the project and it was carried out according to these guidelines. Days before the presentation we decided if there was enough work to be carried out, being able to postpone the presentations the necessary amount of time until they were fully presentable .

Because the work was estimated in this way, a Trello tool was used in which to record all the iterations, which allowed both me and the project manager to track the status of each of the incidents. and functionalities that were being implemented. On the other hand, GitHub was used as a version control system to be able to work with a repository and be able to control versions to carry out project development in a more secure way.

Structure of the memory

This report is structured into various sections with the aim of providing a complete overview of the work carried out. The organization of each section is described below:

Introduction: This section introduces the work, providing a general description of the project, its context and its importance. The motivation of the project is addressed, explaining the reasons that led to its development, and the objectives, which define the expected results. In addition, the methodology used is presented, explaining the approach and techniques used in carrying out the work. The project phases are also detailed, showing the sequence of activities carried out, and the work organization, which describes how the work team was structured and distributed.

State of the art: In this section, a study of the current state of technology and relevant tools for the project is carried out, looking for similar applications that offer similar functionalities to those of our application. Descriptions of Tableau, Salesforce, Microsoft Dynamics 365, and Oracle Sales Cloud are included.

Technology used: This section presents and explains the technology used in the development of the project. The tools, programming languages and frameworks used are described and detailed. Elements such as Angular, Typescript, HTML5, CSS and Bootstrap are included, which are used for building the user interface and layout. Backend technologies such as Firebase, which is used for data storage and management, and Capacitor, which allows the compilation and distribution of the application on different platforms, are also mentioned. In addition, complementary libraries and tools such as Ngx-Scanner, Highcharts, Visual Studio Code, Github and Trello are mentioned, which contribute to the development, management and monitoring of the project.

Requirements Specification: This section details the system requirements. The actors involved in the project are identified and described, specifying their roles and responsibilities. Then, the use case diagrams are presented, which show the interactions between the actors and the system. Finally, a detailed specification of the requirements is provided, dividing them into different sections. This includes user account management, which describes

functionalities related to registration, login and management of user profiles. Product management is also specified, which details the functionalities related to the creation, edition and deletion of products. Sales management describes the functionalities related to making and monitoring sales, while product reports show the functionalities related to the generation and visualization of statistical reports. Finally, user-oriented services are mentioned, which describe additional functionalities that improve the user experience.

Application Architecture: This section describes the general architecture of the application. The structure of the application is presented, detailing the organization of the components and the navigation between the different sections. In addition, the data model used is explained, which defines the structure and relationships between the different entities of the system. The Users, Products, Sales and Surveys collections are described, indicating the relevant fields and attributes for each entity.

Application Design and Implementation: This section covers application design and implementation in detail. The User Accounts module is described, explaining the functionalities related to user registration, login and session exit. The relevant processes, interfaces, and interactions for each functionality are detailed, offering a comprehensive overview of the implementation and design of that module.

Capítulo 2

Estado del arte

En este capítulo se describen algunas aplicaciones con funcionalidades similares al sistema desarrollado en este proyecto.

2.1 Tableau

Tableau[8] es una aplicación de visualización de datos que puede ayudar a los vendedores a analizar sus datos de ventas y predecir las tendencias futuras. La aplicación es fácil de usar y tiene una amplia variedad de herramientas de análisis y visualización.

2.2 Salesforce

Salesforce[9] es una plataforma de gestión de relaciones con los clientes que también tiene una función de predicción de ventas. La aplicación utiliza inteligencia artificial y aprendizaje automático para analizar los datos de ventas y hacer predicciones precisas sobre las ventas futuras.

2.3 Microsoft Dynamics 365

Microsoft Dynamics 365[10] es una solución empresarial completa que incluye una función de predicción de ventas. La aplicación utiliza algoritmos avanzados de aprendizaje automático para analizar los datos de ventas y hacer predicciones precisas sobre las ventas futuras.

2.4 Oracle Sales Cloud

Oracle Sales Cloud[[11](#)] es una aplicación de gestión de ventas que también tiene una función de predicción de ventas. La aplicación utiliza análisis predictivos para ayudar a los vendedores a predecir las tendencias de ventas y optimizar su estrategia de ventas.

Capítulo 3

Tecnología empleada

En este capítulo se describen las principales herramientas y tecnologías empleadas para desarrollar este proyecto.

3.1. Angular

Angular[3] es un framework de desarrollo web de código abierto[12] creado por Google. Es una herramienta muy popular para construir aplicaciones web dinámicas y escalables. Angular se basa en el lenguaje de programación TypeScript, que es un superset de JavaScript[13] y ofrece características adicionales, como tipos de datos estáticos, interfaces y clases. Angular utiliza una arquitectura de componentes para construir aplicaciones web, lo que significa que las aplicaciones se dividen en pequeños componentes que pueden ser reutilizados y ensamblados para construir aplicaciones más grandes. Angular ha sido la herramienta principal para desarrollar todo el proyecto ya que su escalabilidad y su programación por componentes se adaptan a la perfección a las iteraciones e incrementos de la aplicación.

3.2. Typescript

TypeScript[14] es un lenguaje de programación libre y de código abierto que se basa en JavaScript y es desarrollado y mantenido por Microsoft. Fue diseñado para ser una alternativa a JavaScript que ofreciera una tipificación estática, lo que significa que los tipos de datos pueden ser especificados en tiempo de compilación en lugar de inferidos en tiempo de ejecución como ocurre en JavaScript. Esto ayuda a detectar errores de programación temprano y mejora la calidad y mantenibilidad del código.

Además de la tipificación estática, TypeScript también ofrece otras características como clases, interfaces, enumeraciones, herencia, y una sintaxis más clara y concisa que JavaScript, lo que hace que el código sea más fácil de leer y mantener. TypeScript es compatible con

todas las características de JavaScript, lo que significa que cualquier código JavaScript existente se puede integrar fácilmente en un proyecto TypeScript.

3.3. HTML5, CSS y Bootstrap

HTML5[15] es la última versión del lenguaje de marcado básico en la web, HTML. Posee una sintaxis básica, basada en etiquetas, añadiendo esta versión nueva a las ya existentes para satisfacer las necesidades de las páginas web modernas. CSS[16] es el lenguaje de estilo utilizado en una página web, en el que se declaran una serie de reglas que describen cómo se mostrará un elemento elegido mediante un selector, que podrá ser un identificador o una clase.

Por último, Bootstrap[17] es una librería creada para facilitar el uso de las dos tecnologías mencionadas previamente, dado que esta ofrece componentes plantilla con un estilo ya añadido. En esta aplicación se utilizan las 3 tecnologías en la aplicación de gestión de contenido para la maquetación y estilo de la misma.

3.4. Firebase

Firebase[18] es una plataforma de desarrollo de aplicaciones móviles y web desarrollada por Google que proporciona una variedad de herramientas y servicios en la nube para el desarrollo de aplicaciones. Firebase fue adquirido por Google en 2014 y desde entonces ha evolucionado para ofrecer una amplia gama de servicios, en este proyecto se utilizan:

- Realtime Database[19]: una base de datos no relacional, que se encuentra en la nube y se sincroniza en tiempo real con los clientes de la aplicación
- Authentication[20]: un sistema de autenticación de usuarios que permite a los usuarios iniciar sesión en las aplicaciones utilizando diferentes proveedores de autenticación, como Google, Facebook o correo electrónico.

- Storage[21]: un servicio que permite a los desarrolladores almacenar archivos y datos en la nube.

3.5. Capacitor

Capacitor[22] es un framework de código abierto desarrollado por el equipo detrás de Ionic, que permite a los desarrolladores crear aplicaciones móviles multiplataforma utilizando tecnologías web como HTML, CSS y JavaScript. Capacitor proporciona una capa de abstracción que permite a las aplicaciones web acceder a las características y funcionalidades nativas del dispositivo, como la cámara, el almacenamiento, el GPS y otros sensores.

3.6. Ngx-Scanner

Ngx-Scanner[23] es un módulo de escaneo de códigos de barras y QR para aplicaciones web desarrolladas con Angular, que proporciona una fácil integración y personalización del componente de escaneo. Este módulo utiliza la implementación en JavaScript y es una excelente opción para agregar capacidades de escaneo de códigos de barras y QR a las aplicaciones web.

3.7. Highcharts

Highcharts[24] es una biblioteca de gráficos interactiva y de código abierto para el desarrollo de visualizaciones de datos en aplicaciones web. Está construida utilizando tecnologías web estándar, como HTML, CSS y JavaScript, y es compatible con la mayoría de los navegadores modernos.

3.8. Visual Studio Code

Visual Studio Code[25] (también conocido como VS Code) es un editor de código fuente desarrollado por Microsoft. Es una herramienta de desarrollo de software altamente personalizable y potente que se utiliza para escribir y editar código en varios lenguajes de programación.

Ofrece una amplia gama de características útiles para los desarrolladores, incluyendo resaltado de sintaxis, sugerencias de código, autocompletado de código, depuración de código, integración de control de versiones, soporte para extensiones y mucho más.

3.9. Github

GitHub[7] es una plataforma de alojamiento de código fuente y de gestión de proyectos basada en la web que utiliza el sistema de control de versiones Git. Fue fundada en 2008 y adquirida por Microsoft en 2018.

Ofrece una gran cantidad de herramientas y características, como control de versiones, integración continua, pruebas automatizadas, seguimiento de problemas, revisión de código y muchas más, que ayudan a los desarrolladores a trabajar de manera más eficiente y colaborativa.

3.10. Trello

Trello[26] es una aplicación de gestión de proyectos basada en la web que permite a los usuarios organizar y priorizar sus tareas y proyectos en un formato visual y colaborativo.

En Trello, los usuarios crean tableros para sus proyectos y agregan tarjetas a cada tablero para representar tareas, ideas, problemas, etc. Cada tarjeta puede tener una descripción, lista de comprobación, fecha de vencimiento, comentarios, etiquetas y archivos adjuntos. Las tarjetas se pueden mover entre las listas de los tableros, lo que permite a los usuarios rastrear el progreso de sus tareas y proyectos.

Capítulo 4

Especificación de requisitos

Este capítulo está dedicado a la descripción de los actores y la especificación de requisitos realizada.

4.1. Actores

A continuación, se explica cuáles son los principales tipos de actores que podrán utilizar la aplicación:

- **Usuario registrado:** Es el usuario que visita la tienda para buscar y/o comprar los productos de la misma, teniendo asociado funciones para ello.
- **Administrador:** Es el usuario encargado de gestionar los productos de la tienda, así como acceder a la información obtenida del análisis de la información

4.2. Diagramas de casos de uso

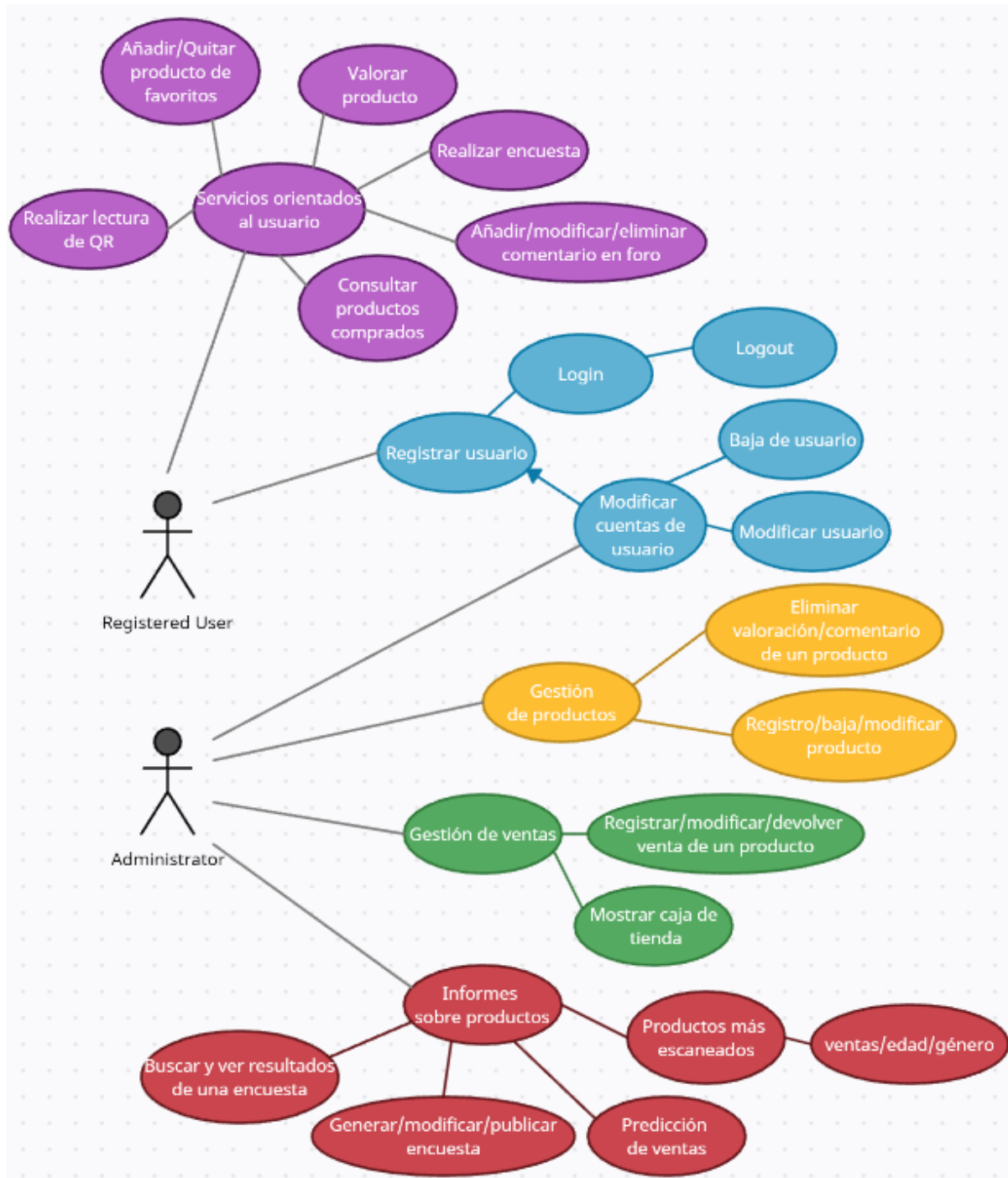


Figura 4.1: Diagrama de casos de uso

4.3. Especificación de requisitos

En esta sección se van a explicar los casos de uso definidos para la aplicación. Para ello los casos de uso se han agrupado modularmente de la siguiente forma:

1. Cuentas de usuario

- Registrar usuario
- Login
- Logout
- Dar de baja usuario
- Modificar datos de usuario

2. Gestión de productos

- Registrar un producto
- Dar de baja un producto
- Modificar un producto
- Eliminar una valoración de un producto o un comentario en el foro.

3. Gestión de ventas

- Registrar ventas de un producto
- Modificar número de ventas de un producto
- Registrar devolución de un producto
- Mostrar caja de tienda

4. Informes sobre productos

- Informe sobre los productos más escaneados + Informe sobre los productos más escaneados con relación a las ventas, edad o género
- Informe sobre predicción de ventas
- Generar/modificar/publicar encuesta para clientes.
- Buscar y ver resultados de una encuesta.

5. Servicios orientados al usuario

- Realizar lectura de QR de un producto

- Añadir/Quitar producto a favoritos
- Consultar productos comprados.
- Realizar encuesta.
- Añadir/Eliminar/modificar comentario en foro.
- Valorar producto

4.3.1 Cuentas de usuario

A continuación, se describen los casos de uso del módulo funcional “Cuentas de usuario”.

Requisito	Registrar usuario	
Identificador	1.1	
Prioridad	Media	
Precondición	NA	
Descripción	Los usuarios pueden registrarse en la aplicación.	
Entrada	Nombre, Apellidos, Género, Edad, CP, Contraseña, Correo Electrónico	
Salida	NA	
Secuencia normal	Paso	Acción
	1	El usuario accede a la aplicación, se le muestra la pantalla principal y hace click en “Registrarse”.
	2	El sistema muestra una pantalla en la que se le pide al usuario toda la información necesaria para registrarse.
	3	El usuario rellena todos los campos y hace click en “Aceptar”.
	4	El sistema valida la información.
5	El sistema muestra de nuevo la pantalla principal.	
Postcondición	Se ha creado la cuenta de usuario.	
Excepciones	Paso	Acción

	3	Se muestra el mensaje correspondiente si la contraseña no cumple con los requisitos.
	3	Se muestra el mensaje correspondiente si el correo electrónico no es correcto.
	3	Se muestra el mensaje correspondiente si el correo electrónico ya está registrado.
	3	El usuario hace click en “Cancelar”
Comentarios	NA	
Actores	Usuario registrado, Admin	

Requisito	Login	
Identificador	1.2	
Prioridad	Alta	
Precondición	El usuario ha de estar registrado.	
Descripción	Los usuarios registrados pueden iniciar sesión.	
Entrada	Correo Electrónico, Contraseña	
Salida	NA	
Secuencia normal	Paso	Acción
	1	El usuario accede a la aplicación, se le muestra la pantalla principal y hace click en “Iniciar sesión”.
	2	El sistema muestra una pantalla en la que se le pide al usuario el correo electrónico y la contraseña.
	3	El usuario rellena los campos y hace click en “Acceder”.
	4	El sistema valida la información.

	5	El sistema accede a la vista de usuario correspondiente a su tipo de cuenta.
Postcondición	El usuario ha iniciado sesión.	
Excepciones	Paso	Acción
	3	El usuario hace click en “Cancelar”
	4	No se ha podido realizar el inicio de sesión.
Comentarios	NA	
Actores	Usuario registrado, Admin	

Requisito	Logout	
Identificador	1.3	
Prioridad	Alta	
Precondición	El usuario tiene que haber iniciado sesión.	
Descripción	Los usuarios pueden salir de la sesión iniciada para iniciar una nueva sesión o navegar sin estar registrados.	
Entrada	Nombre de usuario, Id de usuario	
Salida	NA	
Secuencia normal	Paso	Acción
	1	El usuario accede a la aplicación, se le muestra la pantalla de usuario y hace click en “Logout”.
	2	El sistema cierra la sesión y vuelve a la pantalla principal.
Postcondición	Se ha cerrado la sesión.	
Excepciones	Paso	Acción

	2	No se puede cerrar sesión.
Comentarios	NA	
Actores	Usuario registrado, Admin	

Requisito	Dar de baja usuario	
Identificador	1.4	
Prioridad	Media	
Precondición	El admin tiene que haberse registrado y tener una sesión iniciada.	
Descripción	El admin puede eliminar cuentas de usuario.	
Entrada	ID usuario, Nombre de Usuario	
Salida	NA	
Secuencia normal	Paso	Acción
	1	El usuario accede a la aplicación, se le muestra la pantalla principal y hace click en “Eliminar Usuario”.
	2	El sistema muestra una pantalla en la que se le pide al usuario que introduzca el correo electrónico del usuario que quiere eliminar.
	3	El usuario introduce el correo electrónico del usuario.
	4	El sistema muestra un mensaje preguntando por la confirmación.
	5	El usuario hace click en “Eliminar”.
	6	El sistema muestra la pantalla de usuario.
Postcondición	Se ha eliminado la cuenta de usuario.	
Excepciones	Paso	Acción
	4	No se ha encontrado ningún usuario.

	5	El usuario hace click en “Cancelar”.
Comentarios	NA	
Actores	Admin	

Requisito	Modificar datos de usuario	
Identificador	1.5	
Prioridad	Media	
Precondición	El admin tiene que haberse registrado y tener una sesión iniciada.	
Descripción	El admin puede modificar los datos de otras cuentas de usuario.	
Entrada	Correo Electrónico	
Salida	NA	
Secuencia normal	Paso	Acción
	1	El usuario accede a la aplicación, se le muestra la pantalla de usuario y hace click en “Modificar Usuario”.
	2	El sistema muestra una pantalla en la que se le pide al usuario que introduzca el correo electrónico del usuario que quiere modificar.
	3	El usuario introduce el correo electrónico del usuario.
	4	El sistema muestra todos los datos del usuario.
	5	El usuario modifica los datos y hace click en “Modificar”.
	6	El sistema muestra un mensaje preguntando por la confirmación.
9	El usuario hace click en “Continuar”.	
	10	El sistema muestra los datos del usuario que se ha modificado.

Postcondición	Se han modificado los datos del usuario.	
Excepciones	Paso	Acción
	4	No se ha encontrado ningún usuario.
	9	El usuario hace click en “Cancelar”.
	10	No se han podido modificar los datos del usuario.
Comentarios	NA	
Actores	Admin	

4.3.2 Gestión de productos

A continuación, se describen los casos de uso del módulo funcional “Gestión de productos”.

Requisito	Registrar un producto	
Identificador	2.1	
Prioridad	Alta	
Precondición	Debe haberse iniciado sesión con cuenta de administrador	
Descripción	Dado un usuario con cuenta de administrador, podrá registrar un nuevo producto con información acerca de éste y así generar su código QR	
Entrada	Código EAN, nombre del producto, precio, bullet points, descripción, detalles del producto, imágenes del producto	
Salida	Código QR	
Secuencia normal	Paso	Acción
	1	El administrador accede a la aplicación, se le muestra la pantalla de administrador y selecciona la opción “Registrar un producto”.

	2	El sistema muestra una nueva pantalla con todos los datos necesarios para registrar un nuevo producto.
	3	El usuario introduce todos los datos para registrar un producto y hace click en el botón “Registrar producto”.
	4	El sistema muestra un mensaje confirmando el registro del producto y la nueva etiqueta QR generada para ese producto.
Postcondición	NA	
Excepciones	Paso	Acción
Actores	Usuario con cuenta de administrador	

Requisito	Dar de baja un producto	
Identificador	2.2	
Prioridad	Alta	
Precondición	Debe haberse iniciado sesión con cuenta de administrador	
Descripción	Dado un usuario con cuenta de administrador, podrá dar de baja un producto que se haya registrado previamente en la aplicación	
Entrada	Código EAN	
Salida	Mensaje de eliminación satisfactoria	
Secuencia normal	Paso	Acción
	1	El administrador accede a la aplicación, se le muestra la pantalla de administrador y selecciona la opción “Dar de baja un producto”
	2	El sistema muestra una nueva pantalla con la petición del código EAN del producto
	3	El usuario escribe el código EAN del producto y presiona el botón “Dar de baja”
	4	El sistema muestra los resultados satisfactorios de la baja del producto

Postcondición	NA	
Excepciones	Paso	Acción
Actores	Usuario con cuenta de administrador	

Requisito	Modificar un producto	
Identificador	2.3	
Prioridad	Media	
Precondición	Debe haberse iniciado sesión con cuenta de administrador	
Descripción	Dado un usuario con cuenta de administrador, podrá modificar los datos de un producto que se haya registrado previamente en la aplicación	
Entrada	Código EAN	
Salida	Mensaje de eliminación satisfactoria	
Secuencia normal	Paso	Acción
	1	El administrador accede a la aplicación, se le muestra la pantalla de administrador y selecciona la opción “Modificar un producto”
	2	El sistema muestra una nueva pantalla con la petición del código EAN del producto
	3	El usuario escribe el código EAN del producto y presiona el botón “Modificar detalles del producto”
	4	El sistema muestra una nueva pantalla con los datos: nombre del producto, precio, bullet points, descripción, detalles del producto, imágenes del producto para modificarlos
5	El administrador introduce los datos a cambiar del producto y hace click en “Modificar”	
Postcondición	NA	
Excepciones	Paso	Acción
Actores	Usuario con cuenta de administrador	

Requisito	Eliminar valoración de un producto o un comentario en el foro	
Identificador	2.4	
Prioridad	Media	
Precondición	Debe haberse iniciado sesión con cuenta de administrador	
Descripción	Dado un usuario con cuenta de administrador, podrá eliminar comentarios en el foro o eliminar valoraciones de un producto	
Entrada	Código EAN	
Salida	Mensaje de eliminación satisfactoria	
Secuencia normal	Paso	Acción
	1	El administrador accede a la aplicación, se le muestra la pantalla de administrador y accede al foro o a las valoraciones de un producto
	2	El administrador tiene la posibilidad tanto de eliminar valoraciones de un producto en la página de valoraciones de un producto como de eliminar comentarios en el foro
Postcondición	NA	
Excepciones	Paso	Acción
Actores	Usuario con cuenta de administrador	

4.3.3 Gestión de ventas

A continuación, se describen los casos de uso del módulo funcional “Gestión de ventas”.

