

**Desarrollo de una aplicación para  
planificar visitas temáticas al Botánico  
de Madrid**

**Development of an application to  
schedule thematic visits to the Jardín  
Botánico de Madrid**

**Autores**

**Francisco García Navarrete  
Fabricio Bryan Torres Samueza**

**Director**

**Antonio Sarasa Cabezuelo**

**Curso 2020-2021**

**Trabajo de fin de grado del grado en  
ingeniería informática  
Facultad de informática  
Universidad Complutense de Madrid**





## Agradecimientos

La primera persona a la que le queremos agradecer es a nuestro tutor, Antonio Sarasa Cabezuelo, por guiarnos y apoyarnos a lo largo del desarrollo de este proyecto y también por sus consejos y ánimos, estando siempre disponible para nosotros cuando lo hemos necesitado.

Del mismo modo nos gustaría agradecer a María Bellet Serrano y a todo el personal del Real Jardín Botánico de Madrid que nos ha proporcionado información, feedback y demás para ayudarnos en el desarrollo de la aplicación.

También queremos darle las gracias a nuestras familias ya que de no ser por ellos no habríamos podido embarcarnos en la aventura que es esta carrera y también por todo lo que nos han dado a lo largo de tantos años.

Por último, pero no por ello menos importante, muchas gracias a nuestros amigos y compañeros que han estado ayudándonos a lo largo de la carrera, así como a todas aquellas personas que en algún momento han tenido que aguantarnos.

## Resumen

Este trabajo surge con el objetivo de facilitar las visitas al Real Jardín Botánico de Madrid y también consultar la información sobre las plantas y árboles que contiene.

Actualmente el Real Jardín Botánico de Madrid ofrece un itinerario recomendado para realizar una visita al mismo. No obstante, este itinerario recomendado no tiene por qué satisfacer las expectativas de la persona que lo utilice. A su vez, la consulta de la información sobre las plantas y árboles del jardín no se puede realizar de manera cómoda ya que únicamente puede hacerse desde la propia página del Real Jardín Botánico. Debido a todas estas circunstancias, en este trabajo de fin de grado se ha desarrollado un sistema para compartir itinerarios creados por usuarios a modo de red social y para la consulta de la información anteriormente mencionada.

Para esto cada usuario puede consultar una lista de itinerarios creados y/o acceder a un menú de creación de los mismos. El sistema se encarga de manera automática de generar el camino a seguir en el itinerario en base al orden en el que se hayan añadido las parcelas al mismo. De la misma manera también pueden acceder a una lista en la que consultar la información sobre los elementos del jardín.

Para la implementación de este sistema se ha desarrollado una aplicación para dispositivos móviles Android e IOS junto a una API REST que permite gestionar la base de datos de la aplicación.

## Palabras clave

Real Jardín Botánico, aplicación móviles, itinerario, planta, API REST

## **Abstract**

This work comes with the objective to ease the visits to the Real Jardín Botánico de Madrid and also to consult the information regarding the plants and trees it has.

Currently the Real Jardín Botánica de Madrid offers a recommended itinerary to visit the garden. However, this recommended itinerario doesn't have to keep up with the person using it expectations. Moreover, the plant's and trees' information query can not be done in an easy way because it can only be accessed from the web page of the Real Jardín Botánico itself. Due to all this reasons, in this end-of-grade work a system to share itineraries made by users as a social media network and to query the previously said information has been developed.

To do so each user can consult a list of created itineraries and/or access a menu to create them. The system takes care automatically of generating the path to follow in the itinerary based on the selected squares of the garden. In the same way the users can also access a list where they can consult the information about the elements in the garden.

For the implementation of this system a mobile app for Android and IOS has been developed along an API REST that allows to manage the application's database.

## **KeyWords**

Real Jardín Botánico, mobile application, itinerary, plant, API REST

# Índice

<b>Resumen</b>	<b>4</b>
Palabras clave . . . . .	4
<b>Abstract</b>	<b>5</b>
KeyWords . . . . .	5
<b>Índice de figuras</b>	<b>11</b>
<b>Capítulo 1: Introducción</b>	<b>12</b>
1.1 Motivación . . . . .	12
1.2 Objetivos . . . . .	12
1.3 Estructura de la memoria . . . . .	12
<b>Chapter 1: Introduction</b>	<b>14</b>
1.1 Motivation . . . . .	14
1.2 Objectives . . . . .	14
1.3 Structure of the report . . . . .	14
<b>Capítulo 2: Estado del arte</b>	<b>16</b>
2.1 Sistemas de identificación plantas . . . . .	16
2.1.1 ArbolApp . . . . .	16
2.1.2 PlantNet . . . . .	16
2.2 Sistemas de muestra de detalles de plantas . . . . .	16
2.2.1 RJB Museo vivo . . . . .	17
2.2.2 iKnow Trees 2 LITE . . . . .	17
<b>Capítulo 3: Tecnología empleada</b>	<b>18</b>
3.1 Ionic . . . . .	18
3.2 Angular . . . . .	18
3.3 HTML5 y CSS3 . . . . .	18
3.4 Firebase y Firestore . . . . .	18
3.5 Leaflet . . . . .	19
3.6 Mapbox . . . . .	19
3.7 NodeJS . . . . .	19
3.8 NPM . . . . .	19
3.9 Visual Studio Code . . . . .	19
3.10 Git . . . . .	19
3.11 Gimp . . . . .	20
<b>Capítulo 4: Especificación de requisitos</b>	<b>21</b>
4.1 Actores . . . . .	21
4.2 Módulo información . . . . .	21
4.3 Módulo itinerarios . . . . .	23
4.4 Módulo usuario . . . . .	28
<b>Capítulo 5: Arquitectura de la aplicación y modelo de datos</b>	<b>36</b>
5.1 Arquitectura de la aplicación . . . . .	36
5.1.1 Elementos de la arquitectura . . . . .	36
5.1.1.1 Aplicación móvil . . . . .	36
5.1.1.2 Autenticación . . . . .	37
5.1.1.3 API REST . . . . .	37
5.1.1.4 Base de datos . . . . .	37
5.2 Modelo de datos . . . . .	38

<b>Capítulo 6: Diseño de la aplicación</b>	<b>41</b>
6.1 Colores y tipografía . . . . .	41
6.2. Funcionalidad de la aplicación . . . . .	41
6.2.1 Vista principal . . . . .	41
6.2.1.1 Home . . . . .	41
6.2.1.1 Menú lateral . . . . .	43
6.2.2 Vistas de usuario . . . . .	45
6.2.2.1 Autenticación . . . . .	45
6.2.2.2 Registrarse . . . . .	46
6.2.2.3 Cerrar sesión . . . . .	47
6.2.2.4 Mi perfil . . . . .	48
6.2.2.5 Cambiar nombre de usuario . . . . .	49
6.2.2.6 Cambiar contraseña . . . . .	50
6.2.2.7 Ver historial . . . . .	51
6.2.2.8 Ver favoritos del usuario . . . . .	51
6.2.2.9 Ver itinerarios creados por el usuario . . . . .	52
6.2.2.10 Eliminar cuenta . . . . .	53
6.2.3 Vistas de itinerarios . . . . .	54
6.2.3.1 Buscador de itinerarios . . . . .	54
6.2.3.2 Crear itinerario . . . . .	55
6.2.3.3 Ver itinerario . . . . .	56
6.2.3.4 Iniciar itinerario . . . . .	58
6.2.3.5 Añadir itinerario a favorito . . . . .	59
6.2.3.6 Eliminar itinerario de favorito . . . . .	60
6.2.3.7 Eliminar itinerario . . . . .	60
6.2.3.8 Añadir/eliminar/modificar opinión . . . . .	61
6.2.4 Vistas de información . . . . .	63
6.2.4.1 Ver información sobre plantas . . . . .	63
6.2.4.2 Ver mapa . . . . .	64
6.2.5 Vistas de administrador . . . . .	65
6.2.5.1 Eliminar usuarios . . . . .	66
6.2.5.1 Ver estadísticas . . . . .	67
6.3 API REST . . . . .	69
6.3.1 getRoutesList . . . . .	69
6.3.2 getFavorites . . . . .	70
6.3.3 deleteFav . . . . .	70
6.3.4 addFav . . . . .	71
6.3.5 getHistorialList . . . . .	72
6.3.6 addHistorial . . . . .	72
6.3.7 addRuta . . . . .	73
6.3.8 userList . . . . .	73
6.3.9 deleteUser . . . . .	74
6.3.10 changeUsername . . . . .	74
6.3.11 addOpinion . . . . .	75
6.3.12 deleteOpinion . . . . .	75
6.3.13 updateOpinion . . . . .	76
6.3.14 deleteRuta . . . . .	76
<b>Capítulo 7: Evaluación</b>	<b>78</b>
7.1 Resultados de la encuesta . . . . .	78
7.1.1 Datos generales del encuestado . . . . .	78
7.1.2 Vista del mapa . . . . .	79
7.1.3 Consulta de itinerarios . . . . .	79
7.1.4 Consulta de información sobre plantas . . . . .	80
7.1.5 Menú lateral, inicio de sesión y registro . . . . .	81
7.1.6 Crear itinerario . . . . .	82
7.1.7 Nuevas opciones en los itinerarios . . . . .	82
7.1.8 Opciones de perfil de usuario . . . . .	83
7.1.9 Administrador . . . . .	85
7.1.10 Satisfacción general con la aplicación . . . . .	86