Requisito	Registrar ventas de un producto	
Identificador	3.1	
Prioridad	Alta	

Precondición	El usuario debe haber iniciado sesión con cuenta de administrador	
Descripción	Dado un usuario con cuenta de administrador, podrá registrar las ventas realizadas de uno o más productos registrados en la aplicación	
Entrada	Código EAN del producto, n° de unidades vendidas, mes y año de la venta	
Salida	Pantalla con n° de ventas totales	
Secuencia normal	Paso	Acción
	1	El usuario inicia sesión con su cuenta de administrador, accede a la pantalla de administrador y selecciona la opción “Registrar ventas de uno o más productos”
	2	El sistema muestra una pantalla en la que se le pide al usuario introducir el número de productos de los que registrar sus ventas.
	3	El sistema muestra una pantalla con una línea por cada producto, donde se le pide al usuario el código EAN del producto, el n° de ventas realizadas de ese producto y la fecha en la que se vendieron
	4	El usuario hace click en el botón “Registrar ventas”
5	El sistema muestra una pantalla que con el n° de ventas totales realizadas en los últimos meses	
Postcondición	NA	
Excepciones	Paso	Acción
	4	Se muestra un mensaje de error en caso de que se haya introducido un nombre que no existe o un valor negativo.
Comentarios	Cuando se compra un producto, este se almacena en un registro de productos comprados por el comprador, de forma que el usuario pueda ver qué productos ha comprado en la tienda	
Actores	Usuario registrado	

Requisito	Modificar número de ventas de un producto	
Identificador	3.2	
Prioridad	Media	
Precondición	El usuario debe haber iniciado sesión con cuenta de administrador	
Descripción	Dado un usuario con cuenta de administrador, podrá modificar el número de ventas de un producto en un mes determinado.	
Entrada	Código EAN del producto, Nueva cantidad de unidades vendidas, mes y año de la modificación	
Salida	Cartera de activos financieros y rendimiento	
Secuencia normal	Paso	Acción
	1	El usuario inicia sesión con su cuenta de administrador, accede a la pantalla de administrador y selecciona la opción “Modificar ventas de un producto”
	2	El sistema muestra una pantalla en la que se le pide al usuario introducir el código EAN del producto del que modificar sus ventas.
	3	Se le muestra al usuario una pantalla con los campos a modificar: Código EAN del producto, Nueva cantidad de unidades vendidas, mes y año de la modificación
	4	El usuario hace click en el botón “Modificar ventas”
	5	El sistema muestra una pantalla que con el nº de ventas totales realizadas en los últimos meses
Postcondición	NA	
Excepciones	Paso	Acción
Comentarios	NA	

Actores	Usuario con cuenta de administrador
---------	-------------------------------------

Requisito	Registrar devolución de un producto	
Identificador	3.3	
Prioridad	Alta	
Precondición	El usuario debe haber iniciado sesión con cuenta de administrador	
Descripción	Dado un usuario con cuenta de administrador, podrá registrar la devolución de un producto.	
Entrada	Código EAN del producto, n° de unidades devueltas, mes y año de devolución	
Salida	NA	
	Paso	Acción
Secuencia normal	1	El usuario inicia sesión con su cuenta de administrador, accede a la pantalla de administrador y selecciona la opción “Registrar devolución de un producto”
	2	El sistema muestra una pantalla en la que se le pide al usuario introducir el código EAN, la cantidad de unidades devueltas y el mes y año de la devolución
	3	El usuario hace click en registrar devolución
Postcondición		
Excepciones	Paso	Acción
	2	El código EAN del producto no existe
Comentarios	NA	
Actores	Usuario con cuenta de administrador	

Requisito	Mostrar caja de tienda	
Identificador	3.4	
Prioridad	Media	
Precondición	El usuario debe haber iniciado sesión con cuenta de administrador	
Descripción	Dado un usuario con cuenta de administrador, podrá ver las ventas que se hayan hecho en su tienda	
Entrada		
Salida	Gráfico con caja de tienda	
Secuencia normal	Paso	Acción
	1	El usuario administrador con sesión iniciada, accede a la pantalla de administrador y hace click en “Mostrar caja de tienda”
	2	Se le muestra al usuario un gráfico con las ventas realizadas en relación a los meses
Postcondición		
Excepciones	Paso	Acción
Comentarios	NA	
Actores	Usuario con cuenta de administrador	

4.3.4 Informes de productos

A continuación se describen los casos de uso del módulo funcional “Informes de productos”.

Requisito	Informe sobre los productos más escaneados con relación a las ventas	
Identificador	4.1	
Prioridad	Media	
Precondición	Haber iniciado sesión como usuario administrador	
Descripción	El usuario administrador puede acceder a un informe sobre la cantidad de visitas que tienen los productos por los usuarios y con relación a diferentes factores como las ventas, la edad o el género	
Entrada		
Salida	NA	
Secuencia normal	Paso	Acción
	1	El usuario accede a la aplicación, se le muestra la pantalla principal y selecciona la opción “Diario de Trading”
	2	El usuario administrador inicia sesión a la aplicación, se le muestra la pantalla de administrador y selecciona la opción “productos más populares con relación”
	3	El sistema muestra una pantalla con una gráfica sobre la popularidad de los productos con relación a diferentes factores
Postcondición		
Excepciones	Paso	Acción
Comentarios	NA	
Actores	Usuario con cuenta de administrador	

Requisito	Generar/modificar/publicar encuesta para clientes	
Identificador	4.3	
Prioridad	Alta	
Precondición	Haber iniciado sesión como usuario administrador	
Descripción	Dado un usuario administrador, podrá generar una encuesta modificable y publicarla para que los rellenen los clientes	
Entrada		
Salida	NA	
Secuencia normal	Paso	Acción
	1	El usuario administrador inicia sesión en la aplicación, se le muestra la pantalla de administrador y selecciona la opción “Generar encuesta”
	2	El sistema muestra una pantalla donde se le pide al usuario los campos a añadir en la encuesta
	3	El usuario una vez ha rellenado los datos hace click en “publicar encuesta”
	4	El sistema dirige al usuario a una pantalla donde se le avisa que la encuesta se ha publicado correctamente
Postcondición		
Excepciones	Paso	Acción
Comentarios		
Actores	Usuario administrador	

Requisito	Buscar y ver resultados de la encuesta	
Identificador	4.4	
Prioridad	Alta	
Precondición	Haber iniciado sesión como usuario administrador	

Descripción	Dado un usuario administrador, podrá ver los resultados de la encuesta publicada para los usuarios	
Entrada		
Salida	NA	
Secuencia normal	Paso	Acción
	1	El usuario administrador inicia sesión en la aplicación, se le muestra la pantalla de administrador y selecciona la opción “ver resultados de encuesta”
	2	El sistema muestra una pantalla donde se le muestra al usuario las diferentes encuestas con su nombre, fecha, número de respuestas y código identificador y un campo de texto pidiendo el código identificador de la encuesta
	3	El sistema le muestra al usuario los resultados de la encuesta escogida
Postcondición		
Excepciones	Paso	Acción
	3	El sistema no ha encontrado el id de encuesta
Comentarios	Esta predicción va a tener en cuenta la popularidad de los productos, el número de ventas, el número de devoluciones, el historial de ese producto y otros factores externos como la época del año o la situación económica	
Actores	Usuario administrador	

4.3.5 Servicios orientados al usuario

A continuación, se describen los casos de uso del módulo funcional “Servicios orientados al usuario”.

Requisito	Realizar lectura de QR de un producto	
Identificador	5.1	
Prioridad	Alta	
Precondición	Tiene que existir al menos un producto registrado en la aplicación con ese código QR	
Descripción	Dado un usuario registrado o no registrado de la aplicación, podrá hacer la lectura de QR de un producto y así poder acceder a información acerca de éste	
Entrada		
Salida	Pantalla con información detallada sobre ese producto	
Secuencia normal	Paso	Acción
	1	El usuario accede a la aplicación, se le muestra la pantalla principal y selecciona la opción “Escanear producto”
	2	El sistema muestra una pantalla de detección de QR con la cámara y un encuadre para enfocar el código
	3	El sistema identifica el código QR que ha escaneado el usuario
	4	El sistema muestra una nueva pantalla en la que se le muestra al usuario toda la información que hay guardada acerca de ese producto
Postcondición		
Excepciones	Paso	Acción

	3	Se muestra un mensaje en la pantalla de detección de QR indicando que el código no pertenece a un producto registrado en la aplicación
Actores	Usuario registrado	

Requisito	Añadir/Quitar producto de favoritos	
Identificador	5.2	
Prioridad	Alta	
Precondición	Tiene que existir al menos un producto registrado en la aplicación con ese código QR	
Descripción	Dado un usuario registrado o no registrado de la aplicación, podrá hacer la lectura de QR de un producto y así poder acceder a información acerca de éste	
Entrada		
Salida	Pantalla con la lista de favoritos	
Secuencia normal	Paso	Acción
	1	El usuario accede a la aplicación, se le muestra la pantalla principal y selecciona la opción “Añadir/quitar producto”
	2	El sistema muestra una pantalla donde se le pide al usuario si quiere añadir o eliminar un producto
	3	El sistema muestra una pantalla donde se le pide al usuario el código del producto que quiere añadir o quitar de favoritos
	4	El sistema identifica el código del producto y lo añade o elimina de la lista
Postcondición		
Excepciones	Paso	Acción

	4	El sistema no encuentra el código del producto y notifica al usuario
Actores	Usuario registrado	

Requisito	Consultar productos comprados	
Identificador	5.3	
Prioridad	Alta	
Precondición		
Descripción	Dado un usuario registrado, podrá consultar los productos que ha comprado a través de la aplicación	
Entrada		
Salida	Pantalla con la lista de productos comprados	
Secuencia normal	Paso	Acción
	1	El usuario accede a la aplicación, se le muestra la pantalla principal y selecciona la opción “Consultar productos comprados”
	2	El sistema muestra una lista con los productos comprados por el usuario a través de la aplicación
Postcondición		
Excepciones	Paso	Acción
	2	Si no existen productos comprados por el usuario, no se mostrará una lista y se le notificará al usuario
Actores	Usuario registrado	

Requisito	Realizar encuesta	
Identificador	5.4	
Prioridad	Alta	
Precondición	Tiene que existir una encuesta disponible para el usuario	
Descripción	Dado un usuario registrado de la aplicación, podrá realizar una encuesta que haya sido publicada por el administrador	
Entrada		
Salida	Pantalla con un mensaje de notificación al usuario de encuesta realizada satisfactoriamente	
Secuencia normal	Paso	Acción
	1	El usuario accede a la aplicación, se le muestra la pantalla principal y selecciona la opción “Realizar encuesta”
	2	El sistema muestra una pantalla con una lista de todas las encuestas que han sido publicadas para los usuarios y un campo para introducir el código de la encuesta que quiere realizar
	3	El usuario introduce el código de la encuesta y accede a una pantalla donde puede rellenar la encuesta y enviar los resultados
Postcondición		
Excepciones	Paso	Acción
	2	No existen encuestas
Actores	Usuario registrado	

Requisito	Añadir/eliminar/modificar comentario en foro
------------------	---

Identificador	5.5	
Prioridad	Alta	
Precondición		
Descripción	Dado un usuario registrado en la aplicación, podrá añadir, eliminar o modificar comentarios en foro	
Entrada	Mensaje de texto	
Salida	Pantalla con el foro	
Secuencia normal	Paso	Acción
	1	El usuario accede a la aplicación, se le muestra la pantalla principal y selecciona la opción “Foro”
	2	El sistema muestra una pantalla donde aparece el foro y hará click en añadir/eliminar/modificar comentario
	3	El sistema le dejará al usuario introducir el texto del comentario que quiera realizar en el foro
Postcondición		
Excepciones	Paso	Acción
Actores	Usuario registrado	

Requisito	Valorar producto
Identificador	5.6
Prioridad	Alta

Precondición	Tiene que existir al menos un producto registrado en la aplicación	
Descripción	Dado un usuario registrado de la aplicación, podrá valorar productos de la aplicación	
Entrada	Valoración 1-5, Título, Comentario	
Salida	Pantalla de valoraciones de ese producto	
Secuencia normal	Paso	Acción
	1	El usuario accede a la aplicación, se le muestra la pantalla principal y selecciona la opción “Valorar producto”
	2	El sistema muestra una pantalla con un campo de texto donde el usuario podrá introducir el código del producto del que quiere hacer una valoración y hará click en valorar
	3	El sistema muestra al usuario una pantalla donde se le pide la valoración de 1-5, el título y el comentario y un botón “Enviar”
Postcondición		
Excepciones	Paso	Acción
Actores	Usuario registrado	

Capítulo 5

Arquitectura de la aplicación

En este capítulo se describe la estructura de la aplicación y su modelo de datos.

5.1. Estructura de la aplicación

La aplicación hace uso de una arquitectura sin servidor.

La arquitectura sin servidor es un enfoque del diseño de software que permite a los desarrolladores crear y ejecutar servicios sin tener que administrar la infraestructura subyacente.

En lugar de crear y ejecutar una aplicación en una máquina virtual o en un servidor dedicado, la arquitectura sin servidor permite que el desarrollador cree pequeñas funciones que se ejecutan en respuesta a eventos, como solicitudes HTTP o cambios en una base de datos, permitiendo que el desarrollador se centre en la lógica de la aplicación.

El encargado de realizar la autenticación de las cuentas de usuario, base de datos y almacenamiento en este caso es Firebase y el encargado de crear un servidor local para hacer las pruebas de la aplicación es Angular, que a su vez a través de capacitor crea la aplicación móvil que podrá compartirse con los clientes o en este caso con las personas de la Universidad Complutense que deseen hacer uso de la aplicación para poderla evaluar.

Las funcionalidades específicas de la cuenta de cliente se han diferenciado de la cuenta de administrador. El cliente podrá escanear los códigos QR de los productos registrados en la aplicación a través de ngx-scanner para conocer más acerca de ellos, lo cual generará una señal en la base de datos de Firebase para que el administrador sea capaz de generar gráficos con los datos que analizar posibles tendencias con el módulo Highcharts, como se muestra en la figura [5.1](#).

Lo que muestra la siguiente imagen es la representación de la conexión que existe entre la cuenta de cliente con el módulo de ngx-qr-scanner para escanear productos y la cuenta de administrador con el módulo Highcharts para visualizar datos a través de gráficos. Conectando ambos se encuentra Firebase, ofreciendo a ambos tipos de cuenta los servicios de Realtime Database, Authentication y Cloud Storage.

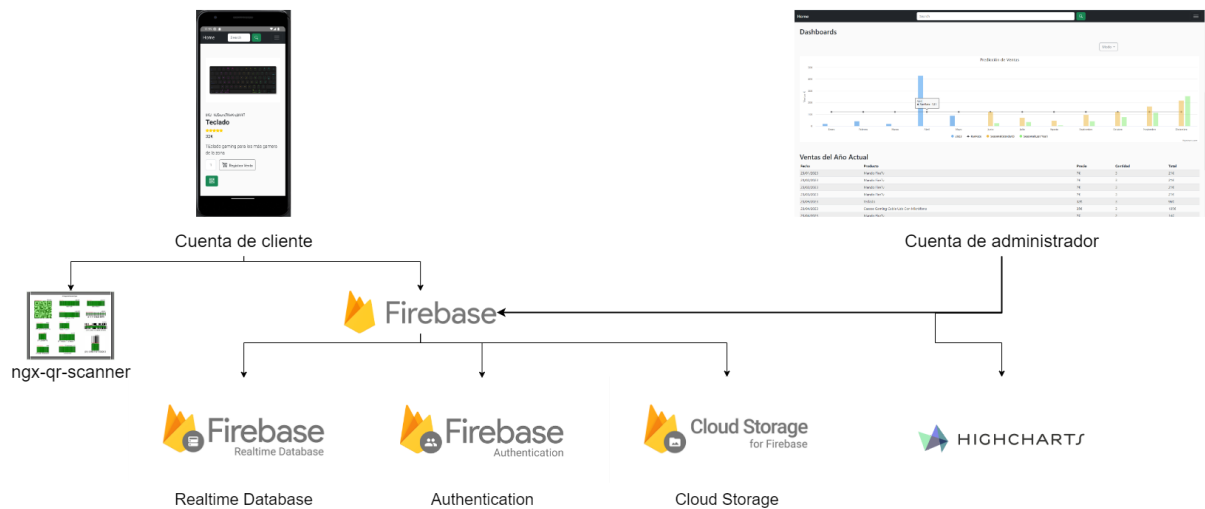


Figura 5.1: Arquitectura de la aplicación

5.2. Modelo de datos

La información almacenada en la base de datos necesita de una forma de almacenamiento persistente, ya que la información de los usuarios es actualizada constantemente y ha de ser actualizada en tiempo real. Necesitaremos almacenar las visitas de los usuarios a los productos, registrar nuevos usuarios, registrar ventas de los usuarios y realizar gráficos en tiempo real con toda esta información.

Para el almacenamiento de estos datos se ha hecho uso de Firebase Realtime Database que es una base de datos en tiempo real, lo que significa que los cambios en los datos se propagan instantáneamente a todos los clientes que están conectados a la base de datos. Esto lo hace ideal para aplicaciones en tiempo real, como ocurre en esta aplicación.

Se divide en distintas entidades relacionadas con las diferentes funcionalidades que ofrece la aplicación: “usuarios”, que son los usuarios que se han registrado en la aplicación, donde se almacenan datos personales, productos favoritos y encuestas contestadas como puede observarse en la [figura 5.2](#). “productos”, que son los distintos productos que el vendedor tendría a la venta en su tienda física, con su título, precio, valoración... “ventas”, son las ventas realizadas de cada producto, se almacena la cantidad de unidades vendidas, el email del comprador, el precio de venta y la fecha en que se realizó la venta. Y la entidad “encuestas”, que almacena toda la información relacionada con las encuestas publicadas o guardadas.

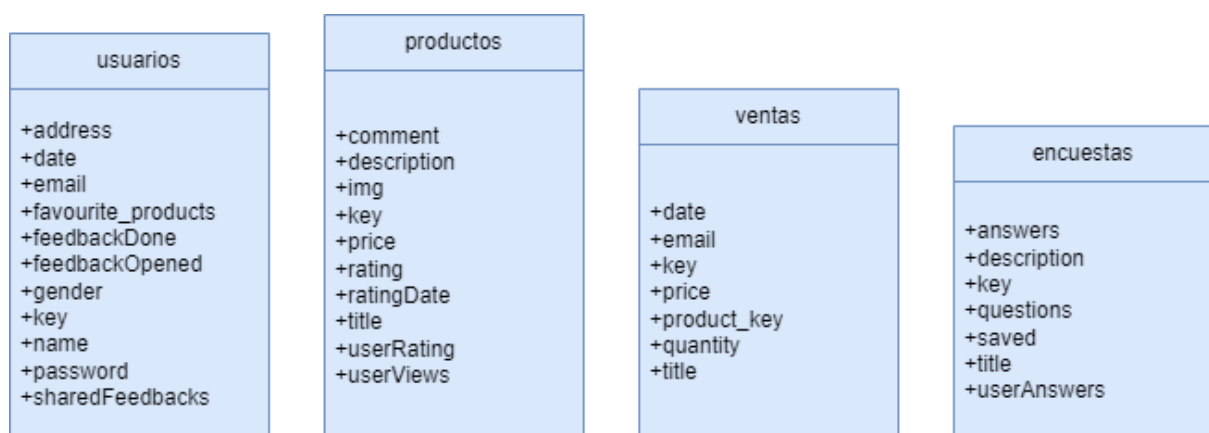


Figura 5.2: Modelo de datos

En la base de datos de Firebase se almacenan los datos en formato JSON[27], representados de esta manera en las tablas de la figura 5.2. Las colecciones de datos utilizadas son las siguientes:

5.2.1. Colección Usuarios

Esta colección se utiliza para almacenar la información personal de los usuarios y datos que guardan los usuarios al interactuar con la aplicación. En ella se registra una clave única, la dirección, la fecha de nacimiento, el email, una lista con los ids de los productos favoritos del usuario, una lista con las encuestas realizadas, una lista con las encuestas que han sido abiertas, el género del usuario, el nombre, la contraseña y la lista de encuestas que han sido compartidas con este usuario como se puede comprobar en la imagen de la figura 5.3.

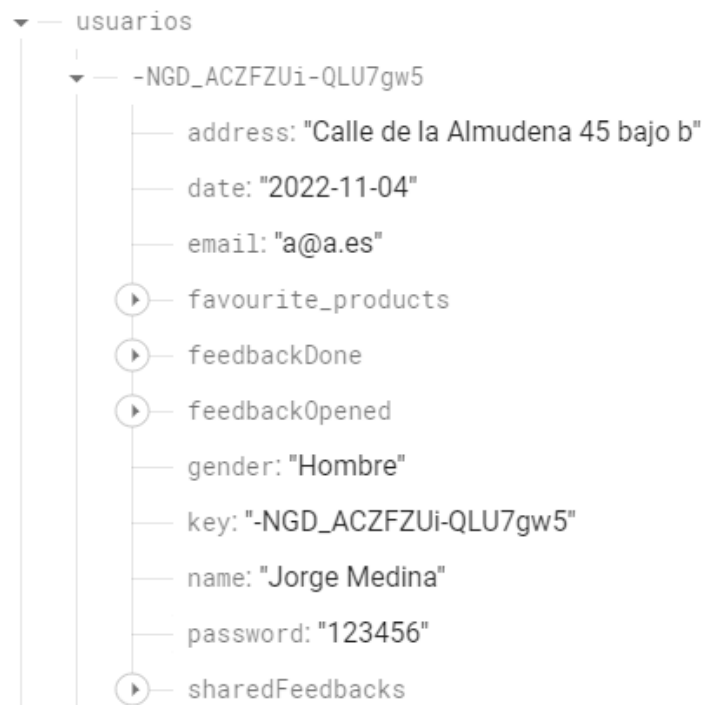


Figura 5.3: Colección Usuarios

5.2.2. Colección Productos

La colección productos se utiliza para guardar información sobre los productos que aparecen en el catálogo de la aplicación, para que el usuario pueda acceder a ellos y buscar sus detalles y el administrador pueda llevar seguimiento de qué productos están siendo más populares. En esta colección se guardan una lista con los comentarios de la valoración del producto, una descripción, un link enlazado a la imagen que se ha registrado en Firebase Storage, una clave única, el precio del producto, una lista con la valoración (1-5), el título del producto, una lista con los usuarios que han realizado una valoración del producto y una lista con los usuarios que han visitado el producto. Se puede ver en la [figura 5.4](#).

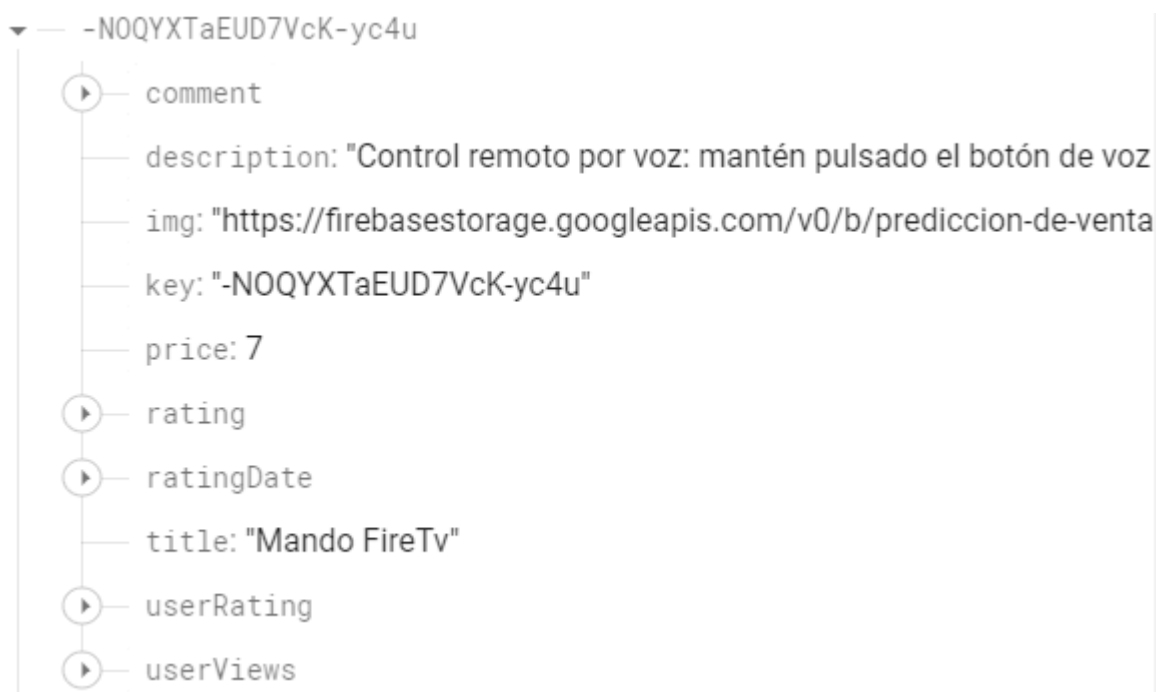


Figura 5.4: Colección Productos

5.2.3. Colección Ventas

Esta colección se utiliza para que el usuario administrador registre las ventas que han realizado los usuarios en la tienda y guarde la información de la venta que se ha realizado. Se registran, una clave única, la fecha de la venta, el email de la persona que realizó la venta, el precio del producto vendido, la clave del producto vendido, la cantidad de productos que se han vendido y el título del producto vendido como se puede observar en la [figura 5.5](#).