<b>Capítulo 8: Conclusiones y trabajo futuro</b>	<b>88</b>
8.1 Conclusiones . . . . .	88
8.2 Trabajo futuro . . . . .	88
<b>Chapter 8: Conclusions and future work</b>	<b>90</b>
8.1 Conclusions . . . . .	90
8.2 Future work . . . . .	90
<b>Capítulo 9: Aportaciones individuales</b>	<b>92</b>
9.1 Francisco García Navarrete . . . . .	92
9.2 Fabricio Bryan Torres Samueza . . . . .	94
<b>Bibliografía</b>	<b>95</b>
<b>Anexos</b>	<b>96</b>
Anexo I: Guía de uso . . . . .	96
1. Menú principal sin autenticar . . . . .	96
2. Ver mapa . . . . .	96
3. Lista de itinerarios . . . . .	98
4. Detalles itinerario sin autenticar . . . . .	99
5. Lista de plantas . . . . .	101
6. Información planta . . . . .	102
7. Menú lateral sin autenticar . . . . .	103
8. Inicio de sesión y registro . . . . .	104
9. Menú principal autenticado . . . . .	106
10. Detalles itinerario autenticado . . . . .	107
11. Crear itinerario . . . . .	111
12. Menú lateral autenticado y vista de perfil . . . . .	117
13. Cambiar nombre de usuario y/o contraseña . . . . .	119
14. Ver historial, favoritos e itinerarios creados . . . . .	120
15. Eliminar cuenta . . . . .	122
16. Menú principal de administrador . . . . .	123
17. Lista de usuarios . . . . .	123
18. Estadísticas . . . . .	124
Anexo II: Guía de instalación de la aplicación . . . . .	126
1. Instalación de la API . . . . .	126
2. Instalación de la aplicación móvil . . . . .	126
Anexo III: Campos del formulario de evaluación . . . . .	127

## Índice de figuras

Figura 1	Diagrama de casos de uso módulo información . . . . .	21
Figura 2	Diagrama de casos de uso módulo itinerarios . . . . .	24
Figura 3	Diagrama de casos de uso módulo usuario . . . . .	28
Figura 4	Arquitectura del sistema . . . . .	36
Figura 5	Diagrama ER de la BBDD . . . . .	38
Figura 6	Colecciones de la BBDD . . . . .	38
Figura 7	Campos de la colección <i>users</i> . . . . .	39
Figura 8	Campos de la subcolección <i>rutasCreadas</i> . . . . .	39
Figura 9	Campos de la subcolección <i>opiniones</i> . . . . .	40
Figura 10	Campos de la subcolección <i>rutasFavoritas</i> . . . . .	40
Figura 11	Campos de la subcolección <i>historial</i> . . . . .	40
Figura 12	Fragmento de código - Códigos de color utilizados en la aplicación móvil .	41
Figura 13	Vista de Home sin autenticación del usuario . . . . .	42
Figura 14	Vista de Home con autenticación del usuario . . . . .	43
Figura 15	Fragmento de código - Código de la vista de Home . . . . .	43
Figura 16	Vista del menú lateral sin autenticación del usuario . . . . .	44
Figura 17	Vista del menú lateral con autenticación del usuario . . . . .	45
Figura 18	Fragmento de código - Código del menú lateral . . . . .	45
Figura 19	Vista inicio de sesión . . . . .	46
Figura 20	Fragmento de código - Código de autenticación de usuario . . . . .	46
Figura 21	Vista de registrar nuevo usuario . . . . .	47
Figura 22	Fragmento de código - Código de registro de usuario . . . . .	47
Figura 23	Fragmento de código - Cerrar sesión . . . . .	48
Figura 24	Vista del perfil de usuario . . . . .	48
Figura 25	Fragmento de código - Código del perfil de usuario . . . . .	49
Figura 26	Vista de cambiar nombre de usuario . . . . .	49
Figura 27	Fragmento de código - Código de cambiar nombre de usuario . . . . .	50
Figura 28	Vista de cambiar contraseña del usuario . . . . .	50
Figura 29	Fragmento de código - Código de cambiar contraseña . . . . .	50
Figura 30	Vista de ver historial . . . . .	51
Figura 31	Fragmento de código - Código del historial . . . . .	51
Figura 32	Vista de ver favoritos . . . . .	52
Figura 33	Fragmento de código - Código del historial . . . . .	52
Figura 34	Vista de itinerarios creados por el usuario . . . . .	53
Figura 35	Fragmento de código - Código de itinerarios del usuario . . . . .	53
Figura 36	Vista de eliminar cuenta del usuario . . . . .	54
Figura 37	Fragmento de código - Código de eliminar cuenta de usuario . . . . .	54
Figura 38	Vista de buscador de itinerarios . . . . .	55
Figura 39	Fragmento de código - Código de buscador de itinerarios . . . . .	55
Figura 40	Vista crear itinerario . . . . .	56
Figura 41	Fragmento de código - Código de crear itinerario . . . . .	56
Figura 42	Vista de ver itinerario de usuario no autenticado . . . . .	57
Figura 43	Vista de ver itinerario de usuario autenticado . . . . .	57
Figura 44	Fragmento de código - Código de ver itinerario . . . . .	58
Figura 45	Vista de iniciar itinerario . . . . .	58
Figura 46	Fragmento de código - Código de iniciar itinerario . . . . .	59
Figura 47	Vista de añadir itinerario a favorito . . . . .	59
Figura 48	Fragmento de código - Código de añadir itinerario a favorito . . . . .	59
Figura 49	Vista de eliminar itinerario de favorito . . . . .	60
Figura 50	Fragmento de código - Código de eliminar itinerario de favorito . . . . .	60
Figura 51	Vista de eliminar itinerario . . . . .	61
Figura 52	Fragmento de código - Código de eliminar itinerario . . . . .	61
Figura 53	Vista de añadir opinión . . . . .	62
Figura 54	Vista de modificar o eliminar opinión . . . . .	62
Figura 55	Fragmento de código - Código de añadir/modificar/eliminar opinión . . . .	63
Figura 56	Vista de eliminar cuenta del usuario . . . . .	64
Figura 57	Fragmento de código - Código de ver información de plantas . . . . .	64
Figura 58	CSV con la información de las plantas . . . . .	64

Figura 59	Vista ver mapa . . . . .	65
Figura 60	Fragmento de código - Código de ver mapa . . . . .	65
Figura 61	Vista de administrador . . . . .	66
Figura 62	Vista de eliminar usuarios . . . . .	67
Figura 63	Fragmento de código - Código de eliminar usuarios . . . . .	67
Figura 64	Vista de ver estadísticas . . . . .	68
Figura 65	Fragmento de código - Código de ver estadísticas . . . . .	69
Figura 66	Fragmento de código - Código del endpoint getRoutesList . . . . .	70
Figura 67	Fragmento de código - Código del endpoint getFavorites . . . . .	70
Figura 68	Fragmento de código - Código del endpoint deleteFav . . . . .	71
Figura 69	Fragmento de código - Código del endpoint addFav . . . . .	72
Figura 70	Fragmento de código - Código del endpoint getHistorialList . . . . .	72
Figura 71	Fragmento de código - Código del endpoint addHistorial . . . . .	73
Figura 72	Fragmento de código - Código del endpoint addRuta . . . . .	73
Figura 73	Fragmento de código - Código del endpoint userList . . . . .	74
Figura 74	Fragmento de código - Código del endpoint deleteUser . . . . .	74
Figura 75	Fragmento de código - Código del endpoint changeUsername . . . . .	74
Figura 76	Fragmento de código - Código del endpoint addOpinion . . . . .	75
Figura 77	Fragmento de código - Código del endpoint deleteOpinion . . . . .	76
Figura 78	Fragmento de código - Código del endpoint updateOpinion . . . . .	76
Figura 79	Fragmento de código - Código del endpoint deleteRuta . . . . .	77
Figura 80	Respuestas de la primera pregunta del cuestionario: edad . . . . .	78
Figura 81	Respuestas de la segunda pregunta del cuestionario: sexo . . . . .	78
Figura 82	Respuestas de la tercera pregunta del cuestionario: visitas al jardín . . . . .	79
Figura 83	Respuestas de la cuarta pregunta del cuestionario: vista del mapa . . . . .	79
Figura 84	Respuestas de la quinta pregunta del cuestionario: información de las parcelas . . . . .	79
Figura 85	Respuestas de la sexta pregunta del cuestionario: lista de itinerarios . . . . .	80
Figura 86	Respuestas de la séptima pregunta del cuestionario: buscador de itinerarios . . . . .	80
Figura 87	Respuestas de la octava pregunta del cuestionario: datos de los itinerarios . . . . .	80
Figura 88	Respuestas de la novena pregunta del cuestionario: lista de plantas . . . . .	80
Figura 89	Respuestas de la décima pregunta del cuestionario: buscador de plantas . . . . .	81
Figura 90	Respuestas de la undécima pregunta del cuestionario: información sobre plantas . . . . .	81
Figura 91	Respuestas de la duodécima pregunta del cuestionario: menú lateral . . . . .	81
Figura 92	Respuestas de la decimotercera pregunta del cuestionario: inicio de sesión . . . . .	82
Figura 93	Respuestas de la decimocuarta pregunta del cuestionario: registro sobre plantas . . . . .	82
Figura 94	Respuestas de la decimoquinta pregunta del cuestionario: creación de itinerarios . . . . .	82
Figura 95	Respuestas de la decimosexta pregunta del cuestionario: botón de favoritos . . . . .	83
Figura 96	Respuestas de la decimoséptima pregunta del cuestionario: botón de comentarios . . . . .	83
Figura 97	Respuestas de la decimoctava pregunta del cuestionario: tu perfil . . . . .	83
Figura 98	Respuestas de la decimonovena pregunta del cuestionario: cambiador nombre de usuario . . . . .	84
Figura 99	Respuestas de la vigésima pregunta del cuestionario: cambiar contraseña . . . . .	84
Figura 100	Respuestas de la vigésimo primera pregunta del cuestionario: ver historial . . . . .	84
Figura 101	Respuestas de la vigésimo segunda pregunta del cuestionario: ver favoritos . . . . .	85
Figura 102	Respuestas de la vigésimo tercera pregunta del cuestionario: ver itinerarios creados . . . . .	85
Figura 103	Respuestas de la vigésimo cuarta pregunta del cuestionario: menú principal de administrador . . . . .	85
Figura 104	Respuestas de la vigésimo quinta pregunta del cuestionario: lista de usuarios . . . . .	86
Figura 105	Respuestas de la vigésimo sexta pregunta del cuestionario: estadísticas . . . . .	86
Figura 106	Respuestas de la vigésimo séptima pregunta del cuestionario: diseño general de la aplicación . . . . .	86
Figura 107	Respuestas de la vigésimo octava pregunta del cuestionario: fluidez de la aplicación . . . . .	87
Figura 108	Menú principal de la aplicación sin autenticar . . . . .	96
Figura 109	Mapa del jardín . . . . .	97
Figura 110	Información sobre las parcelas del jardín . . . . .	98