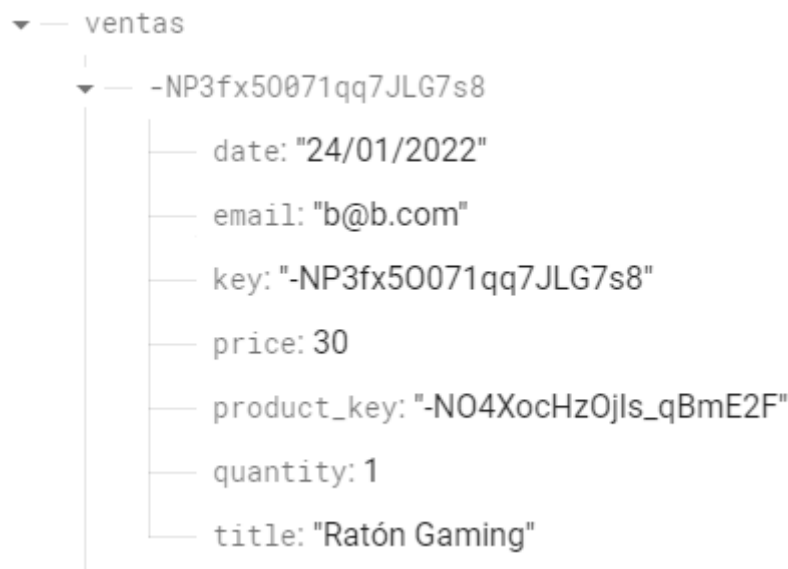


Figura 5.5: Colección Ventas

5.2.4. Colección Encuestas

La colección encuestas registra datos sobre las encuestas que el usuario administrador decide publicar o guardar y sus respuestas. Se registra una clave única, una lista bidimensional que guarda para cada nombre de usuario cada una de las respuestas dadas en la encuesta, la descripción de la encuesta, una lista de pares con el tipo de pregunta y título de la pregunta, un campo guardado para saber si la encuesta aparece en la lista de plantillas de encuesta guardadas por el usuario, el título de la encuesta y la cantidad de respuestas que se han realizado. Se puede ver la colección encuestas en la [figura 5.6](#).

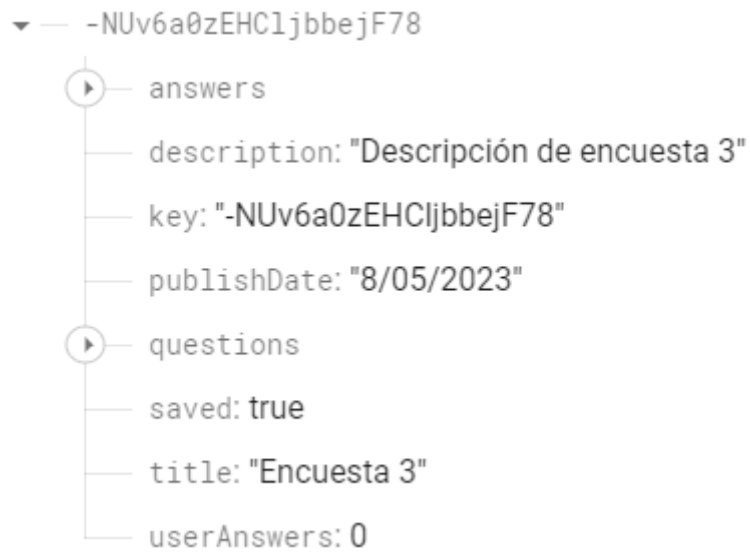


Figura 5.6: Colección Encuestas

Capítulo 6

Implementación y diseño de la aplicación

A continuación se exponen los detalles de la implementación y el diseño de la aplicación, agrupando las funcionalidades según el módulo al que pertenecen:

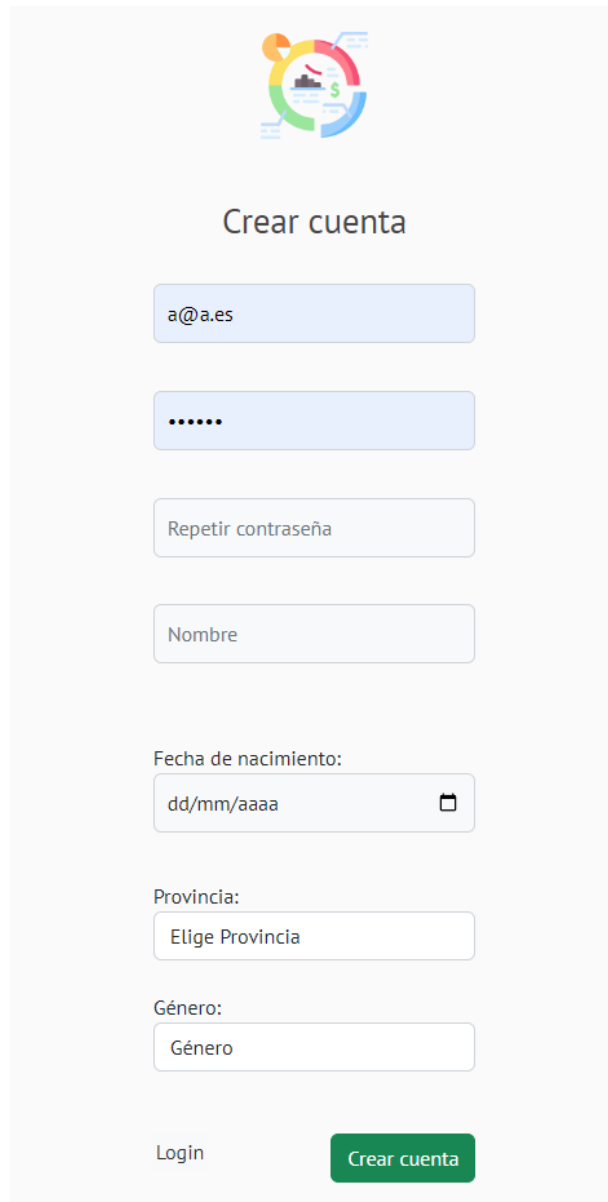
- Módulo Cuentas de Usuario
- Módulo Gestión de Productos
- Módulo Gestión de Ventas
- Módulo Informes de Productos
- Servicios Orientados al Usuario

6.1. Módulo Cuentas de Usuario

6.1.1. Registrar usuario

Cuando el usuario hace clic en “Crear cuenta” en la pantalla de login, accede al formulario de registro de sesión para dar de alta a los usuarios. Los campos a introducir son e-mail, contraseña, repetir contraseña, nombre, fecha de nacimiento, provincia y género como se observa en la [figura 6.1](#). Una vez estos campos han sido rellenados con los datos del usuario que quiere registrarse, hará clic en el botón “Crear cuenta” que ahora aparecerá en verde debajo del formulario. Una vez hecho esto, si las contraseñas coinciden y el resto de datos han sido rellenados, entonces se creará la cuenta del nuevo usuario.

Se han empleado los campos de texto de la clase input-form de bootstrap, junto con campos de clase select-form y date-form, dando una apariencia moderna, sencilla y fácil de utilizar para el usuario. Finalmente se han empleado los colores rojo amarillo, azul y verde para el logo del inicio de sesión, el color rojo al clicar el botón de “Login” el cual nos dirige a la pantalla de login y el color verde para el botón de “Crear cuenta”, con el que enviamos el formulario para crear una nueva cuenta.



The image shows a user registration form with the following fields and elements:

- Logo at the top center: a circular icon with a dollar sign, a bar chart, and a gear.
- Title: "Crear cuenta"
- Email field: "a@a.es"
- Password field: "*****"
- Repeat password field: "Repetir contraseña"
- Name field: "Nombre"
- Birth date field: "Fecha de nacimiento:" with a sub-field "dd/mm/aaaa" and a calendar icon.
- Province field: "Provincia:" with a sub-field "Elige Provincia".
- Gender field: "Género:" with a sub-field "Género".
- Buttons at the bottom: "Login" and "Crear cuenta" (highlighted in green).

Figura 6.1: Registrar usuario

Se da de alta al usuario tanto en registerService, figura [6.2](#), (módulo Authentication de firebase) como en la tabla de usuarios de “Realtime Database” como se puede ver en la figura [6.3](#). El motivo es que el módulo authentication ofrece servicios de autenticación de usuarios, registro, login y logout de usuarios, gestión de usuarios y verificación de credenciales de usuarios, a parte de ofrecer una forma segura de mantener la información registrada en la base de datos.

```

newRegister(registerForm: NgForm) {
  var formValue = registerForm.value
  this.user = User.emptyUser()
  this.user.email, this.user.name, this.user.date, this.user.address, this.user.gender,
  this.user.password = formValue["email"], formValue["name"], formValue["date"],
  formValue["address"], formValue["gender"], formValue["password"]

  if(this.validatePassword(formValue["password"], formValue["passwordr"])){
    this.registerService.register(formValue).then(response => {
      console.log("Correct register: " + formValue["email"]);
      this.dataService.saveUser(this.user);
      registerForm.reset();
    })
    .catch(error=>alert(error));
  }
}

```

Figura 6.2: newRegister()

```

saveUser(user:User){
  this.httpClient.post('https://prediccion-de-ventas-default-rtdb.europe-west1.' +
  'firebase.database.app/usuarios.json', user).subscribe(
    response=> {
      console.log("Correct register: " + user.email)
      // @ts-ignore
      user.key = response['name']
      this.updateUser(user)
    },
    error=>console.error("Error saving user: " + user.email + ' -> ' + error)
  );
}

```

Figura 6.3: dataService.saveUser()

6.1.2. Login

Cuando el usuario accede a la aplicación por primera vez, la primera pantalla que aparece es la pantalla de login como puede verse en la figura [6.4](#). En esta pantalla hay dos campos de texto, E-mail y contraseña. El usuario que quiere hacer login debe de introducir estos campos para poder hacer login en la aplicación y acceder al contenido de ésta haciendo clic en el botón verde “Login”. Una vez ha clicado, si los datos introducidos pasan la verificación de

Firebase Authenticator, es decir, la cuenta existe y ha sido registrada correctamente en la base de datos, entonces el usuario podrá acceder a la aplicación.

Esta pantalla utiliza el mismo diseño que la pantalla de registro, empleando el mismo logo y usando el mismo tipo de botones y clases de implementación “input-form” para los campos de texto y botones que ofrece bootstrap. En este caso el botón “Crear cuenta” nos dirige a la pantalla de registro, mientras que el botón “Login” si el usuario se ha verificado correctamente, nos llevará a la pantalla home de la aplicación.

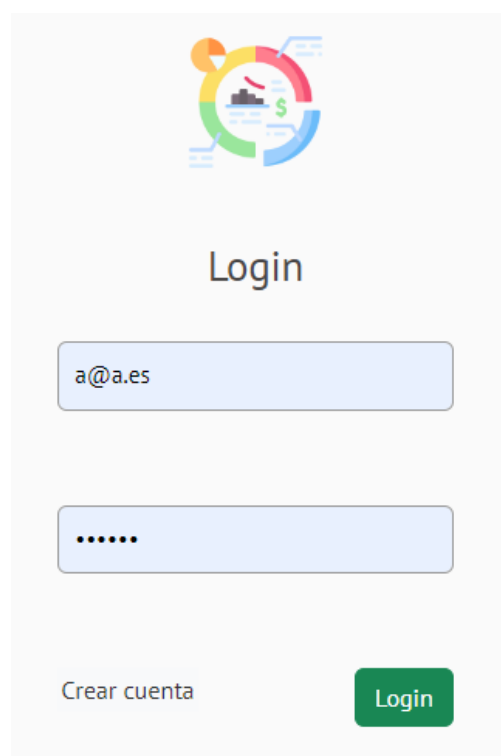


Figura 6.4: Login

```

login(loginForm:NgForm){
  var formValue = loginForm.value;
  this.loginService.signIn(new Register(formValue.email, formValue.password)).then(response=>{
    this.dataService.loadUsers().subscribe(user=>{
      let users = Object.values(user)
      users.forEach(user => {
        if(user.email == this.cookie.get("email")){
          this.fillCookieWithUserData(user)
          console.log(this.cookie.get("favouriteProducts"))
        }
      })
    })
  }).catch(error => console.log(error));
}

```

Figura 6.5: login()

6.1.3. Logout

Cuando el usuario hace clic en el botón de logout, entonces la aplicación llevará al usuario a la pantalla de login, perdiendo así la autenticación de haber hecho login con una cuenta y por lo tanto el acceso al contenido de la aplicación. Antes debe acceder a la barra de navegación extensible que se muestra en la figura [6.6](#). El botón de logout se encuentra en la esquina superior izquierda de la barra de navegación desplegable que puede verse en la figura [6.7](#).

En este caso los colores empleados mezclan el blanco para el texto, el botón extensible que abre la barra de navegación y el botón de logout de la barra de navegación cuyo icono se ha obtenido de bootstrap. Y el color negro “black-navbar” de bootstrap para distinguir tanto el header como la barra de navegación extensible.

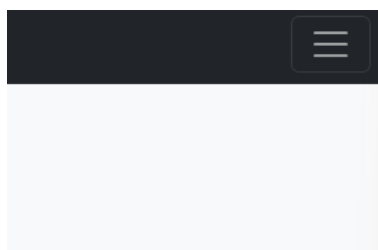


Figura 6.6: Botón desplegable

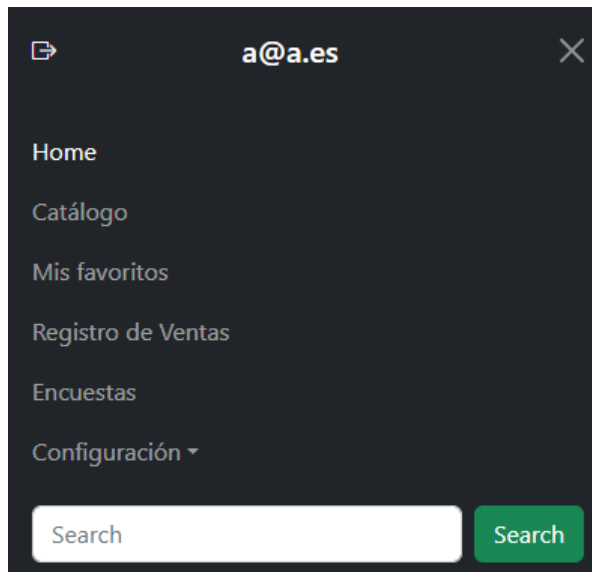


Figura 6.7: Barra de navegación

```
public logout(){
    this.loginService.logout();
}
```

Figura 6.8 : logout()

6.1.4. Dar de baja usuario

Como usuario administrador, accediendo a “Configuración” -> “Administrar perfiles”, podemos ver todos los usuarios que han sido registrados en la aplicación como se puede ver en la figura [6.9](#) en forma de tabla obtenida de bootstrap, donde tenemos dos botones en la columna “Acciones” de la tabla. Clicando en el icono azul de la basura, podemos dar de baja (lógica) usuarios de la aplicación, de forma que no volverán a salir en la tabla y estos usuarios no podrán hacer login en la aplicación con las cuentas dadas de baja.

Se ha empleado el color verde de la clase table-success de bootstrap para distinguir el header de la tabla y el color blanco de la clase table-light para mostrar el contenido de la tabla.

Administrar Perfiles

E-mail	Fecha	Acciones
a@a.es	2022-11-04	 
b@b.com	2022-11-21	 
c@c.com	2022-11-10	 

Figura 6.9: Administrar perfiles

Por otra parte, todos los usuarios podrán eliminar su propio perfil. Podrán acceder a través de la barra de navegación en “Configuración” -> “Modificar Perfil”, ejemplo mostrado en la figura [6.10](#). En esta página los usuarios tienen el botón “Delete User” el cual borrará la propia cuenta. En este caso se ha empleado el color azul oscuro para diferenciar el botón de modificar perfil con diseño de bootstrap button.

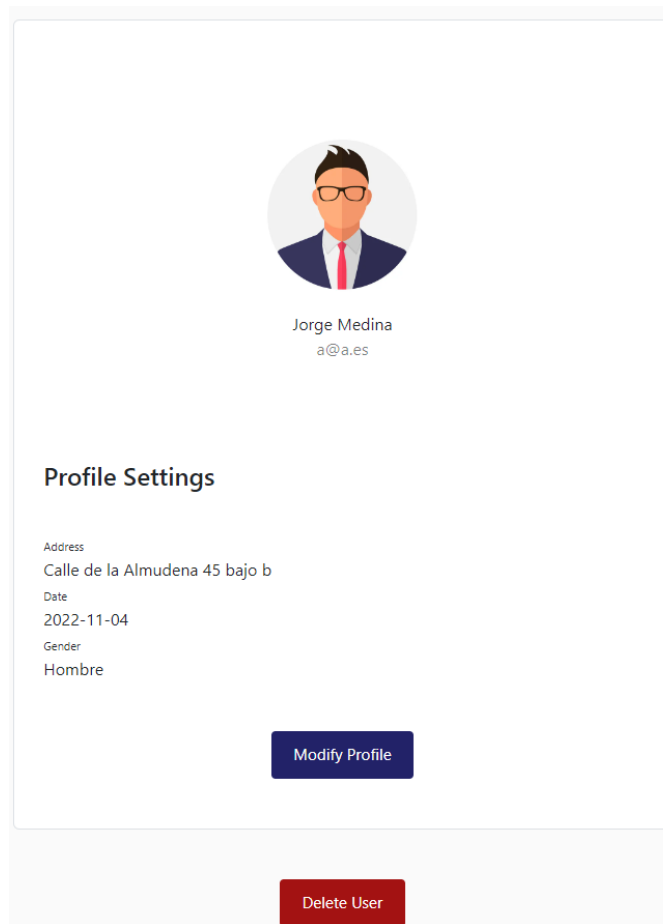


Figura 6.10: Modificar perfil

Debemos dar de baja al usuario en varios servicios, debemos tener en cuenta que usamos tanto el loginService que ofrece Firebase (Authentication) mostrado en la figura [6.11](#), como la baja del usuario en la tabla de usuarios de “Realtime Database” que se muestra en la figura [6.11](#).

```
deleteUser(){  
  this.loginService.delUser(this.user, false)  
  this.dataService.deleteUser(this.user)  
}
```

Figura 6.11: deleteUser()

```

deleteUser(user:User){
  this.httpClient.delete('https://prediccion-de-ventas-default-rtdb.europe-west1.' +
    'firebase.database.app/usuarios/' + user.key + '/.json').subscribe(
    response=>console.log("Correct deletion of user " + user.email),
    error=>console.error("Error on user deletion: " + user.email + ' -> ' + error)
  );
}

```

Figura 6.11: `dataService.deleteUser()`

6.1.5. Modificar datos de usuario

Todos los usuarios podrán modificar su propio perfil a través de la barra de navegación en “Configuración” -> “[Modificar Perfil](#)”. En esta página los usuarios tienen el botón “Modify Profile” el cual servirá para modificar los datos de la propia cuenta.

A parte el usuario administrador podrá modificar cuentas de usuario accediendo a “Configuración” -> “[Administrar Perfiles](#)” pulsando en el botón de configuración asignado a cada usuario.

Se realizará una modificación a través de un método post directamente sobre la base de datos de firebase como se muestra en el código que aparece en la figura [6.13](#).

```

updateUser(){
  this.dataService.updateUser(this.user)
  this.editing = false
}

```

Figura 6.12: `updateUser()`

```

updateUser(user:User){
  if(user.key) {
    this.httpClient.put('https://prediccion-de-ventas-default-rtdb.europe-west1.' +
      'firebase.database.app/usuarios/' + user.key + '/.json', user).subscribe(
      response=>console.log("Correct update: " + user.email),
      error=>console.error("Error updating user: " + user.email + ' -> ' + error)
    );
  }else {
    alert("Trying to update with empty key for user: " + user)
  }
}

```

Figura 6.13: *dataService.updateUser()*

6.2. Módulo Gestión de Productos

6.2.1. Registrar un producto

Con la cuenta de administrador, accediendo a través de la barra de navegación a “Catálogo”, accedemos al catálogo con todos los productos de la aplicación como se ve en la figura [6.14](#), en la parte superior de esta vista se encuentra un botón de la clase bootstrap button, de color light con forma de “+”, icono que se ha obtenido de la colección de bootstrap.

Clicando este botón aparecerá un modal con varios campos de texto para introducir los detalles del producto como aparece en la figura [6.15](#), se pueden registrar nuevos productos que se añadirán automáticamente en el catálogo de productos una vez rellenados todos los campos necesarios para añadir un nuevo producto. Se utilizará entonces el método `addNewProduct` que aparece en la figura [6.16](#).

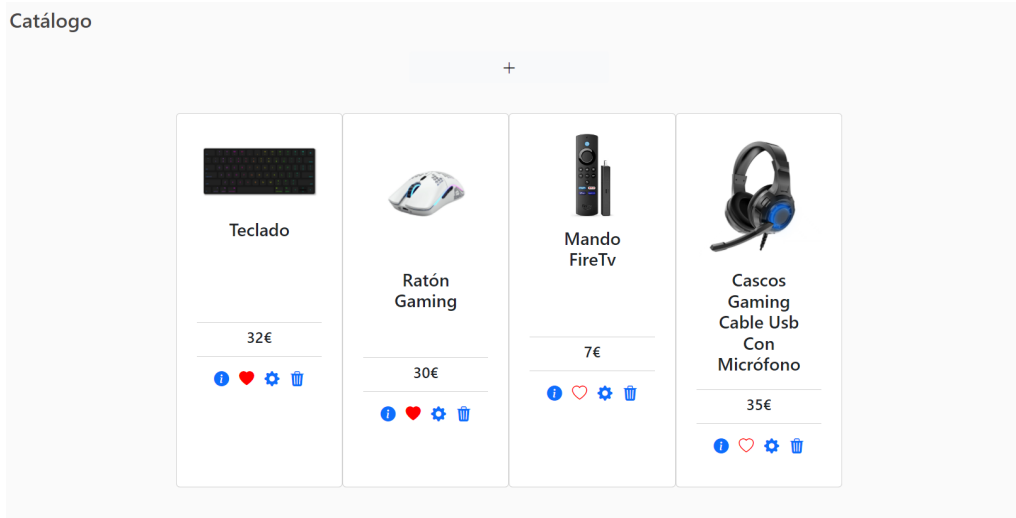


Figura 6.14: Catálogo de productos

Registrar Producto

Título

Descripción sobre el producto

Precio

€

Imagen

Seleccionar archivo Ninguno archivo selec.

Añadir Producto

Figura 6.15: Registrar producto

```

addNewProduct(registerForm: NgForm) {
  const imagesRef = ref(this.storage, 'images')
  var formValue = registerForm.value

  let image = formValue["img"].substring(12,formValue.length)

  listAll(imagesRef)
  .then(async response => {
    for(let item of response.items){
      console.log(item.name + "==" + image)
      if (item.name == image){
        const url = await getDownloadURL(item)
        this.product = new Product(formValue["title"],formValue["description"],formValue["price"],url,"",[],[],[],[],[])
        this.dataService.saveProduct(this.product)
        registerForm.reset()
        this.getProductData()
        this.router.navigate(['/products'])
      }
    }
  })
  .catch(error => console.log(error))
}

```

Figura 6.16: addNewProduct()

```

saveProduct(product:Product){
  this.httpClient.post('https://prediccion-de-ventas-default-rtdb.europe-west1.' +
  'firebase.database.app/products.json', product).subscribe(
    response=> {
      console.log("Correct register: " + product.title)
      // @ts-ignore
      product.key = response['name']
      this.updateProduct(product)
    },
    error=>console.error("Error saving user: " + product.title + ' -> ' + error)
  );
}

```

Figura 6.17: dataService.saveProduct()

6.2.2. Dar de baja un producto

Con la cuenta de administrador, una vez dentro del catálogo de productos, en cada producto se puede ver un icono azul con forma de papelera, icono de bootstrap. Clicando en el icono,

que aparece en la figura [6.14](#), se podrán eliminar los productos del catálogo. Se desplegará un modal de confirmación como aparece en la figura [6.18](#). El proceso de eliminación del producto aparece en la figura [6.19](#).

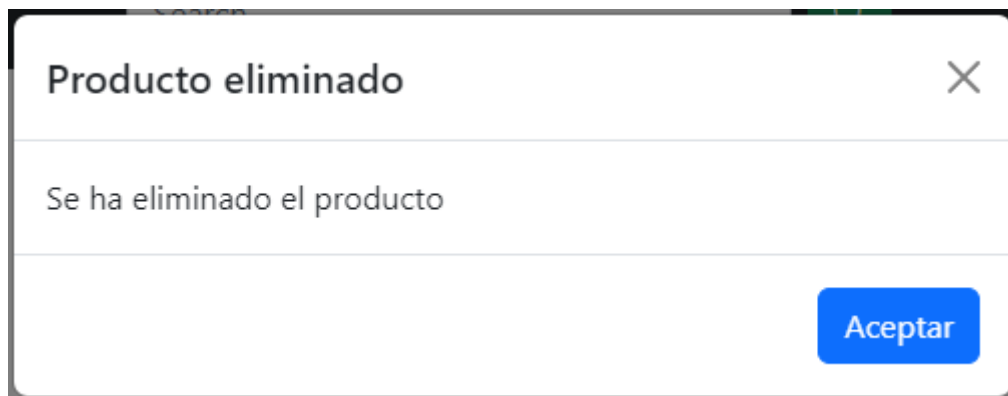


Figura 6.18: Baja de producto

```
deleteProduct(title:string){
  let p = this.getProductByTitle(title)
  this.dataService.deleteProduct(p)
  if (p !== Product.emptyProduct()){
    this.dataService.loadUsers().subscribe(user=>{
      this.allUsers = Object.values(user)
      this.allUsers.forEach(user => {
        user.favourite_products = user.favourite_products.filter(prod =>{
          return prod !== p.key
        })
        this.dataService.updateUser(user)
        if(user.email == this.cookie.get("email"))
          this.cookie.set("favouriteProducts", user.favourite_products.toString())
      })
      this.getProductData()
      this.router.navigate(['/products'])
    })
  }
}
```

Figura 6.19: `deleteProduct()`

```
deleteProduct(product:Product){
  this.httpClient.delete('https://prediccion-de-ventas-default-rtdb.europe-west1.' +
    'firebase.database.app/productos/' + product.key + '/.json').subscribe(
    response=>console.log("Correct deletion of user " + product.title),
    error=>console.error("Error on user deletion: " + product.title + ' -> ' + error)
  );
}
```

Figura 6.20: `dataService.deleteProduct()`

6.2.3. Modificar un producto

En el catálogo de productos si se está registrado con la cuenta de administrador, se puede modificar productos clicando en el icono de ajustes de cada producto, que desplegará un modal con todos los datos a modificar de un producto como aparece en la figura [6.21](#).

Para el diseño de este modal se utilizan campos input de bootstrap y un botón de envío que utiliza el color de bootstrap `button-dark`.

Modificar Producto ✕

Título

Descripción sobre el producto

Precio

€

Imagen

Seleccionar archivo | Ninguno archivo selec.

Modificar Producto

Figura 6.21: Modificar Producto

```

updateProduct(product:Product){
  if(product.key) {
    this.httpClient.put('https://prediccion-de-ventas-default-rtdb.europe-west1.'+
      'firebase.database.app/products/' + product.key + '/.json', product).subscribe(
      response=>console.log("Correct update: " + product.title),
      error=>console.error("Error updating product: " + product.title + ' -> ' + error)
    );
  }else {
    alert("Trying to update with empty key for product: " + product.title)
  }
}
}

```

Figura 6.22: updateProduct()

6.2.4. Eliminar valoración de un producto o un comentario en el foro

Accediendo a los detalles de producto, clicando en el icono de información de un producto de la vista de “Catálogo”, podemos ver los detalles de dicho producto y algo más abajo las valoraciones que tiene un producto, figura [6.23](#). Con cualquier usuario podemos añadir una valoración. Como usuario administrador podemos modificar y eliminar valoraciones, que desplegarán modales como el que se puede observar en la figura [6.24](#).

Para eliminar una valoración clicaremos en el icono de bootstrap azul de la papelera que aparece asociado a cada valoración en cada fila de la tabla de valoraciones.

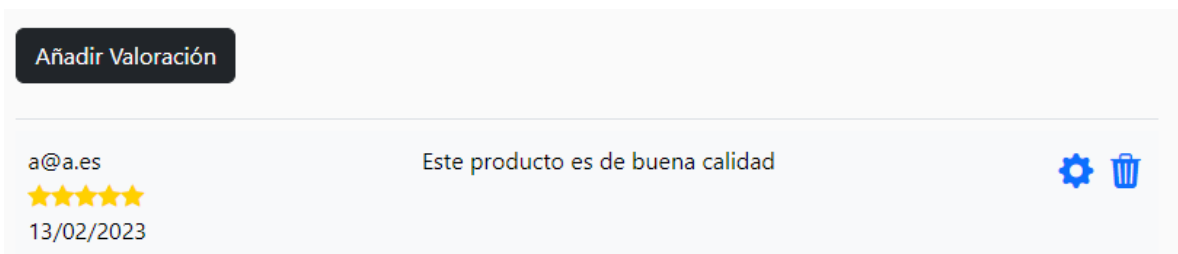


Figura 6.23: Valoraciones



Figura 6.24: Eliminar Valoración

```
deleteReview(){
  this.product.userRating.splice(this.id, 1);
  this.product.rating.splice(this.id,1)
  this.product.comment.splice(this.id,1)
  this.product.ratingDate.splice(this.id,1)
  this.dataService.updateProduct(this.product)
}
```

Figura 6.25: deleteReview()

```
updateProduct(product:Product){
  if(product.key) {
    this.httpClient.put('https://prediccion-de-ventas-default-rtdb.europe-west1.'+
      'firebase.database.app/productos/' + product.key + '/.json', product).subscribe(
      response=>console.log("Correct update: " + product.title),
      error=>console.error("Error updating product: " + product.title + ' -> ' + error)
    );
  }else {
    alert("Trying to update with empty key for product: " + product.title)
  }
}
```

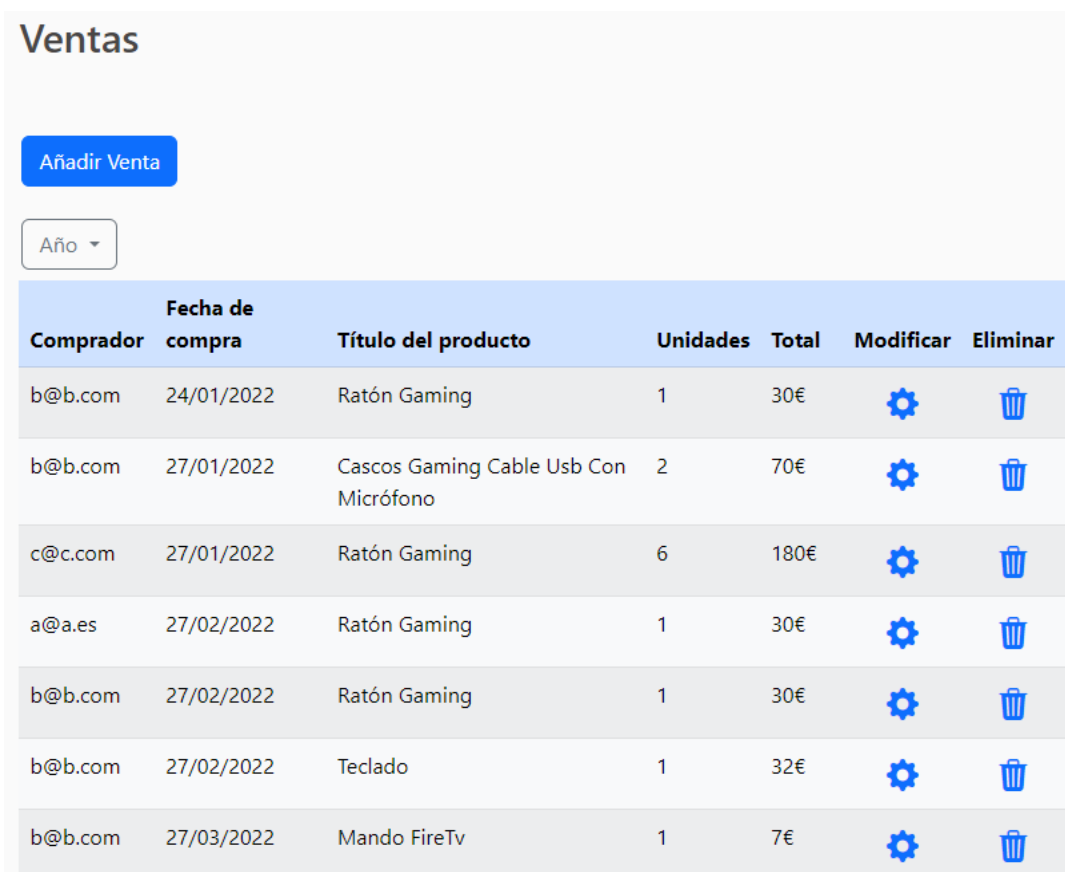
Figura 6.26: dataService.updateProduct()

6.3. Módulo Gestión de Ventas

6.3.1. Registrar ventas de un producto

Accediendo a la barra de navegación, clicando en “Registro de Ventas” aparece una tabla bootstrap con las ventas totales como puede verse en la figura [6.27](#). Para registrar una nueva venta clicar en “Añadir Venta”. Se despliega un modal, figura [6.28](#), en el que se elige comprador y producto y el número de unidades que se le vendieron a ese cliente. Una vez seleccionados todos los campos hacer clic en “Registrar Venta”.

Para añadir una venta tenemos el botón “Añadir Venta” con el diseño de color azul que ofrece el botón de bootstrap de clase button-primary.



Comprador	Fecha de compra	Título del producto	Unidades	Total	Modificar	Eliminar
b@b.com	24/01/2022	Ratón Gaming	1	30€		
b@b.com	27/01/2022	Cascos Gaming Cable Usb Con Micrófono	2	70€		
c@c.com	27/01/2022	Ratón Gaming	6	180€		
a@a.es	27/02/2022	Ratón Gaming	1	30€		
b@b.com	27/02/2022	Ratón Gaming	1	30€		
b@b.com	27/02/2022	Teclado	1	32€		
b@b.com	27/03/2022	Mando FireTv	1	7€		

Figura 6.27: Registro de Ventas

Registro de nueva venta

Elige comprador

Elige producto

Unidades

1

Cerrar Registrar Venta

Figura 6.28: Registro de nueva venta

```
registerNewSale(salesForm:NgForm){
  var formValue = salesForm.value
  let sale = Sale.emptySale()
  let product = this.products.filter(prod =>{
    return prod.title == formValue.title
  })

  sale.date = this.getCurrentDate()
  sale.email = formValue.email
  sale.quantity = formValue.quantity

  sale.title = formValue.title
  sale.price = product[0].price
  sale.product_key = product[0].key

  this.dataService.saveSale(sale)
  this.sales.push(sale)
  this.calculateTotalEarned()
}
```

Figura 6.29: registerNewSale()

```

saveSale(sale:Sale){
  this.httpClient.post('https://prediccion-de-ventas-default-rtdb.europe-west1.'+
    'firebasedatabase.app/ventas.json', sale).subscribe(
    response=> {
      console.log("Correct sale save: " + sale.email + " " + sale.title + " " +
        sale.price)
      // @ts-ignore
      sale.key = response['name']
      this.updateSale(sale)
    },
    error=>console.error("Error saving sale: " + sale.email + " " + sale.title +
      " " + sale.price + " -> " + error)
  );
}

```

Figura 6.30: *dataService.saveSale()*

6.3.2. Modificar número de ventas de un producto

En la vista de Registro de Ventas, clicando en el botón de Modificar de cada venta que aparece en la tabla con el icono gear de bootstrap, aparece un modal en el que se podrá modificar la información que se ha registrado de esa venta. Se nos despliega un modal en el que se pueden modificar los datos comprador, producto y la cantidad de unidades vendidas. A parte se podrán observar los datos que ya posee esa venta que se quieren modificar. El modal puede observarse en la figura [6.31](#).

Modificar venta
✕

Comprador: **b@b.com**
 Fecha de compra: **24/01/2022**
 Título del producto: **Ratón Gaming**
 Precio del producto: **30€**
 Cantidad: **1 unidades**
 Total: **30€**

Cantidad

Cerrar
Modificar Venta

Figura 6.31: Modificar venta

```

modifySale(salesForm:NgForm){
  var formValue = salesForm.value
  let sale = this.sales[this.id]

  if(formValue.email)
    sale.email = formValue.email

  if(formValue.quantity)
    sale.quantity = formValue.quantity

  if(formValue.title){
    let product = this.products.filter(prod =>{
      return prod.title == formValue.title
    })
    sale.title = formValue.title
    sale.price = product[0].price
    sale.product_key = product[0].key
  }

  this.dataService.updateSale(sale)
  this.sales[this.id] = sale
  this.calculateTotalEarned()
}

```

Figura 6.32: *modifySale()*

```
updateSale(sale:Sale){
  if(sale.key) {
    this.httpClient.put('https://prediccion-de-ventas-default-rtdb.europe-west1.'+
      'firebase.database.app/ventas/' + sale.key + '/.json', sale).subscribe(
      response=>console.log("Correct sale update: " + sale.email + " " +
        sale.title + " " + sale.price),
      error=>console.error("Error updating sale: " + sale.email + " " +
        sale.title + " " + sale.price)
    );
  }else
    alert("Trying to update with empty key for sale: " + sale.email + " " +
      sale.title + " " + sale.price)
}
```

Figura 6.33: *dataService.updateSale()*

6.3.3. Registrar devolución de un producto

Para cada venta que aparece en la tabla de la vista Registro de Ventas, existirá la opción de eliminar. Haciendo así que una venta que se había registrado, se dé de baja, no sumando el total de esa venta al total de ventas y no volviendo a aparecer en la tabla, lo que haría la función de devolver un producto de la tienda. Clicando en el icono de la papelera para eliminar la venta que se desee devolver y después en el botón “Eliminar” del modal que aparece en la [figura 6.34](#).

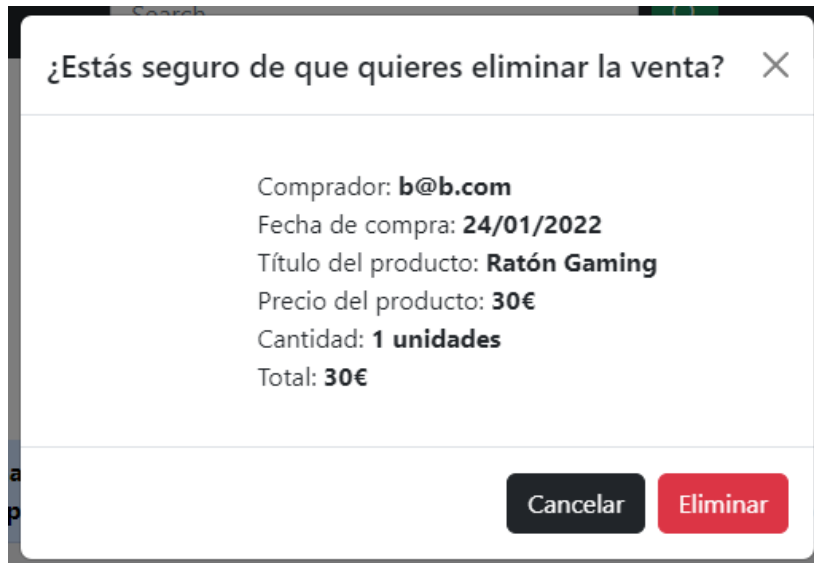


Figura 6.34: Eliminar venta

```
deleteSale(){
  let sale = this.sales[this.id]
  this.dataService.deleteSale(sale)
  this.sales.splice(this.id, 1)
  this.calculateTotalEarned()
}
```

Figura 6.35: deleteSale()

```
deleteSale(sale:Sale){
  this.httpClient.delete('https://prediccion-de-ventas-default-rtdb.europe-west1.'+
'firebase.firebaseio.com/ventas/' + sale.key + '/.json').subscribe(
  response=>console.log("Correct deletion of sale " + sale.email),
  error=>console.error("Error on user deletion: " + sale.email + ' -> ' + error)
);
}
```

Figura 6.36: dataService.deleteSale()

6.3.4. Mostrar caja de tienda

La caja de tienda es la propia tabla de la vista Registro de Ventas en la que se pueden ver todas las ventas registradas hasta la fecha. También se podrán filtrar por mes y año con los botones de arriba “Año” y “Mes”. Al final de la tabla se calcula el total de las ventas para el año y mes seleccionados. Puede verse en la [figura 6.37](#).

La tabla tiene el diseño de bootstrap table-striped con el color de header table-primary.



Año ▾		Mes ▾				
Fecha de compra	Título del producto	Unidades	Total	Modificar	Eliminar	
b@b.com	27/02/2022	Ratón Gaming	1	30€		
b@b.com	27/02/2022	Ratón Gaming	1	30€		
b@b.com	27/02/2022	Teclado	1	32€		
Total	6/05/2023	3	92€			

Figura 6.37: Filtro de Ventas

6.4. Módulo Informes de Productos

6.4.1. Informe sobre los productos más escaneados y con relación a las ventas, edad o género

Con la cuenta de administrador, en la pantalla Home, una vez habiendo hecho login en la aplicación, aparece un apartado de la página con el título “Predicción de Ventas”, justo debajo hay un botón rojo con un gráfico como muestra la [figura 6.38](#). Clicando en este botón

se desplegará un gráfico de “Highcharts” con las ventas hasta la fecha y una predicción de ventas que se refleja a partir de los meses siguientes al mes actual como muestran las figuras [6.39](#) y [6.40](#). Se realiza con tres métodos:

- RunRate, también conocido como tasa de ejecución o tasa de ejecución anualizada, es una técnica utilizada para proyectar o estimar el rendimiento o los resultados futuros en función de los datos actuales. Se aplica comúnmente para estimar los ingresos, gastos o utilidades anuales en función de los datos disponibles en un período más corto, como un mes o un trimestre. Se calcula extrapolando o proyectando los resultados actuales en un período de tiempo más largo. El cálculo se realiza dividiendo las ventas totales en la cantidad de meses transcurridos en el año contable utilizando esa como la media de ventas que se proyecta a futuro. El método aparece desarrollado en la [figura 6.45](#). Para este método en el gráfico se ha utilizado una línea por puntos de color negro.
- Seasonal(Standard) es una aproximación de la cantidad de ventas que se tendrían por cada época del año, basadas en las ventas por temporada del año de una tienda online real de la que el autor es propietario (Amazon Seller[28]), ya que hay temporalidades en las que se vende más y conviene ajustar una predicción a dichos sucesos para poder así anticipar el control del stock. Esta medida no pretende ser más que un modelo basado en un ejemplo de la vida real, el cual podría ser ajustado en el código para futuras predicciones. Puede verse el desarrollo del código en la [figura 6.46](#). Para este método se ha utilizado el color naranja en forma de barra en el gráfico.
- Seasonal(Last Year) que mezcla la media de RunRate y un aprendizaje obtenido en base al factor de ventas que se obtuvieron el año pasado en cada mes del año. El cálculo se realiza cargando una lista con las ventas totales por cada mes de las ventas que existieron el año anterior y divide cada una de ellas por las ventas totales que existieron en la totalidad del año contable anterior. De esta forma se obtiene un factor por cada mes que indicará cuáles meses tenderán a ser más propensos a tener más volumen de ventas. Este factor finalmente irá multiplicado por el resultado del método RunRate explicado previamente, obteniendo así una mezcla entre una predicción basada en ventas pasadas y una proyección basada en la actividad reciente de la tienda. El algoritmo aparece en detalle en la [figura 6.47](#). Para mostrar este método se ha hecho uso del color verde en el gráfico en forma de gráfico de barras.

Se puede también elegir el Modo de gráfico que se podrá actualizar clicando el botón que aparece encima del gráfico “Modo”, se podrá elegir entre visualizar una predicción sobre las ventas, las unidades o una visualización en forma de gráfico por sectores para conocer la cantidad de visitas que se han registrado por cada producto.

Predicción de Ventas

Obtén una predicción de ventas basada en la tendencia de ventas del año actual y anterior por cada temporada



Figura 6.38: Predicción de ventas

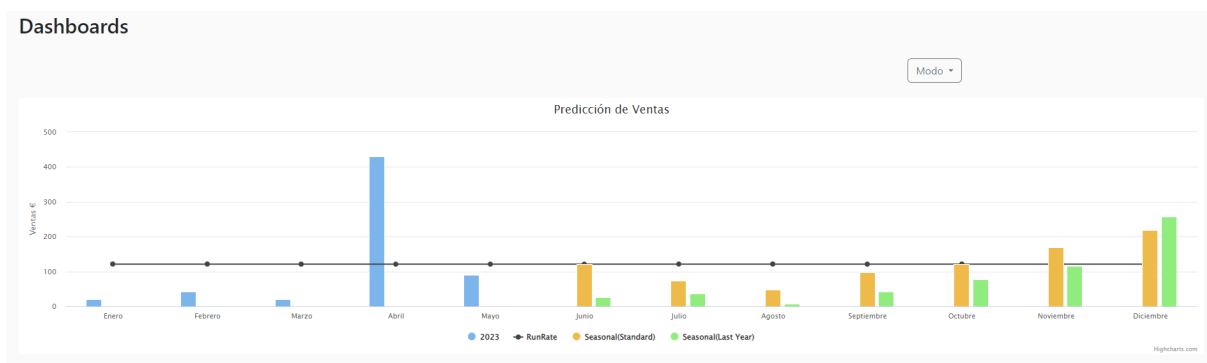


Figura 6.39: Predicción de ventas()

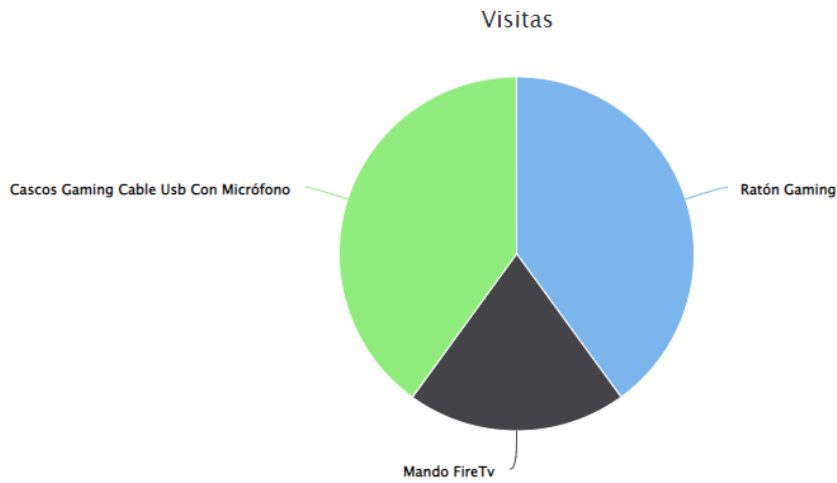


Figura 6.40: Gráfico de visitas

```

linechart3: any = {
  series: [
    {
      name: this.year,
      data: this.productChart
    }
  ],
  chart: {
    type: 'pie',
  },
  title: {
    text: 'Visitas ',
  },
  xAxis: {
    categories: this.productNames
  },
  yAxis: {
    title: {
      text: 'Visitas a Productos'
    }
  },
}

```

Figura 6.41: linechart

```
loadChart(dispatch: Boolean){
  this.display = disp
  this.yearSales(this.year)
  this.yearUnitsSold(this.year)
  this.runRateMethod()
  this.seasonalMethod()
  this.lastYearSeasonalMethod()
}
```

Figura 6.42: loadChart()

```
yearSales(year: string){
  console.log("This year sales: ", this.sales)

  this.ySales = this.sales.filter(sale=>{
    let y = "0"
    if (sale.date){
      let arrDate = sale.date.split("/")
      // console.log("Date: ", arrDate)
      y = arrDate[2]
    }
    // console.log("Date: ", sale.date)
    return year == y
  })

  console.log("yearSales", this.ySales)

  this.ySales.forEach(sale => {
    let arrDate = sale.date.split("/")
    let m = arrDate[1]
    this.allSalesInYear[parseInt(m) - 1] += sale.price * sale.quantity
  })

  console.log("allSalesInYear", this.allSalesInYear)
}
```

Figura 6.43: yearSales()

```

yearUnitsSold(year:string){
  console.log("This year sales: ", this.sales)

  this.ySales = this.sales.filter(sale=>{
    let y = "0"
    if (sale.date){
      let arrDate = sale.date.split("/")
      console.log("Date: ", arrDate)
      y = arrDate[2]
    }
    console.log("Date: ", sale.date)
    return year == y
  })

  console.log("yearSales", this.ySales)

  this.ySales.forEach(sale => {
    let arrDate = sale.date.split("/")
    let m = arrDate[1]
    this.allUnitsInYear[parseInt(m) - 1] += sale.quantity
  })

  console.log("allUnitsInYear", this.allUnitsInYear)
}

```

Figura 6.44: yearUnitsSold()

```

runRateMethod(){
  let total = 0
  let totalUnits = 0
  let actualMonth = this.getActualMonth()
  let runRate: number[]=[0,0,0,0,0,0,0,0,0,0,0,0]
  let runRateUnits: number[]=[0,0,0,0,0,0,0,0,0,0,0,0]

  this.allSalesInYear.forEach(sale=>{
    | total += sale
  })

  this.allUnitsInYear.forEach(unit=>{
    | totalUnits += unit
  })

  console.log("Total: ", total)
  console.log("ActualMonth: ", actualMonth)

  //Number of sales per month
  let currentAvg = total / actualMonth
  let currentAvgUnits = totalUnits / actualMonth
  for(let i = 0; i < 12; i++){
    | runRate[i] = currentAvg
    | runRateUnits[i] = currentAvgUnits
  }

  this.runRate = runRate
  this.runRateUnits = runRateUnits

  this.linechart2.series.push({name: "RunRate", data: runRateUnits, type:"line"})
  this.linechart.series.push({name: "RunRate", data: runRate, type: "line"})
}

```

Figura 6.45: runRateMethod()

```

seasonalMethod(){
  //Basado en datos de una tienda online real de Amazon Seller

  let seasonal: number[] = [0,0,0,0,0,0,0,0,0,0,0,0]
  let seasonalUnits: number[] = [0,0,0,0,0,0,0,0,0,0,0,0]

  let actualMonth = this.getActualMonth()
  for(let i = actualMonth; i < 12; i++){
    let factor = 1

    if( i == 1){
      factor = 1
    }else if ( i == 2 ){
      factor = 0.8
    }else if ( i == 3 ){
      factor = 0.85
    }else if( i == 4 ){
      factor = 1.15
    }else if(i == 5){
      factor = 1
    }else if( i == 6){
      factor = 0.6
    }else if ( i == 7){
      factor = 0.4
    }else if ( i == 8){
      factor = 0.8
    }else if ( i == 9){
      factor = 1
    }else if ( i == 10){
      factor = 1.40
    }else if ( i == 11){
      factor = 1.80
    }

    seasonal[i] = Math.round(factor * this.runRate[i])
    seasonalUnits[i] = factor * this.runRateUnits[i]
  }
  this.linechart2.series.push({name: "Seasonal(Standard)", data: seasonalUnits, color: "#EEBB4A"})
  this.linechart.series.push({name: "Seasonal(Standard)", data: seasonal, color: "#EEBB4A"})
}

```

Figura 6.46: *seasonalMethod()*

```

lastYearSeasonalMethod(){
  let allSalesInYear=[0,0,0,0,0,0,0,0,0,0,0,0]
  let allUnitsInYear=[0,0,0,0,0,0,0,0,0,0,0,0]
  let seasonallLastYear = [0,0,0,0,0,0,0,0,0,0,0,0]
  let seasonalUnitsLastYear=[0,0,0,0,0,0,0,0,0,0,0,0]
  let lastYearFactors=[0,0,0,0,0,0,0,0,0,0,0,0]

  console.log("This year sales: ", this.sales)

  let lastySales = this.sales.filter(sale=>{
    let y = "0"
    if (sale.date){
      let arrDate = sale.date.split("/")
      console.log("Date: ", arrDate)
      y = arrDate[2]
    }
    console.log("Date: ", sale.date)
    return (parseInt(this.year) - 1).toString() == y
  })

  console.log("yearSales", this.ySales)

  let total = 0
  lastySales.forEach(sale => {
    total += sale.price*sale.quantity
    let arrDate = sale.date.split("/")
    let m = arrDate[1]
    allSalesInYear[parseInt(m) - 1] += sale.quantity * sale.price
    allUnitsInYear[parseInt(m) - 1] += sale.quantity
  })

  let i = 0
  allSalesInYear.forEach(sale=>{
    lastYearFactors[i] = (sale / total)
    i++
  })

  for(let i = this.getActualMonth(); i < 12; i++){
    seasonallLastYear[i] = Math.round(lastYearFactors[i] * this.runRate[i] * 10)
    seasonalUnitsLastYear[i] = lastYearFactors[i] * this.runRateUnits[i] * 10
  }

  this.linechart.series.push({name: "Seasonal(Last Year)", data: seasonallLastYear})
  this.linechart2.series.push({name: "Seasonal(Last Year)", data: seasonalUnitsLastYear})

  console.log("lastYearFactors: ", this.lastYearFactors)
}

```

Figura 6.47: lastYearSeasonalMethod()

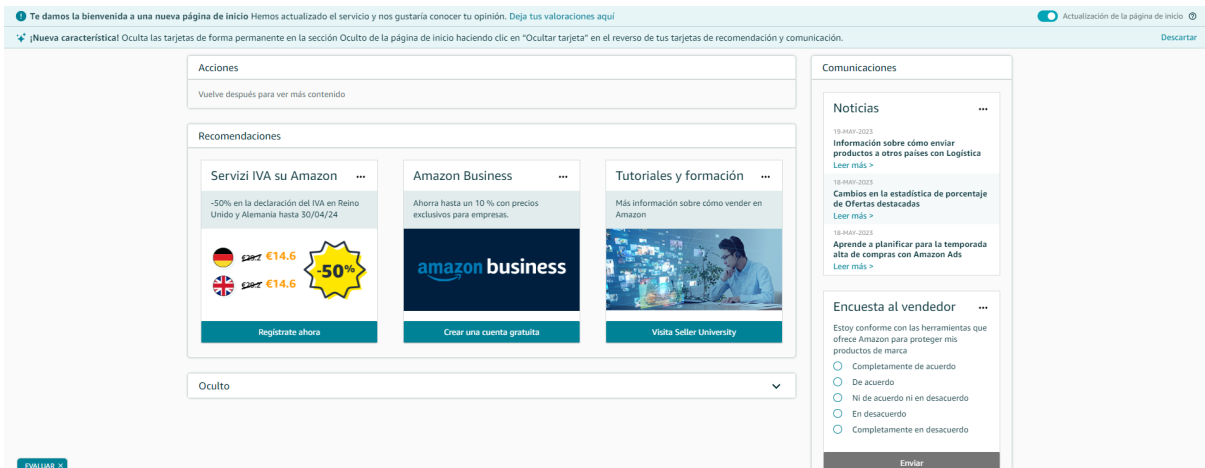


Figura 6.48: Amazon Seller

6.4.2. Generar/Modificar/Publicar encuesta para clientes

Desde la barra de navegación estando registrado con cuenta de administrador, y haciendo clic en “Encuestas” se accede a la vista Encuestas, mostrado en la figura 6.49. Una vez en Encuestas, haciendo clic en el botón “Nueva Encuesta”, aparecerá un modal para introducir título de la encuesta y descripción de la encuesta como se observa en la figura 6.50. Una vez rellenados ambos campos y habiendo hecho clic en enviar, se creará la nueva encuesta con el botón “Añadir Campo” para ir añadiendo nuevas preguntas a la encuesta como puede verse en la figura 6.51. Clicando en dicho botón se abrirá un modal en el que se deberá elegir el tipo de pregunta y el título de la pregunta, figura 6.52. Una vez la encuesta tenga las preguntas que se deseen añadir, hacer clic en publicar encuesta para hacerla llegar a los usuarios elegidos, desplegando el modal visto en la figura 6.53, otra opción es hacer clic en guardar para guardar la encuesta.

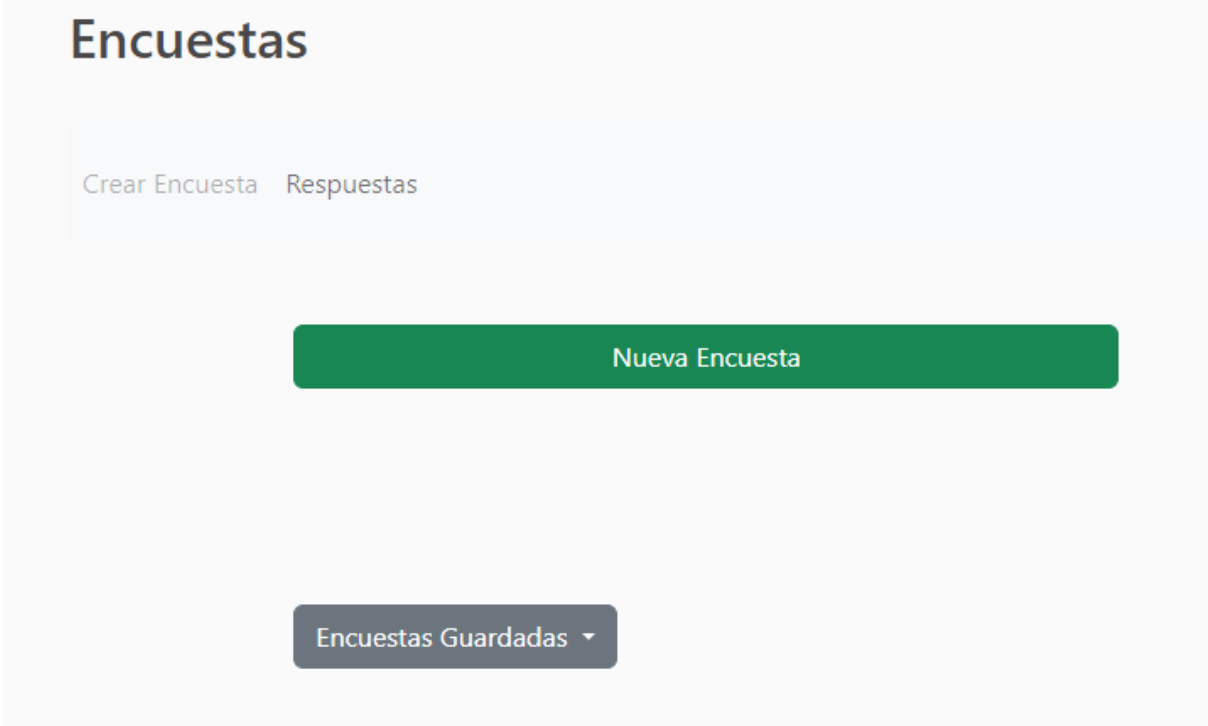


Figura 6.49: Encuestas

Añade el Título de la Encuesta

Título:

Descripción:

Cerrar Añadir

Figura 6.50: Añadir título de la encuesta

Encuestas

Crear Encuesta Respuestas

Título 1

Descripción de encuesta 1

Añadir Campo

Guardar Encuesta

Publicar Encuesta

Figura 6.51: Nueva encuesta

Nueva Pregunta ✕

Tipo:

Título:

Cerrar Añadir

Figura 6.52: Añadir Campo

Publicar Encuesta ✕

El formulario se guardará automáticamente al publicarlo

Selecciona Grupo de Usuarios ▾

Cerrar Publicar

The image shows a modal dialog box with a title bar containing the text 'Publicar Encuesta' and a close button (✕). Below the title bar, there is a message: 'El formulario se guardará automáticamente al publicarlo'. Underneath this message is a dropdown menu with the text 'Selecciona Grupo de Usuarios' and a downward arrow (▾). At the bottom right of the dialog, there are two buttons: a grey 'Cerrar' button and a blue 'Publicar' button.

Figura 6.53: Publicar Encuesta

```

publishFeedback(ngForm : NgForm){
  var emails = ngForm.value.emails
  console.log(emails)
  if(emails){
    if(emails != "allUsers"){
      let product = Product.emptyProduct()
      this.products.forEach(prod =>{
        if(prod.key == emails)
          product = prod
      })
      this.users = this.users.filter(user=>{
        return product.userViews.includes(user.email)
      })
      console.log("Users with views in product: ", this.users)
    }
    this.users.forEach(user=>{
      if(!user.sharedFeedbacks)
        user.sharedFeedbacks = []
      if(!user.publishDate)
        user.publishDate = []
      if(!user.feedbackDone)
        user.feedbackDone = []
      if(!user.feedbackOpened)
        user.feedbackOpened = []
      user.sharedFeedbacks.push(this.feedBack.title)
      user.feedbackDone.push(false)
      user.feedbackOpened.push(false)
      this.feedBack.publishDate = this.getCurrentDate()
    })
    console.log("Users to share with: ", this.users)
    this.users.forEach(user => {
      this.dataService.updateUser(user)
    })
  }
}

```

Figura 6.54: *publishFeedback()*

```

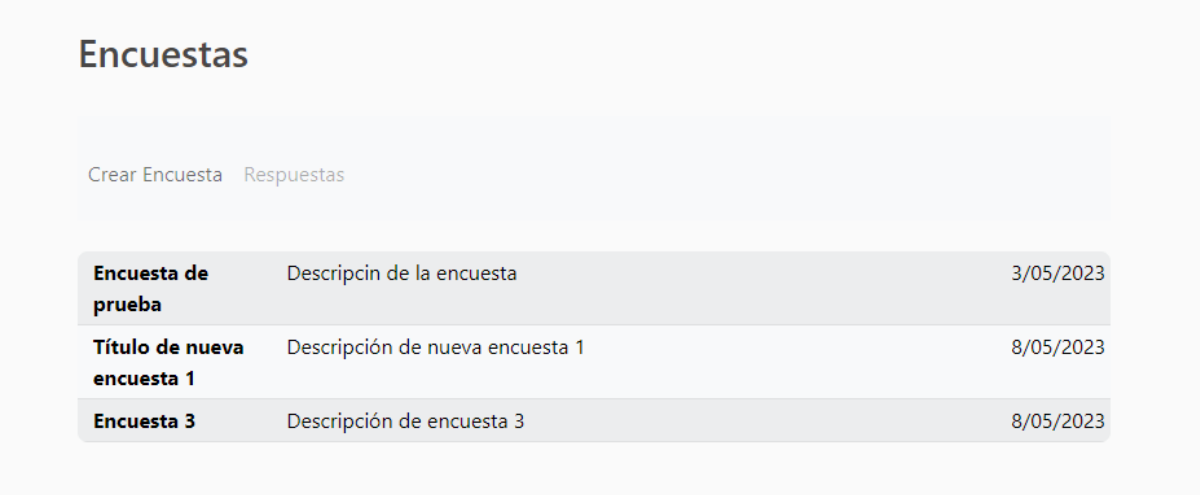
updateUser(user:User){
  if(user.key) {
    this.httpClient.put('https://prediccion-de-ventas-default-rtdb.europe-west1.' +
      'firebase.database.app/usuarios/' + user.key + '/.json', user).subscribe(
      response=>console.log("Correct update: " + user.email),
      error=>console.error("Error updating user: " + user.email + ' -> ' + error)
    );
  }else {
    alert("Trying to update with empty key for user: " + user)
  }
}

```

Figura 6.55: *updateUser()*

6.4.3. Buscar y ver resultados de la encuesta

Con la cuenta de administrador es posible ver los resultados de la encuesta, para eso en la página “Encuestas” como aparece en la figura [6.56](#), hacer clic en el botón “Respuestas” de la barra de navegación, entonces se mostrarán todas las respuestas en forma de tabla “table-striped” con el título, descripción y fecha de publicación de cada una. Clicando en cada una de las encuestas de la tabla se pueden conocer las respuestas que dieron cada uno de los usuarios encuestados como se observa en la figura [6.57](#).



The screenshot shows a web interface titled "Encuestas". Below the title, there are two buttons: "Crear Encuesta" and "Respuestas". Below these buttons is a table with three rows, each representing a survey. The table has three columns: the survey title, the description, and the date. The rows are: 1. "Encuesta de prueba" with description "Descripción de la encuesta" and date "3/05/2023". 2. "Título de nueva encuesta 1" with description "Descripción de nueva encuesta 1" and date "8/05/2023". 3. "Encuesta 3" with description "Descripción de encuesta 3" and date "8/05/2023".

Encuesta de prueba	Descripción de la encuesta	3/05/2023
Título de nueva encuesta 1	Descripción de nueva encuesta 1	8/05/2023
Encuesta 3	Descripción de encuesta 3	8/05/2023

Figura 6.56: Respuestas

Encuesta de satisfacción de Ratón Gaming ✕

User: b@b.com

Queremos saber tu opinión acerca de el artículo que visitaste en el catálogo "Ratón Gaming".

1. ¿Cuánto te llama la atención este producto?
4

2. ¿Comprarías este producto?
Si

3. ¿Por qué?
Me parece un producto de buena relación calidad/precio

User ▾

Cerrar

Figura 6.57: Respuestas de usuarios encuestados

6.5. Servicios Orientados al Usuario

6.5.1. Realizar lectura de QR de un producto

Todos los usuarios pueden realizar lecturas de QR con la aplicación. Sólo se debe haber hecho login y estar en la pantalla Home. Aparece un botón de color verde con estilo bootstrap button-success con la imagen de un escaner de QR, aquí es donde se debe clicar para iniciar el análisis de QR, como puede verse en la figura [6.58](#). Se abrirá un modal y clicando el botón de inicio iniciaremos la lectura de QR de la aplicación que nos servirá para leer QRs de productos de la aplicación y acceder a sus respectivas páginas de detalles de forma instantánea.



Figura 6.58: Lector QR

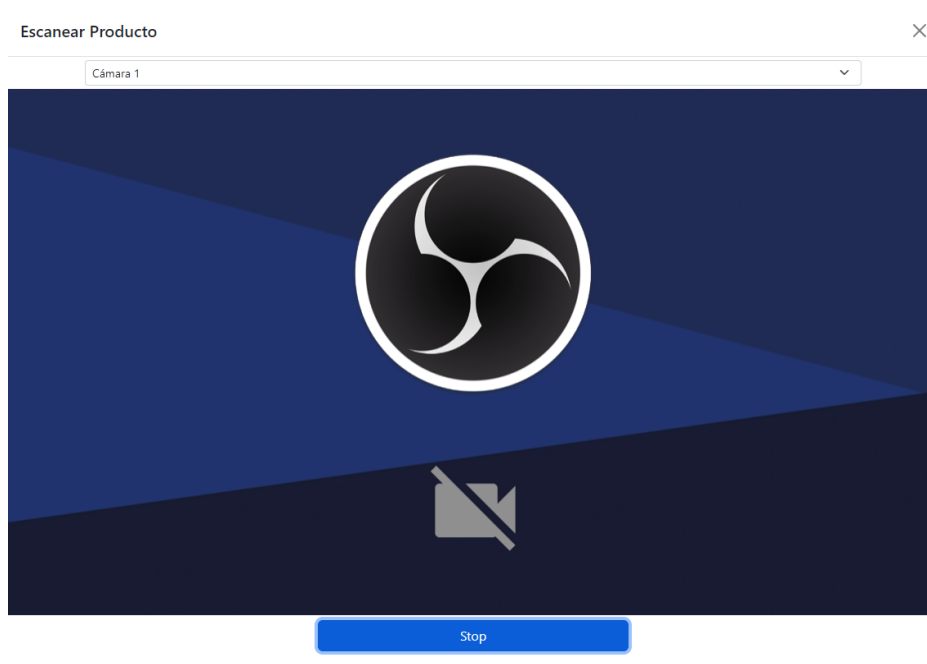


Figura 6.59: Escanear Producto

```
onCodeDecode(result:ScannerQRCodeResult[], action : any) {
  this.result = result
  if (this.result && this.result.length > 0){
    this.cookie.set("qrCode",this.result[0].value.substring(22))
    this.closeButton.nativeElement.click()
    this.router.navigate([this.cookie.get("qrCode")])
  }
}
```

Figura 6.60: onCodeDecode()

6.5.2. Añadir/Quitar producto de favoritos

El usuario tendrá una lista de productos favoritos figura [6.61](#). Accediendo al catálogo de productos podrá hacer clic en el corazón que aparece en la tarjeta de cada producto y añadir los productos que desee a su lista de favoritos. Luego podrá acceder a su lista de favoritos a

través de la barra de navegación clicando en “Mis Favoritos”, donde podrá clicar en el icono del corazón nuevamente para quitar un producto de favoritos.

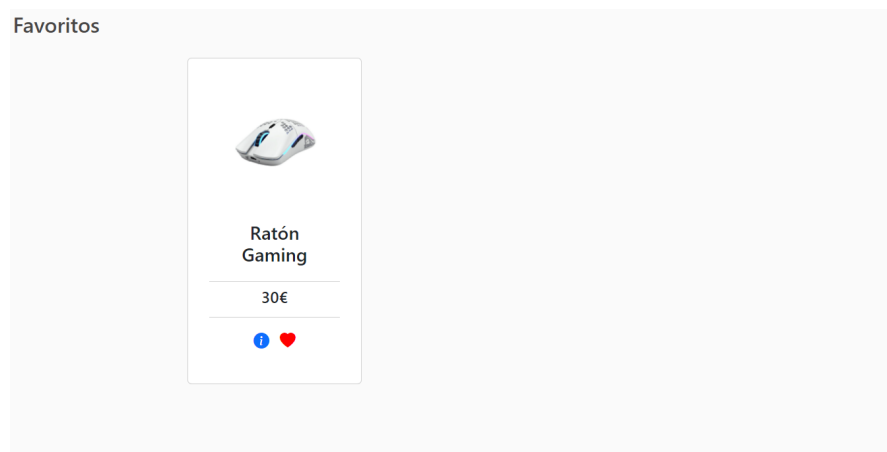


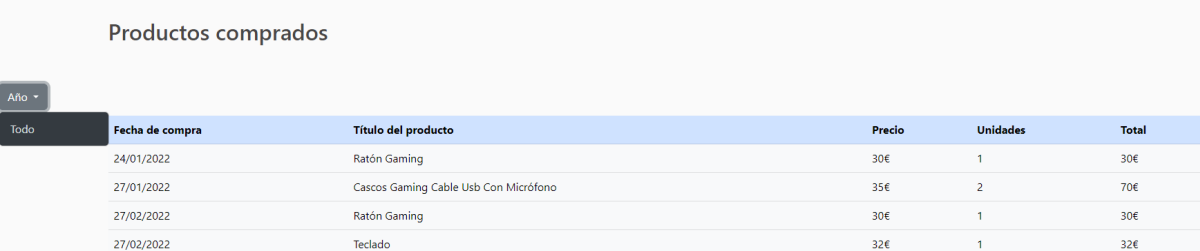
Figura 6.61: Favoritos

```
addFavouriteProduct(productKey:string){
  this.dataService.loadUsers().subscribe(user=>{
    let users = Object.values(user)
    users.forEach(user => {
      if(user.email == this.cookie.get("email")){
        if(user.favourite_products && user.favourite_products.length > 0)
          this.userProducts = user.favourite_products
        this.userProducts.push(productKey)
        user.favourite_products = this.userProducts
        this.dataService.updateUser(user)
        this.cookie.set("favouriteProducts",user.favourite_products)
        console.log(user)
      }
    });
  });
}
```

Figura 6.62: addFavouriteProduct()

6.5.3. Consultar productos comprados

Los usuarios pueden acceder a través de la barra de navegación a “Consultar Compras” donde podrán ver una lista con todos los productos comprados por ellos durante el año seleccionado como se puede ver en la figura [6.63](#).



Productos comprados

Año ▾

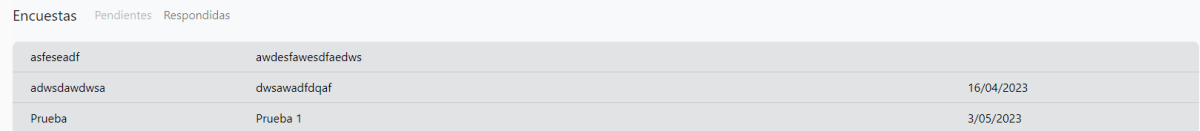
Todo

Fecha de compra	Título del producto	Precio	Unidades	Total
24/01/2022	Ratón Gaming	30€	1	30€
27/01/2022	Cascos Gaming Cable Usb Con Micrófono	35€	2	70€
27/02/2022	Ratón Gaming	30€	1	30€
27/02/2022	Teclado	32€	1	32€

Figura 6.63: Productos Comprados

6.5.4. Realizar encuesta

Los usuarios podrán detectar si han recibido una encuesta nueva a realizar, en la barra de navegación se mostrará un número con la cantidad de encuestas no abiertas. Clicando en encuestas se podrá acceder a todas las encuestas que aún no han respondido los usuarios como se puede observar en la figura [6.64](#). Clicando en cada una de ellas se abrirá la encuesta con un formulario para contestar y enviar las respuestas como se ve en la figura [6.65](#).



Encuestas Pendientes Respondidas

asfeseadf	awdesfawesdfaedws	
adwsdawdwsa	dwsawadfdqaf	16/04/2023
Prueba	Prueba 1	3/05/2023

Figura 6.64: Encuestas

Prueba

Prueba 1

Pregunta 1

Cerrar Enviar

Figura 6.65: Responder encuesta

```

formResponse(ngForm:NgForm){
  var formValue = ngForm.value
  var list = []
  list.push(this.user.email)

  let i = 0
  this.feedbacks[this.index].questions.forEach(q=>{
    list.push(formValue[i])
    i++
  })

  if(!this.feedbacks[this.index].answers)
    this.feedbacks[this.index].answers=[]

  this.feedbacks[this.index].answers.push(list)

  this.user.feedbackDone[this.index] = true
  console.log("User data: ", this.user)

  this.loadTable()

  this.dataService.updateFeedback(this.feedbacks[this.index])
  this.dataService.updateUser(this.user)
  // this.user.feedbackDone
  // this.feedbacks[this.index].answers[answers].push
}

```

Figura 6.66: formResponse()

6.5.5. Añadir/Eliminar/Modificar valoración en foro

Los usuarios podrán añadir valoraciones a los productos y valorarlos como puede verse en la figura [6.67](#). En la pantalla de detalles de cada producto existe un foro con valoraciones en los que aparecen valoraciones de todos los usuarios que han comentado. Haciendo clic en el botón “Añadir Valoración”, un usuario puede añadir su valoración sobre el producto en el que se encuentra, seleccionando un valor entre 1 y 5 y escribiendo un comentario. Después deberá darle a enviar. Añadir valoración puede verse en la figura [6.68](#).

El usuario administrador podrá modificar o eliminar dichas valoraciones.

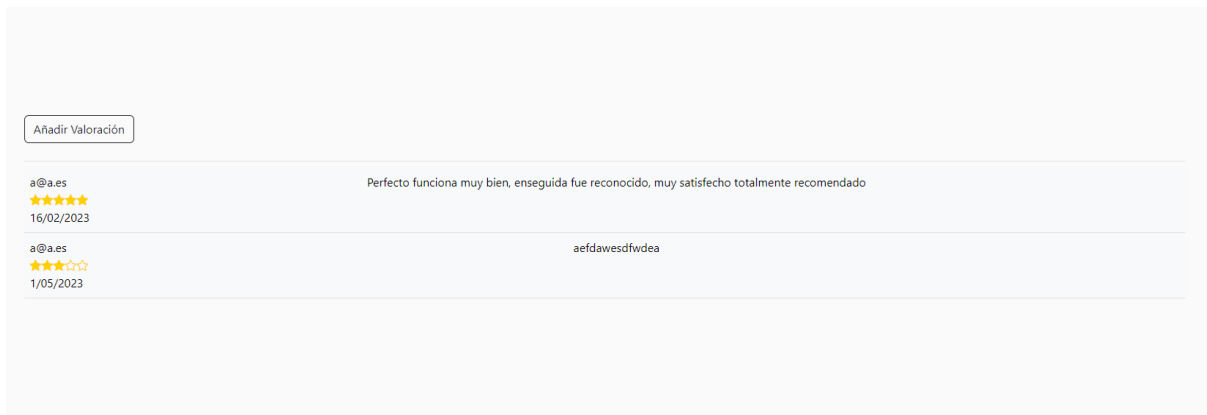


Figura 6.67: Valoraciones

Registrar Valoración ✕

Valoración:

1○ 2○ 3○ 4○ 5○

Comentario

Publicar

Figura 6.68: Añadir valoración

Capítulo 7

Evaluación de la aplicación

En este capítulo se describen los resultados obtenidos en una evaluación de usabilidad realizada entre un conjunto de usuarios.

7.1. Diseño de la evaluación

En esta evaluación se ha utilizado Google Forms, ya que permite crear y enviar cuestionarios a los usuarios de manera eficiente y sencilla para recopilar información sobre la percepción que tienen los usuarios de una aplicación. Se han hecho un total de 9 preguntas sobre la idea que tienen los usuarios del tema sobre el que trata la aplicación, su edad y sobre su satisfacción con los diferentes módulos que ofrece la aplicación puntuando del 1 al 5.

La muestra ha sido tomada de un total de 18 personas con edades comprendidas entre 21 y 66 años de edad.

7.2. Resultados de la encuesta

A continuación se muestran los resultados de la encuesta.

¿Cuál es tu edad?

Como se puede ver en la figura [7.1](#) el 66,5 % de personas que han realizado el cuestionario son menores de 30 años, mientras que el otro 33,5% ha sido de personas mayores de 39 años.

¿Cuál es tu edad?

18 respuestas

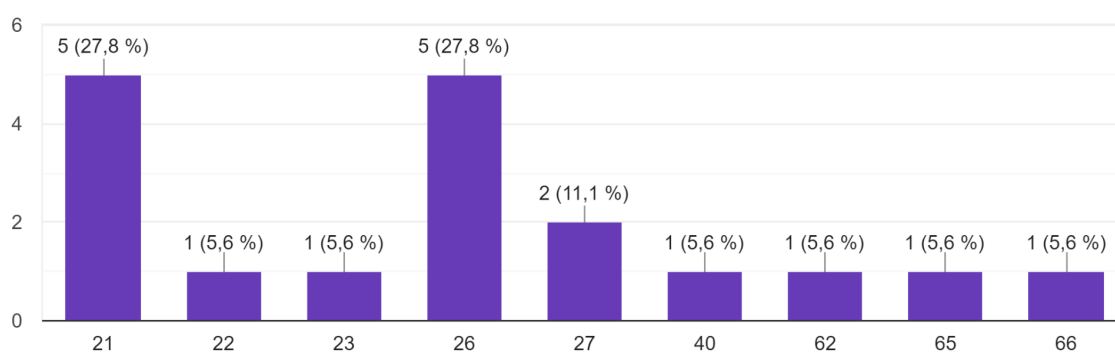


Figura 7.1: ¿Cuál es tu edad?

¿Crees que una herramienta de predicción de ventas es útil para un negocio?

La mayoría de los encuestados considera que una herramienta de predicción de ventas es útil para un negocio, lo que destaca la importancia que tienen estas herramientas para mejorar la toma de decisiones y la rentabilidad de las empresas, como se ve en la figura [7.2](#). El resto, un 11,1% opina que “Tal vez”, lo cual ni confirma ni desmiente.

¿Crees que una herramienta de predicción de ventas es útil para un negocio?

18 respuestas

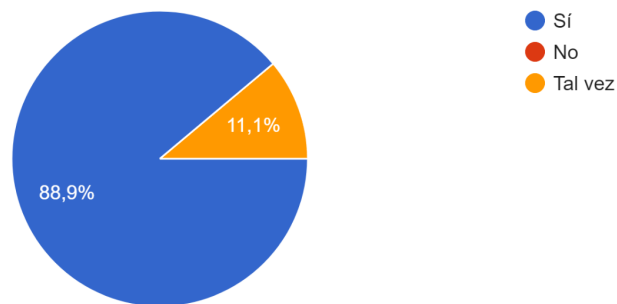


Figura 7.2: ¿Crees que una herramienta de predicción de ventas es útil para un negocio?

Nivel de satisfacción con la forma de mostrar el catálogo.

Revela que un 38,9% de los encuestados calificaron la forma de mostrar el código con un 4, mientras que un 61,1% de los encuestados la calificaron con un 5. Esto indica que la gran mayoría de los encuestados han dado la máxima puntuación a la forma de mostrar el catálogo y que hay un porcentaje que ha respondido con una puntuación cercana al valor máximo, como puede verse en la figura [7.3](#).

Nivel de satisfacción con la forma de mostrar el catálogo

18 respuestas

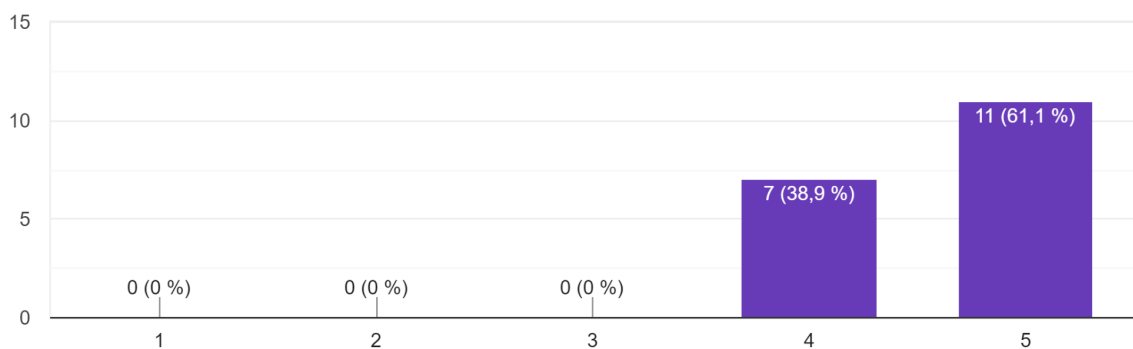


Figura 7.3: Nivel de satisfacción con la forma de mostrar el catálogo

Nivel de satisfacción con la página de detalles del producto.

Muestra que todos los encuestados menos uno han dado la máxima puntuación con la página de detalles del producto, lo que sugiere que se ha hecho un buen trabajo en su diseño y funcionalidad. Los resultados pueden verse en la figura [7.4](#).

Nivel de satisfacción con la página de detalles del producto

18 respuestas

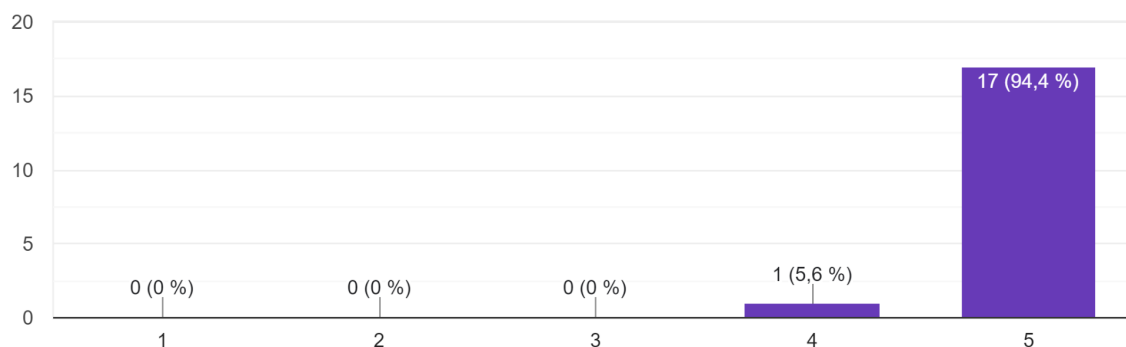


Figura 7.4: Nivel de satisfacción con la página de detalles del producto

Nivel de satisfacción con el scanner de código QR de la aplicación.

Los resultados sugieren que, en general, la mayoría de los encuestados están satisfechos con el scanner de código QR de la aplicación. El hecho de que un 61,1% de los encuestados calificara el scanner con un 5 indica que la mayoría de ellos lo consideran muy efectivo, fácil de usar y útil para escanear códigos QR. Esto es un indicador positivo, ya que un scanner de código QR eficiente puede mejorar la experiencia del usuario y aumentar la eficiencia en la realización de tareas como se puede observar en la figura [7.5](#).

También el hecho de que el resto de los encuestados calificara el scanner con un 4 o un 3 indica que, aunque están satisfechos en general, aún hay algunas áreas que pueden mejorarse.

Nivel de satisfacción con el scanner de código QR de la aplicación

18 respuestas

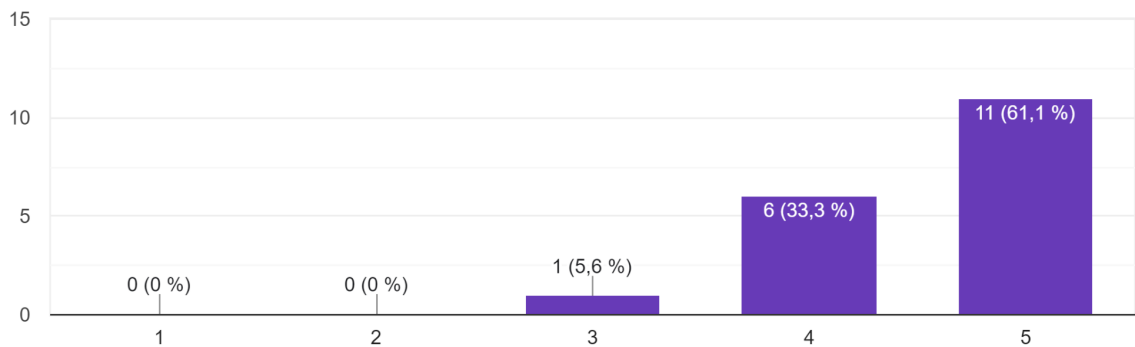


Figura 7.5: Nivel de satisfacción con el scanner de código QR de la aplicación

Nivel de satisfacción con la pantalla perfil.

Analizando los resultados, hay un pequeño porcentaje de encuestados que han votado 3 sobre 5, aunque la mayoría de los usuarios han dado la máxima puntuación o una puntuación cercana al valor máximo, siendo estos mayoría (50%). Esto podría indicar que la pantalla de perfil es efectiva y cumple con las necesidades de la mayoría de los usuarios como se puede comprobar en la figura [7.6](#).

Nivel de satisfacción con la pantalla perfil

18 respuestas

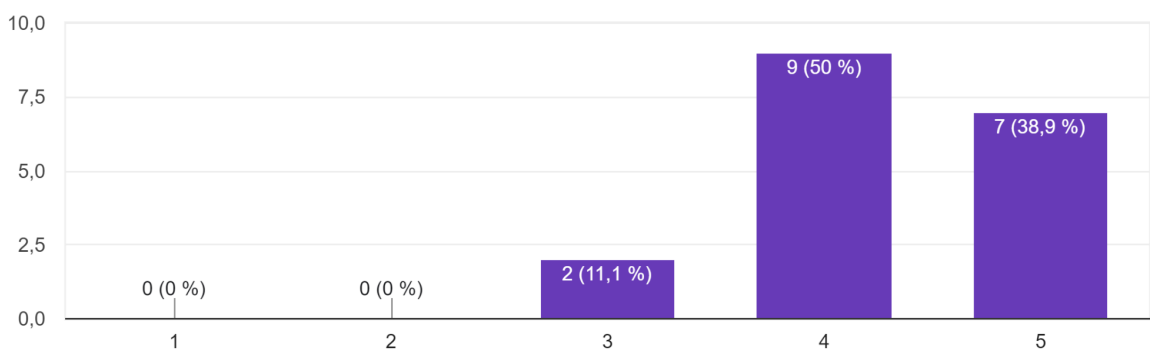


Figura 7.6: Nivel de satisfacción con la pantalla perfil

Nivel de satisfacción con la pantalla de ventas.

En general, la alta satisfacción de los usuarios con la pantalla de ventas es una buena señal de que la pantalla está funcionando satisfactoriamente. Sería importante seguir escuchando las opiniones de los usuarios y trabajar en mejoras, pero la alta satisfacción indica que la mayoría de los usuarios están contentos con la pantalla y que la aplicación en general está funcionando bien en lo que respecta a la presentación de información de ventas, figura [7.7](#).

Nivel de satisfacción con la pantalla de ventas

18 respuestas

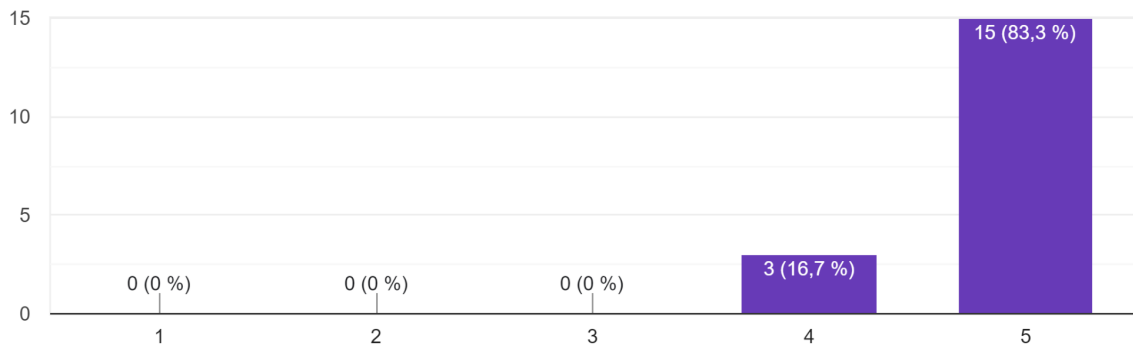


Figura 7.7: Nivel de satisfacción con la pantalla de ventas

Nivel de satisfacción con la realización de encuestas.

En general, la alta satisfacción de los usuarios con la realización de encuestas es una buena señal de que se ha hecho un buen trabajo con la calidad de las encuestas. Mirar en la figura [7.8](#).

Nivel de satisfacción con la realización de encuestas

18 respuestas

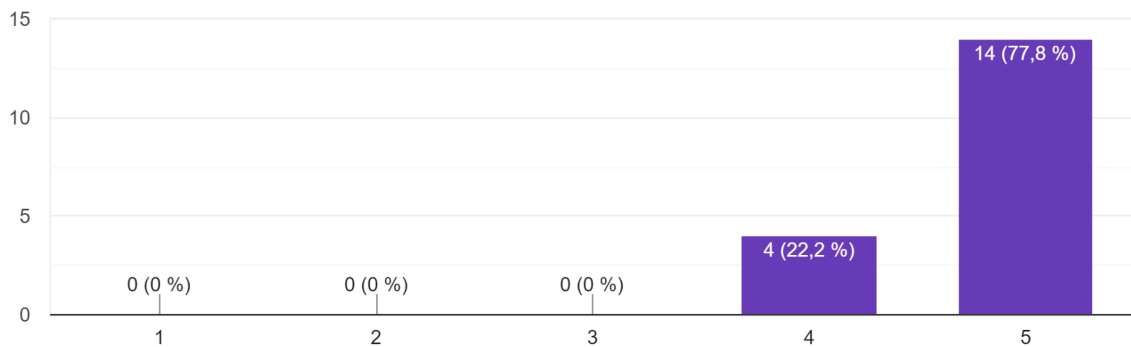


Figura 7.8: Nivel de satisfacción con la realización de encuestas

Nivel de satisfacción general con la aplicación.

La mayoría de los usuarios están muy satisfechos con la aplicación en general, y que la aplicación cumple con sus expectativas como se puede ver en los resultados de la figura [7.9](#).

Es importante tener en cuenta que, aunque algunos usuarios puedan haber calificado la aplicación con un 4, esto no significa necesariamente que no estén satisfechos con ella. Es posible que algunos usuarios tengan preferencias o necesidades específicas que no estén siendo completamente satisfechas, pero esto no significa que la aplicación no sea efectiva o útil para la mayoría de los usuarios.

En general, la alta satisfacción de los usuarios con la aplicación es una buena señal de que se ha realizado un buen trabajo de diseño y desarrollo de la aplicación. Sería importante seguir escuchando las opiniones de los usuarios y trabajar en mejorarla, pero la alta satisfacción indica que la mayoría de los usuarios están contentos con la aplicación y que esta está cumpliendo con su propósito de manera efectiva.

Nivel de satisfacción general con la aplicación

18 respuestas

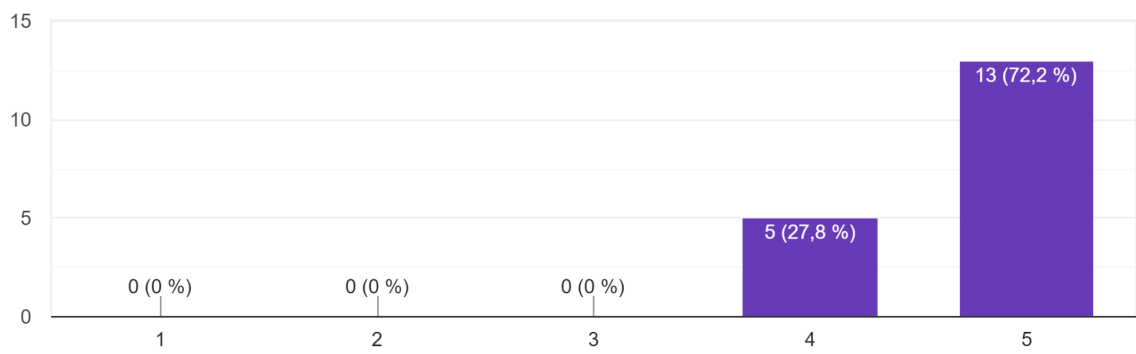


Figura 7.9: Nivel de satisfacción general con la aplicación

Capítulo 8

Conclusiones y trabajo futuro

8.1. Conclusiones

En este proyecto se ha desarrollado un sistema orientado a facilitar a los propietarios de tiendas físicas la predicción de futuras ventas y control de stocks de sus tiendas, para poder aumentar la rentabilidad de sus negocios y poder hacerse una idea de cómo está siendo la progresión de las ventas de su tienda. Gracias a esto y poder anticipar las compras necesarias que tendrán que realizar, aminorarán gastos en el negocio haciendo de cada mes contable una ventaja competitiva frente a los propietarios de otros negocios que no hagan uso de esta aplicación. Los vendedores podrán saber cuáles son los productos que más interesan a sus clientes, aquellos que son más visitados o aquellos que son mejor valorados, a parte del sistema de creación de encuestas que ofrece la aplicación con la que unívocamente podrá conocer la opinión de los clientes acerca de productos o acerca de su satisfacción en general con la marca.

Todos los ficheros fuente del proyecto se encuentran alojados en el siguiente repositorio de GitHub:

<https://github.com/Jomedi/Prediccion-de-ventas>

8.2. Trabajo futuro

A pesar de que todos los objetivos y funcionalidades planteados durante la fase de toma de requisitos han sido alcanzados y desarrollados, se han planteado, durante la realización del proyecto, nuevas funcionalidades que podrían hacer una aplicación más completa en caso de realizar trabajo futuro:

- **Más sistemas de predicción.** El desarrollo de nuevos sistemas de predicción permitiría que la aplicación se adapte mejor a las necesidades específicas de los vendedores. Por ejemplo, podrían implementarse sistemas de predicción personalizables por producto, que permitan a los usuarios obtener pronósticos más precisos para cada uno de los artículos que venden. También se podrían implementar otros sistemas de predicción de ventas basados en diferentes técnicas de aprendizaje automático más duraderas en el tiempo, lo que aumentaría la variedad de herramientas disponibles para los vendedores y mejoraría la calidad de las predicciones.
- **Tienda online.** La implementación de una tienda online integrada en la aplicación permitiría a los vendedores vender sus productos directamente a través de ella, lo que mejoraría la experiencia del usuario y aumentaría la eficiencia en la gestión de ventas. Esto también podría aumentar la cantidad de clientes de la aplicación, ya que muchos vendedores podrían estar interesados en tener su propia tienda en línea y muchos más clientes de estos vendedores podrían comprar sin necesidad de ir a la tienda física, lo que mejoraría a su vez la calidad en la predicción de las ventas de la aplicación.
- **Ofertas con la realización de encuestas.** La posibilidad de crear encuestas dentro de la aplicación y ofrecer ofertas especiales a los usuarios que las completen, podría mejorar la fidelización de los clientes y aumentar la interacción de los clientes con la aplicación. Al hacerlo, se podría conocer mejor a los clientes y ofrecerles productos y servicios más adecuados a sus necesidades y poder apostar de una forma más precisa por productos ganadores.
- **Encuestas de satisfacción automáticas.** La implementación de encuestas de satisfacción automáticas permitiría a los usuarios obtener una retroalimentación constante sobre la calidad del servicio que ofrecen y la experiencia del cliente sin

necesidad de que el usuario administrador las cree y las mande a los clientes. Esto podría mejorar la calidad de los productos y servicios que ofrece la tienda, así como aumentar la satisfacción del cliente y la retención de los mismos. Además, la información recopilada a través de estas encuestas podría ser utilizada para mejorar la aplicación y su funcionalidad en el futuro.

Conclusions and future work

Conclusions

In this project, a system has been developed aimed at making it easier for physical store owners to predict future sales and stock control in their stores, in order to increase the profitability of their businesses and to get an idea of how the progression of their sales is being. your store sales. Thanks to this and being able to anticipate the necessary purchases that they will have to make, they will reduce expenses in the business, making each accounting month a competitive advantage over the owners of other businesses that do not use this application. Sellers will be able to find out which products are of most interest to their customers, those that are most visited or those that are best valued, apart from the system for creating surveys offered by the application, with which they can unequivocally know the opinion of customers. about products or about their overall satisfaction with the brand.

All the project source files are hosted in the following GitHub repository:

<https://github.com/Jomedi/Prediccion-de-ventas>

Future work

Despite the fact that all the objectives and functionalities proposed during the requirements gathering phase have been achieved and developed, new functionalities have been proposed during the project, which could make a more complete application in case of carrying out future work:

- **More prediction systems.** The development of new prediction systems would allow the application to be better adapted to the specific needs of sellers. For example, customizable forecasting systems could be implemented by product, allowing users to obtain more accurate forecasts for each of the items they sell. Other sales prediction systems based on different machine learning techniques that are more durable over

time could also be implemented, which would increase the variety of tools available to sellers and improve the quality of predictions.

- **Online store.** The implementation of an online store integrated into the application would allow sellers to sell their products directly through it, which would improve the user experience and increase the efficiency of sales management. This could also increase the number of customers of the application, since many sellers might be interested in having their own online store and many more customers of these sellers could buy without going to the physical store, which would in turn improve the quality in the prediction of the sales of the application.
- **Offers with the completion of surveys.** The ability to create in-app surveys and offer special offers to users who complete them could improve customer loyalty and increase customer interaction with the app. By doing so, you could get to know your customers better and offer them products and services more suited to their needs and be able to bet more precisely on winning products.
- **Automatic satisfaction surveys.** The implementation of automatic satisfaction surveys would allow users to obtain constant feedback on the quality of the service they offer and the customer experience without the need for the administrator user to create them and send them to customers. This could improve the quality of the products and services offered by the store, as well as increase customer satisfaction and customer retention. In addition, the information collected through these surveys could be used to improve the application and its functionality in the future.

Bibliografía

- [1] Predicción de ventas: https://es.wikipedia.org/wiki/Pron%C3%B3stico_de_venta#:~:text=Previsi%C3%B3n%20de%20ventas%20o%20Demanda,y%20otros%20elementos%20de%20negocios.
- [2] QR: https://es.wikipedia.org/wiki/C%C3%B3digo_QR
- [3] Angular: <https://angular.io/>
- [4] Web: https://es.wikipedia.org/wiki/P%C3%A1gina_web
- [5] iOS: <https://es.wikipedia.org/wiki/IOS>
- [6] Android: <https://es.wikipedia.org/wiki/Android>
- [7] GitHub: <https://github.com/>
- [8] Tableau: <https://www.tableau.com/>
- [9] Salesforce: <https://www.salesforce.com/es/>
- [10] Microsoft Dynamics 365: <https://dynamics.microsoft.com/es-es/>
- [11] Oracle sales Cloud: <https://www.oracle.com/cx/sales/>
- [12] Código abierto: https://es.wikipedia.org/wiki/C%C3%B3digo_abierto
- [13] Javascript: <https://es.wikipedia.org/wiki/JavaScript>
- [14] Typescript: <https://es.wikipedia.org/wiki/TypeScript>
- [15] HTML5: <https://es.wikipedia.org/wiki/HTML5>
- [16] CSS: <https://developer.mozilla.org/es/docs/Web/CSS>
- [17] Bootstrap: <https://getbootstrap.com/docs/5.0/getting-started/introduction/>
- [18] Firebase: <https://firebase.google.com/>
- [19] Firebase Realtime Database: <https://firebase.google.com/docs/database?hl=es-419>

- [20] Firebase Authentication: <https://firebase.google.com/docs/auth?hl=es-419>
- [21] Firebase Storage: <https://firebase.google.com/docs/storage?hl=es-419>
- [22] Capacitor: <https://capacitorjs.com/solution/angular>
- [23] Ngx-scanner-qrcode: <https://www.npmjs.com/package/ngx-scanner-qrcode>
- [24] Highcharts: <https://www.highcharts.com/blog/tutorials/highcharts-with-angular-v14/>
- [25] Visual Studio Code: <https://code.visualstudio.com/>
- [26] Trello: <https://trello.com/>
- [27] JSON: <https://es.wikipedia.org/wiki/JSON>
- [28] Amazon Seller: <https://sellercentral.amazon.es/>

Anexo A

Guía de uso

En este anexo se explica cómo hacer uso de la aplicación de gestión de contenido para el administrador y posteriormente se explica cómo usar la aplicación móvil para el resto de usuarios.

A.1. Vista principal

A.1.1 Usuario

En la vista principal se observa la primera página “Home” que observará el usuario después de hacer login, donde para todos los usuarios, aparecerá el botón verde de Lectura de QR que se muestra en la figura [A.1](#), con el que podrá escanear códigos QR de los productos de la tienda física. Una vez clicado le saltará una ventana, mostrada en la figura [A.2](#), en la que solamente tendrá que pulsar el botón Start para iniciar el escaneo QR.



Figura A.1: Lector QR

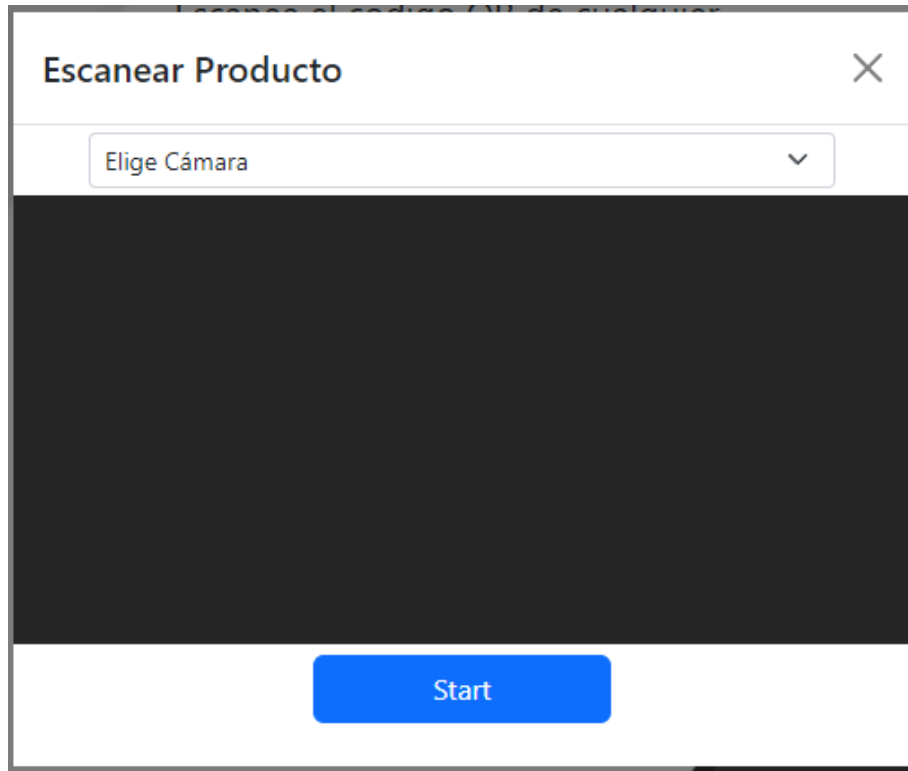


Figura A.2: Escanear producto

A.1.2 Administrador

En el caso del usuario administrador, aparecerá más abajo también el botón rojo de predicción de ventas, figura [A.3](#), con el que haciendo clic, podrá observar la predicción de ventas de los productos vendidos de la tienda. Clicando dicho botón aparecerá el gráfico como en la figura [A.4](#), en el que podrá seleccionar el Modo, figura [A.5](#), con el botón desplegable que aparece encima.

Predicción de Ventas

Obtén una predicción de ventas basada en la tendencia de ventas del año actual y anterior por cada temporada



Figura A.3: Predicción de ventas

Dashboards

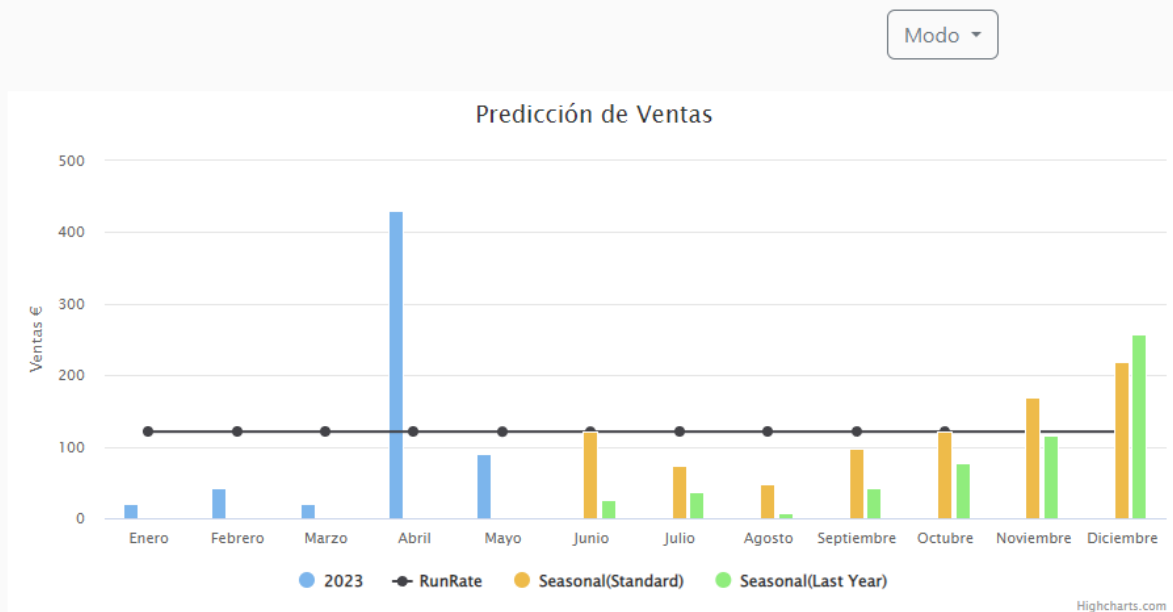


Figura A.4: Gráfico de ventas

Dashboards

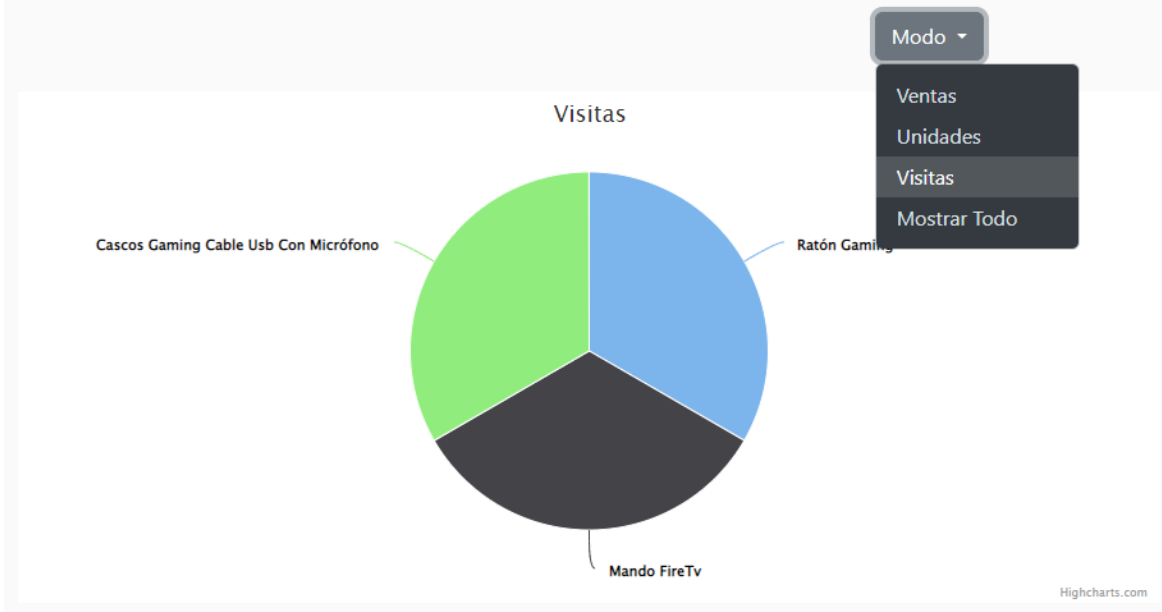


Figura A.5: Modo de gráfico

A.2. Catálogo

A.2.1. Usuario

En la pantalla catálogo es donde encontramos todos los productos tal y como se muestra en la figura [A.6](#). Se podrán buscar los productos deseados a través de la barra de búsqueda que aparece en la parte superior en el header de la página que se puede ver en la figura [A.7](#).

Si queremos añadir un producto a favoritos, haremos clic en el icono del corazón del producto que nos haya interesado, podremos ver todos los productos favoritos entrando en “Mis favoritos”. Si lo que se desea es obtener información más detallada del producto, lo que se deberá hacer será clicar la imagen del producto o clicar el icono de información que aparece en la tarjeta de cada producto. Con ello se accederá a la página de detalles del producto mostrada en la figura [A.8](#).

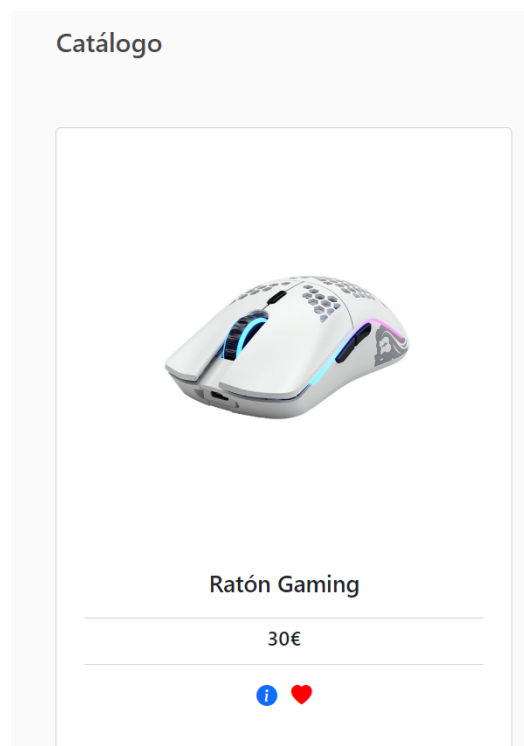


Figura A.6: Producto

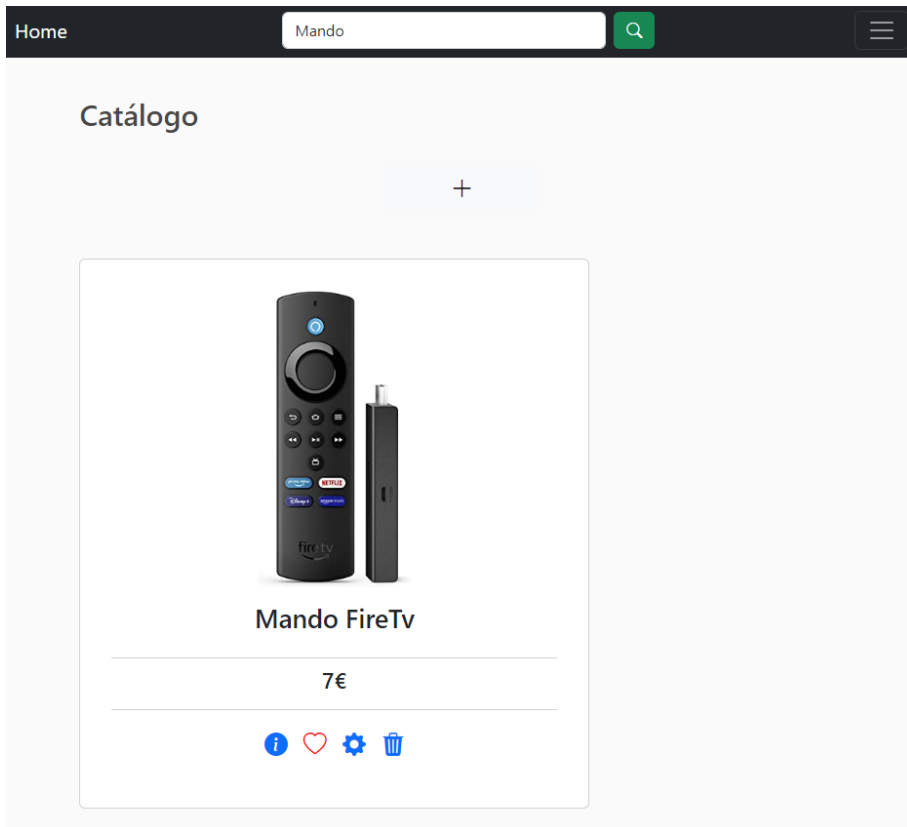



Figura A.7: Mando FireTv

SKU: -NO4XocHzOjls_qBmE2F


Ratón Gaming

★★★★☆

30€



Durante más de una década, la línea razer deathadder ha sido un pilar en el ámbito mundial de los deportes electrónicos. Se ha ganado una reputación de confiabilidad que los jugadores juran debido a su durabilidad y ergonomía probadas. Ahora, lo estamos haciendo aún más accesible con su último sucesor: el razer deathadder essential.

1  Registrar Venta




Figura A.8: Detalles del producto

En la página de detalles del producto aparecerá el título, la valoración media y una descripción del producto. También se podrá obtener el código QR del producto clicando en el botón verde con icono de QR, como aparece en la figura [A.9](#) y registrar compras que el propio usuario haya hecho de ese producto mostrado en la figura [A.10](#).

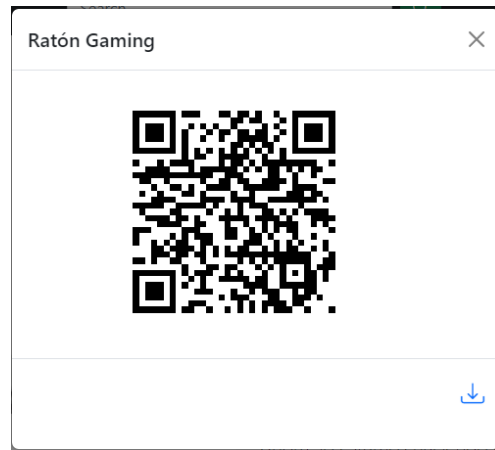


Figura A.9: Código QR de producto

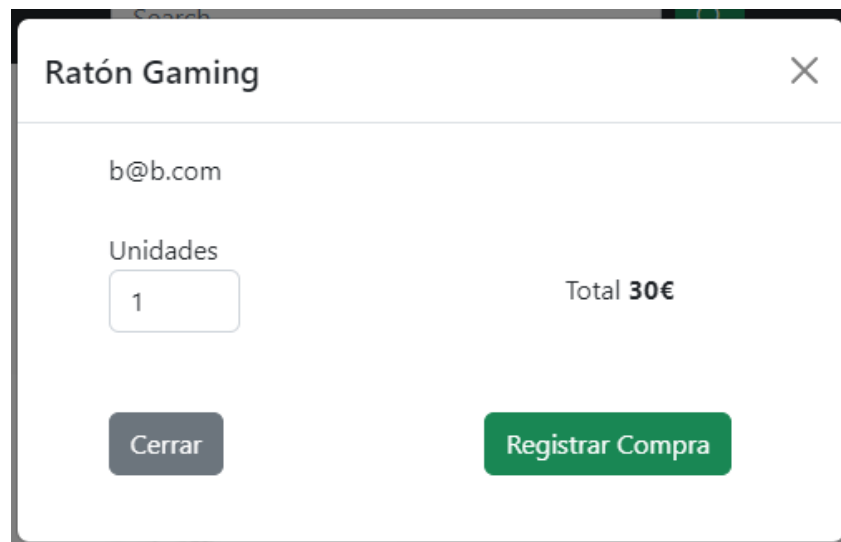


Figura A.10: Registrar Compra

En la pantalla de detalles también se podrán observar valoraciones que los usuarios han puesto al producto, o añadir una valoración haciendo uso del botón “Añadir Valoración” que aparece en la figura [A.11](#). Entonces nos aparecerá el modal que podemos ver en la figura [A.12](#) con campos de texto que debemos rellenar y pulsar en el botón de “Publicar” para poder publicar nuestra valoración en la página de detalles del producto. También podremos modificar o eliminar los comentarios que hayamos subido con la cuenta con la que estemos registrados, en este caso estamos registrados con la cuenta “b@b.com”.

Para modificar el producto haremos clic en el icono de modificar para que nos salga el modal de modificación de valoración de la figura [A.13](#).

Para eliminarlo haremos clic en el icono de eliminar y aparecerá el modal de la figura [A.14](#).



Figura A.11: Valoraciones

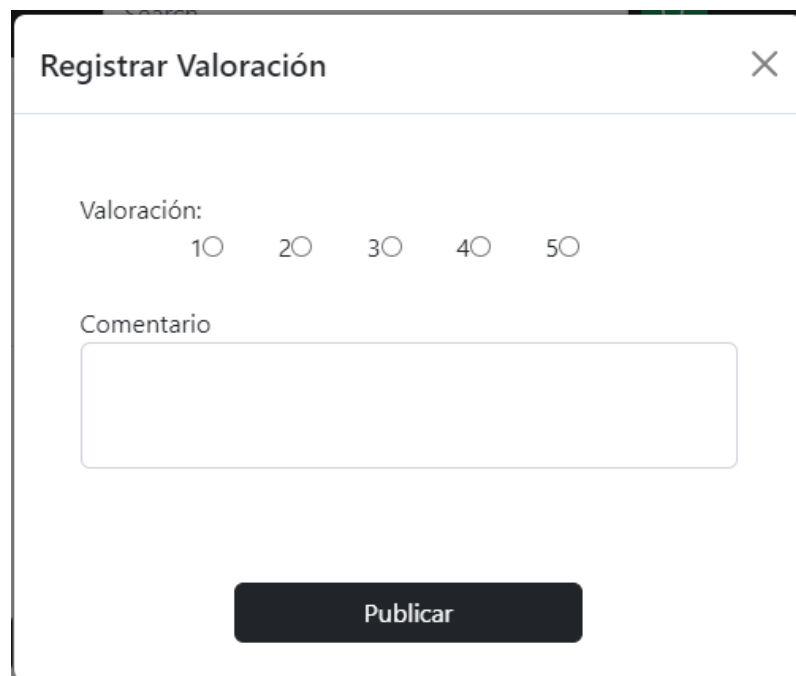


Figura A.12: Registrar Valoración

Modificar Valoración ✕

b@b.com
★★★★☆
fesaefsdcdxf

Valoración:

1○ 2○ 3○ 4○ 5○

Comentario

fesaefsdcdxf

Modificar

Cerrar

Figura A.13: Modificar Valoración

¿Estás seguro de que quieres eliminar la valoración? ✕

b@b.com
★★★★☆
fesaefsdcdxf

Cancelar **Eliminar**

Figura A.14: Eliminar Valoración

A.2.2. Administrador

Como usuario administrador además de tener acceso a todas las funcionalidades de la cuenta de usuario, se pueden modificar o eliminar productos del catálogo, haciendo clic en los botones extra de configuración o de eliminación que aparecen en la tarjeta de cada producto cuando se ha accedido con la cuenta de administrador mostrada en la figura [A.15](#).

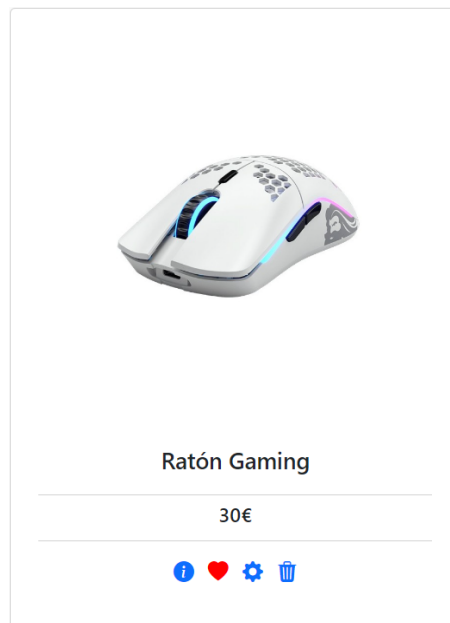


Figura A.15: Producto con cuenta de administrador

Aparte, en la página de detalles del producto, el usuario administrador podrá registrar ventas asociadas a cualquier usuario de la aplicación como muestra la figura [A.16](#) o modificar o eliminar las valoraciones de productos de cualquier usuario figura [A.17](#).

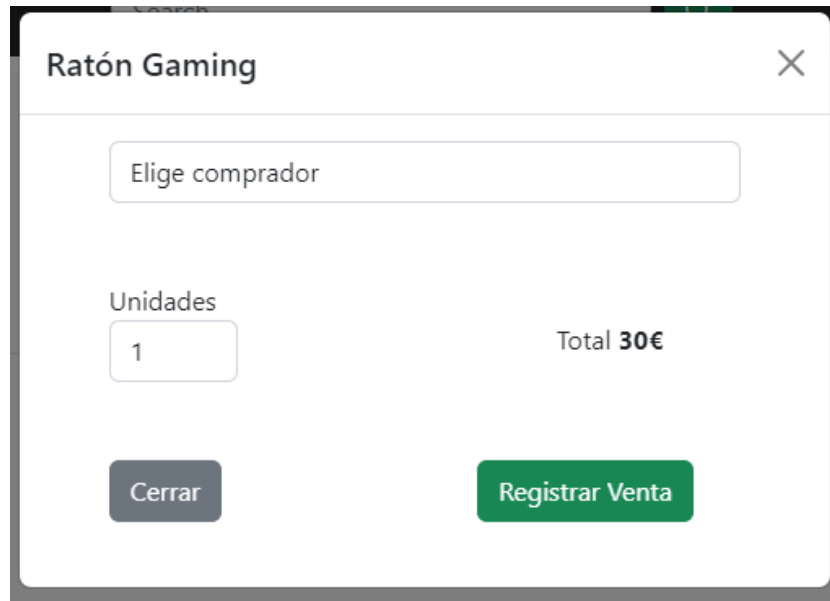


Figura A.16: Registrar venta de producto con cuenta de administrador

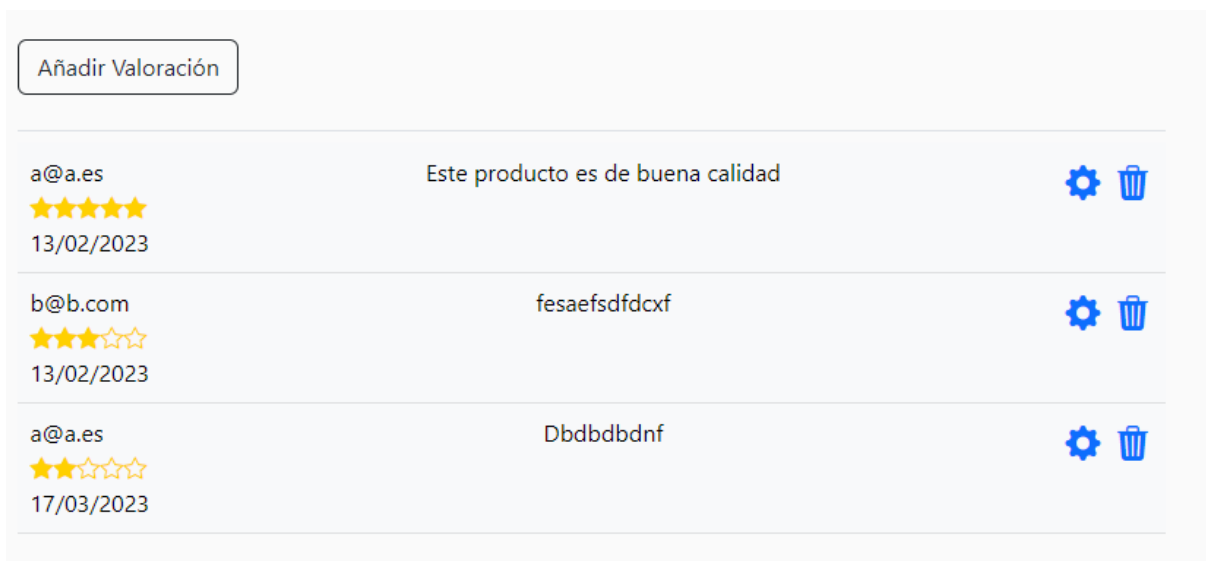


Figura A.17: Valoraciones con cuenta de administrador

A.3. Ventas

A.3.1. Usuario

Como usuario se podrá acceder a “Consultar Compras” desde la barra de navegación, donde se podrán consultar todas las compras asociadas a la propia cuenta de usuario como se muestra en la figura [A.18](#).





The screenshot shows a web interface titled "Productos comprados". At the top left, there is a dropdown menu labeled "Año". Below it is a table with the following data:

Fecha de compra	Título del producto	Precio	Unidades	Total
24/01/2022	Ratón Gaming	30€	1	30€
27/01/2022	Cascos Gaming Cable Usb Con Micrófono	35€	2	70€
27/02/2022	Ratón Gaming	30€	1	30€
27/02/2022	Teclado	32€	1	32€
27/03/2022	Mando FireTv	7€	1	7€
27/04/2022	Cascos Gaming Cable Usb Con Micrófono	35€	1	35€
28/06/2022	Mando FireTv	7€	3	21€

Figura A.18: Productos comprados

A.3.2. Administrador

Accediendo a través de la barra de navegación a “Registro de Ventas” con la cuenta del usuario administrador, se podrá ver un registro de todas las ventas como se puede ver en la figura [A.19](#). Dentro de esta pantalla se podrá filtrar por año y mes o hacer clic en el botón de “Añadir Venta” el cual permitirá a través del modal de la figura [A.20](#), añadir una nueva venta asignando un email, un producto y una cantidad.

Home  

Ventas

[Añadir Venta](#)

Año ▾









Comprador	Fecha de compra	Título del producto	Unidades	Total	Modificar	Eliminar
b@b.com	24/01/2022	Ratón Gaming	1	30€		
b@b.com	27/01/2022	Cascos Gaming Cable Usb Con Micrófono	2	70€		
c@c.com	27/01/2022	Ratón Gaming	6	180€		
a@a.es	27/02/2022	Ratón Gaming	1	30€		

Figura A.19: Ventas como usuario administrador

Registro de nueva venta ✕

Elige comprador

Elige producto

Unidades

Figura A.20: Registrar nueva venta como usuario administrador

A.4. Encuestas

A.4.1. Usuario

Como usuario de la aplicación cuando se publica una encuesta nueva, aparecerá el aviso de nueva encuesta que aparece en la figura [A.21](#).

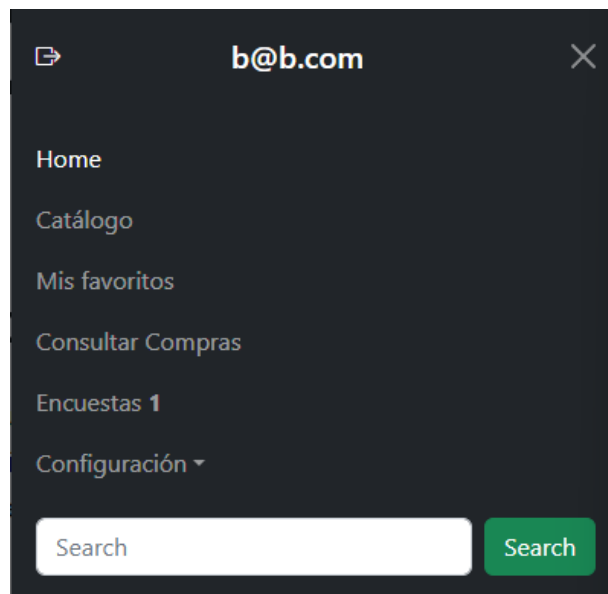
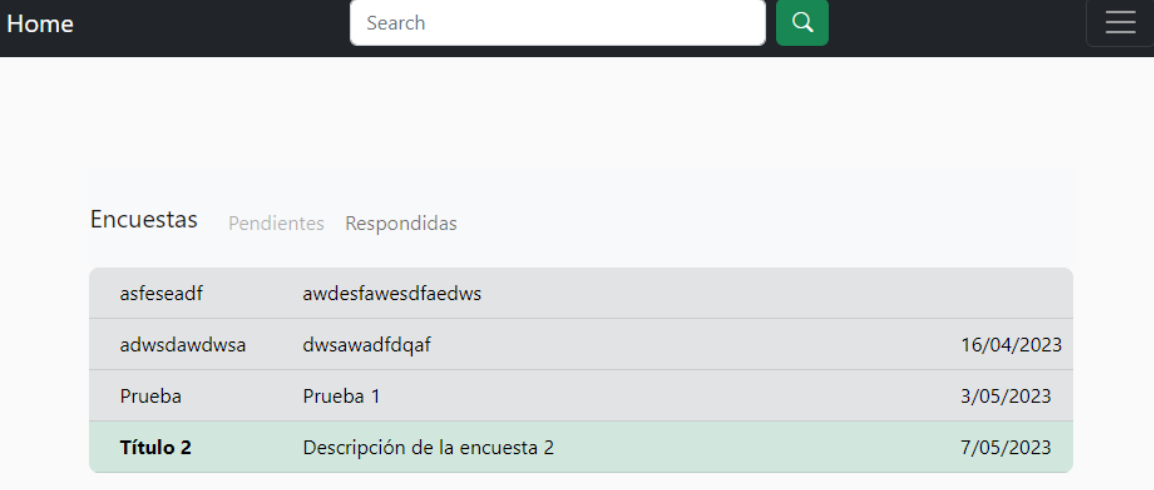


Figura A.21: Aviso de Encuesta

Accediendo a sus encuestas, figura [A.22](#), encontrará una nueva encuesta, la cual aparecerá en un color diferente por no haberla abierto aún. El usuario podrá clicar en la encuesta, responder las preguntas de la encuesta y enviarla, como se puede ver en el ejemplo de la figura [A.23](#).



Encuestas		
Pendientes		
Respondidas		
asfeseadf	awdesfawesdfaedws	
adwsdawdwsa	dwsawadfdqaf	16/04/2023
Prueba	Prueba 1	3/05/2023
Título 2	Descripción de la encuesta 2	7/05/2023

Figura A.22: Encuestas pendientes

The image shows a mobile application window for responding to a survey. The window has a title bar with the text 'Título de la encuesta' and a close button (X) in the top right corner. Below the title bar, the text 'Descripción de la encuesta' is displayed. The form contains three question types: 1. 'Pregunta de tipo texto' with a single-line text input field. 2. 'Pregunta de tipo valoración' with a horizontal slider control featuring a blue dot in the center. 3. 'Pregunta de tipo multivalor' with a multi-select dropdown menu. At the bottom right of the form, there are two buttons: a dark grey 'Cerrar' button and a green 'Enviar' button.

Figura A.23: Responder encuesta

A.4.1. Administrador

La página encuestas le permite al usuario administrador crear nuevas encuestas pulsando en el botón de “Nueva Encuesta”, como muestra la figura [A.24](#). Se desplegará una ventana en donde se pedirá añadir el título y descripción de la nueva encuesta.

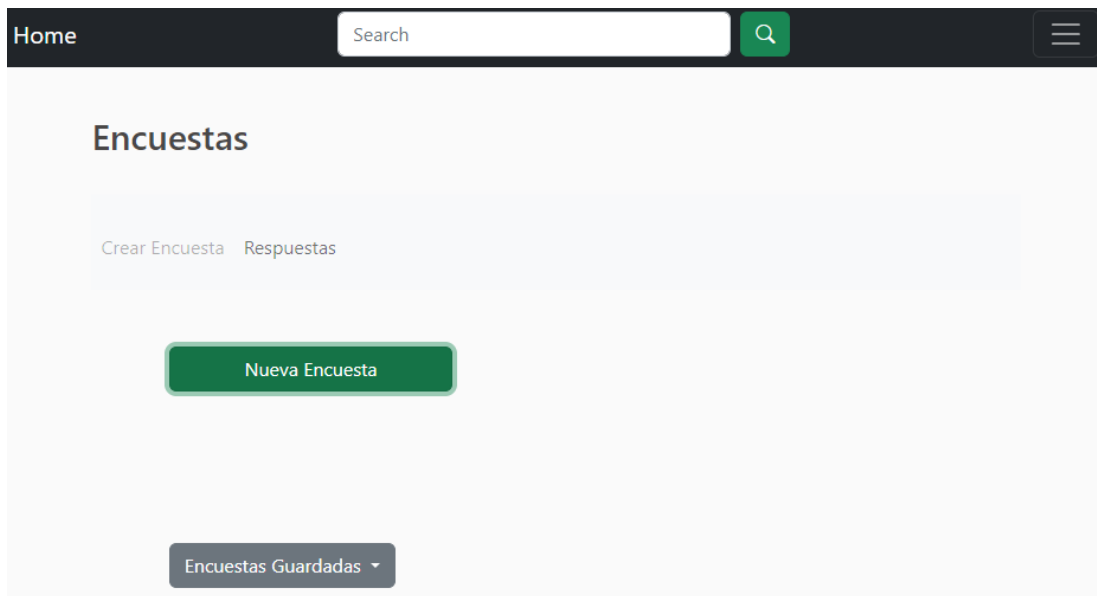


Figura A.24: Encuestas

Una vez creada la encuesta se podrán añadir campos con diferentes tipos de preguntas y añadir el título de cada una de las preguntas como muestra la figura [A.25](#).

Home Search

Encuestas

Crear Encuesta Respuestas

Título

Descripción de la encuesta

Título de la pregunta 1

Añadir Campo

Guardar Encuesta Publicar Encuesta

Figura A.25: Nueva encuesta

Una vez terminada la encuesta, se podrá publicar al resto de usuarios, presionando en el botón Publicar Encuesta y eligiendo al grupo de usuarios al que se le quiere enviar la encuesta a través del modal que aparece en la figura [A.26](#).

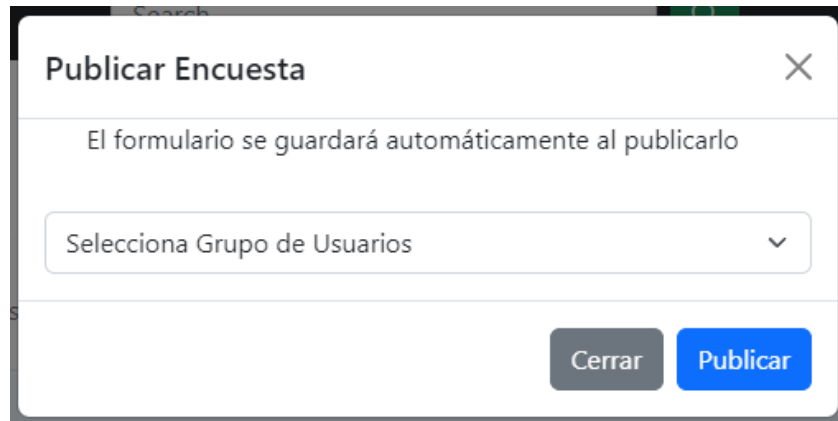


Figura A.26: Publicar Encuesta

El usuario administrador también podrá acceder a las respuestas de las encuestas realizadas, utilizando el botón de la barra de navegación superior “Respuestas” como se puede observar en la figura [A.27](#).

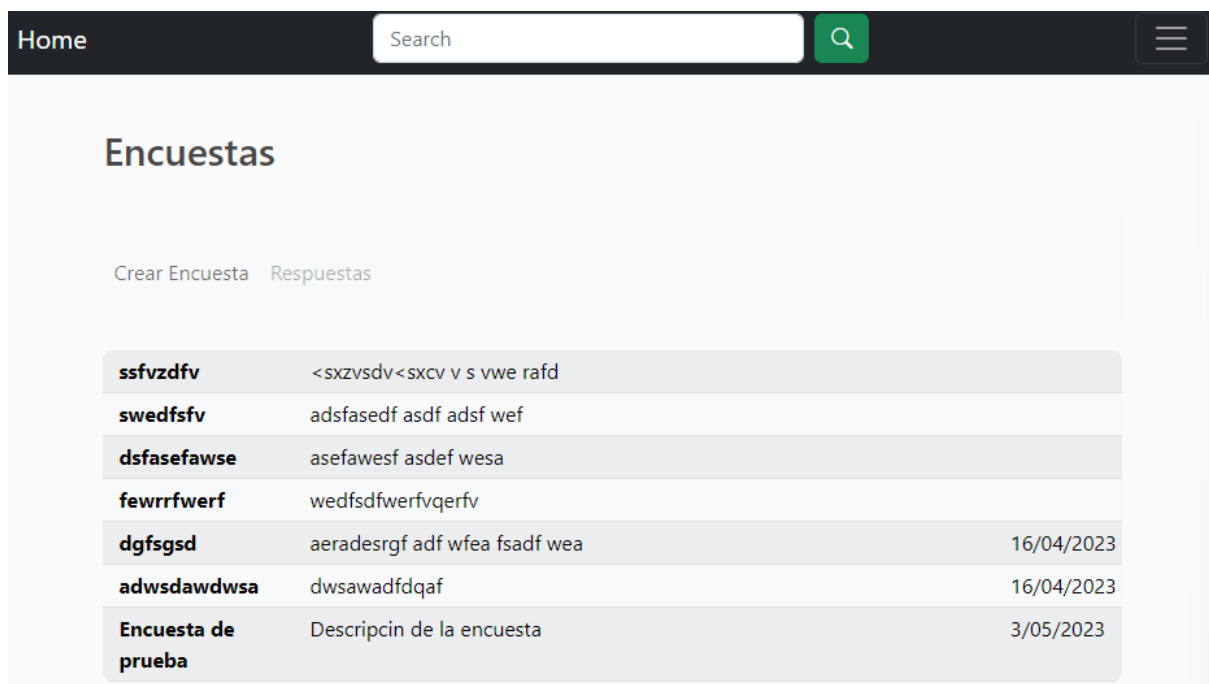
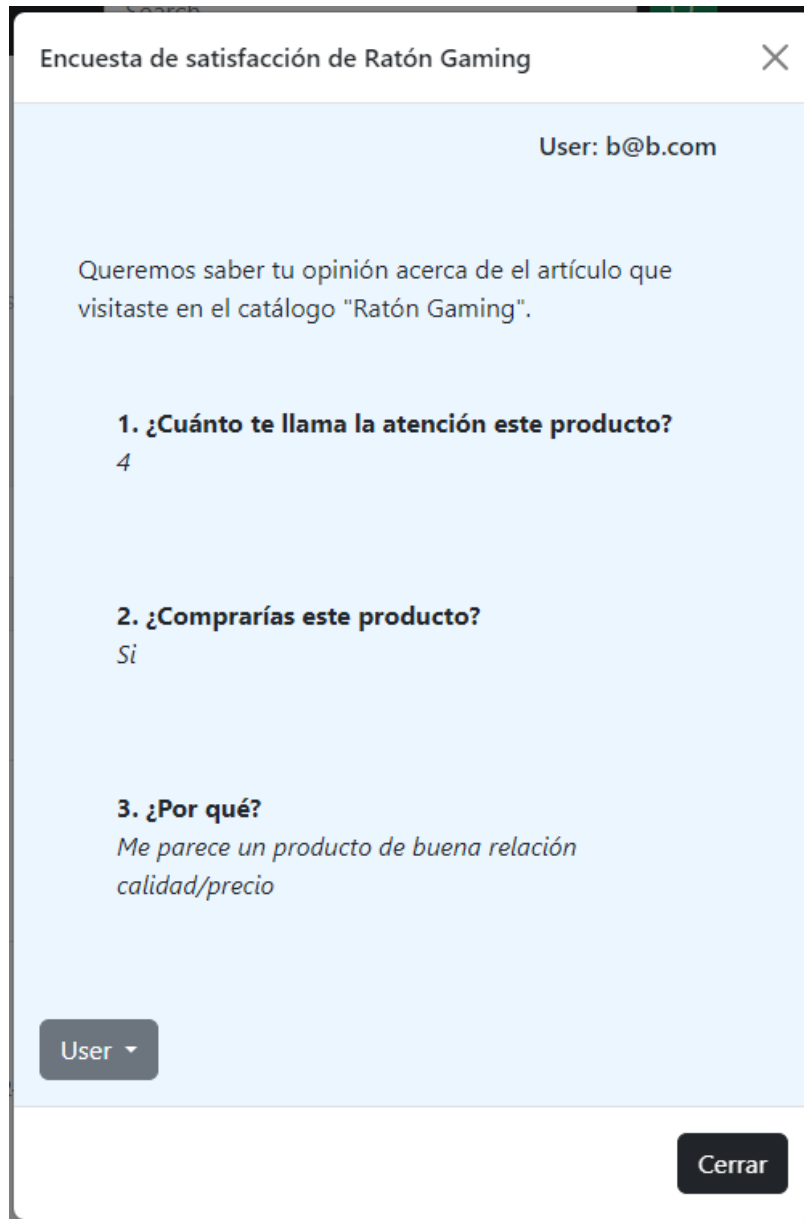


Figura A.27: Respuestas

Haciendo clic en cada una de las filas de la tabla accederemos a las respuestas que han dado cada uno de los usuarios en esa encuesta en concreto como aparece en la figura [A.28](#).



The image shows a modal window titled "Encuesta de satisfacción de Ratón Gaming" with a close button (X) in the top right corner. The window has a light blue background. At the top right, it displays "User: b@b.com". Below this, a message reads: "Queremos saber tu opinión acerca de el artículo que visitaste en el catálogo 'Ratón Gaming'". There are three survey questions:

- 1. ¿Cuánto te llama la atención este producto?**
4
- 2. ¿Comprarías este producto?**
Si
- 3. ¿Por qué?**
Me parece un producto de buena relación calidad/precio

At the bottom left, there is a dark grey button labeled "User" with a downward arrow. At the bottom right, there is a dark grey button labeled "Cerrar".

Figura A.28: Respuestas de usuario

A.5. Perfil

A.5.1. Usuario

Todos los usuarios podrán modificar su perfil en la pantalla Perfil. Harán clic en el botón Modificar Perfil [A.29](#), al clicar los campos de texto se volverán editables como en la figura [A.30](#). También podrán borrar su cuenta con el botón Borrar Cuenta [A.31](#).

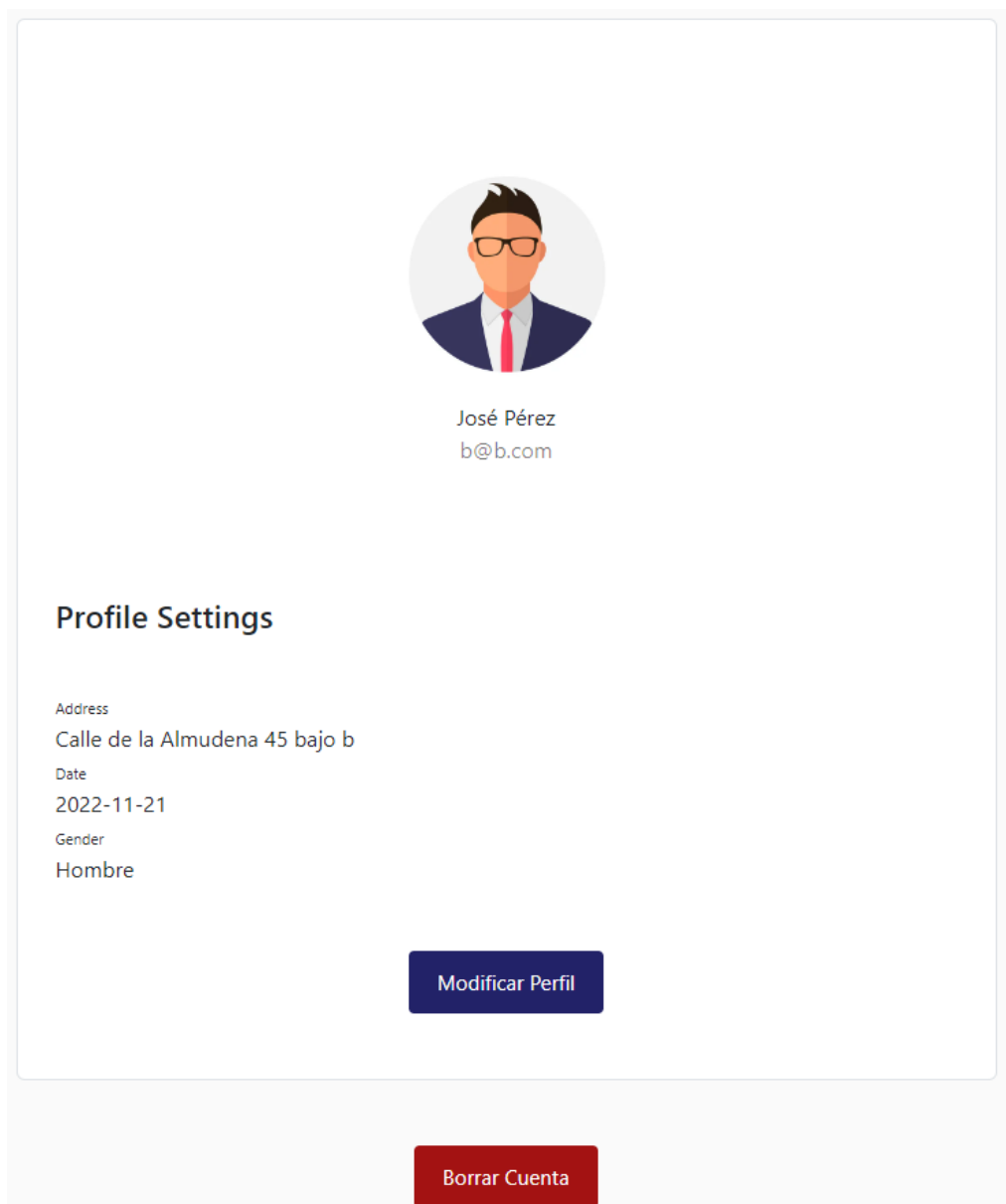



Figura A.29: Perfil




Andrés Pajares
b@b.com

Profile Settings

Name

Address

Date

Gender

[Guardar Perfil](#)

Figura A.30: Modificar Perfil

[Borrar Cuenta](#)

Figura A.31: Borrar cuenta

A.6.1. Administrador

El usuario administrador además del enlace en la barra de navegación a modificar su propia cuenta, tendrá un enlace a “Administrar Cuentas” como aparece en la figura [A.32](#).

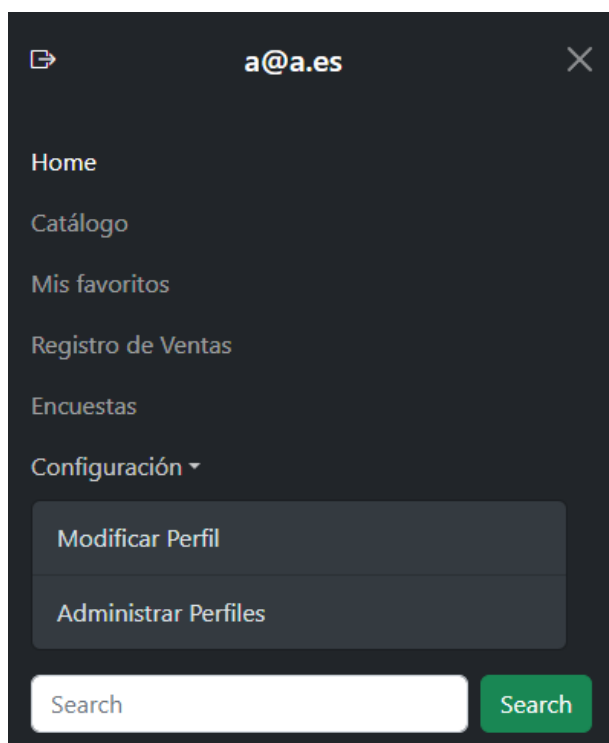


Figura A.32: Administrar Cuentas


Una vez habiendo accedido a administrar perfiles con la cuenta de administrador, figura [A.33](#), se podrán ver todos los perfiles registrados en la aplicación, con las opciones de modificar o eliminar cuentas de usuario. Si se desean cualquiera de estas opciones, se deberá pulsar cualquiera de los iconos de las acciones en la fila del usuario donde se desee aplicar la acción. Con el icono de modificación, saltará el modal de la figura [A.34](#), o si se desea se podrá eliminar el perfil que aparece en la misma fila que cada usuario de la cuenta clicando en el botón de la papelera asociado a cada cuenta de usuario.

Administrar Perfiles

E-mail	Fecha	Acciones
a@a.es	2022-11-04	 
b@b.com	2022-11-21	 
c@c.com	2022-11-10	 

Figura A.33: Administrar Perfiles

Profile Settings




Name: Daniela Valera

Email: c@c.com

Name

Address

Date 

Gender

Figura A.34: Modificar Perfil

AUTORIZACIÓN PARA LA DIFUSIÓN DEL TRABAJO FIN DE GRADO Y SU DEPÓSITO EN EL REPOSITORIO INSTITUCIONAL E-PRINTS COMPLUTENSE

Los abajo firmantes, alumno/a y tutor/a del Trabajo Fin de Grado (TFG) de la Facultad de Informática, autorizan a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor el Trabajo Fin de Grado (TFG) cuyos datos se detallan a continuación. Así mismo, autorizan a la Universidad Complutense de Madrid a que sea depositado en acceso abierto en el repositorio institucional con el objeto de incrementar la difusión, uso e impacto del TFG en Internet y garantizar su preservación y acceso a largo plazo.

Periodo de embargo (opcional): 6 meses 12 meses

Título del TFG: **Desarrollo de un sistema de predicción de ventas en tiendas físicas**
Curso académico: 22/23

Alumno/a
Jorge Medina Carretero

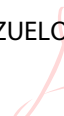
Tutor/a y su departamento
Antonio Sarasa Cabezuelo (DSIC)

Fecha y firma alumno/a



Fecha y firma tutor/a

SARASA CABEZUELO
ANTONIO -
51064599E



Firmado digitalmente por SARASA CABEZUELO
ANTONIO - 51064599E
Nombre de reconocimiento (DN): c=ES,
serialNumber=IDCES-51064599E,
givenName=ANTONIO, sn=SARASA
CABEZUELO, cn=SARASA CABEZUELO ANTONIO
- 51064599E
Fecha: 2023.05.25 20:18:06 +02'00'