Figura 111	Lista de itinerarios . . . . .	99
Figura 112	Vista detallada de un itinerario . . . . .	100
Figura 113	Instrucciones de un itinerario . . . . .	101
Figura 114	Lista de plantas . . . . .	102
Figura 115	Información detallada de una planta . . . . .	103
Figura 116	Menú principal de la aplicación sin autenticar . . . . .	104
Figura 117	Inicio de sesión . . . . .	105
Figura 118	Registro de usuario . . . . .	106
Figura 119	Menú principal de la aplicación autenticado . . . . .	107
Figura 120	Vista detallada de un itinerario al estar autenticado I . . . . .	108
Figura 121	Vista detallada de un itinerario al estar autenticado II . . . . .	109
Figura 122	Añadir opinión . . . . .	110
Figura 123	Eliminar itinerario . . . . .	111
Figura 124	Menú de creación de itinerarios I . . . . .	112
Figura 125	Menú de creación de itinerarios II . . . . .	113
Figura 126	Menú de creación de itinerarios III . . . . .	114
Figura 127	Menú de creación de itinerarios IV . . . . .	115
Figura 128	Menú de creación de itinerarios V . . . . .	116
Figura 129	Menú de creación de itinerarios VI . . . . .	117
Figura 130	Menú lateral autenticado . . . . .	118
Figura 131	Vista de perfil . . . . .	119
Figura 132	Cambiar nombre de usuario . . . . .	120
Figura 133	Cambiar contraseña . . . . .	120
Figura 134	Historial . . . . .	121
Figura 135	Favoritos . . . . .	121
Figura 136	Itinerarios creados . . . . .	122
Figura 137	Eliminar cuenta . . . . .	122
Figura 138	Menú principal de administrador . . . . .	123
Figura 139	Lista de usuarios . . . . .	124
Figura 140	Estadísticas . . . . .	125
Figura 141	Sección 1 del formulario: datos generales del encuestado . . . . .	127
Figura 142	Sección 2 del formulario: satisfacción con la vista del mapa . . . . .	128
Figura 143	Sección 3 del formulario: satisfacción con la consulta y vista de itinerarios . . . . .	129
Figura 144	Sección 4 del formulario: satisfacción con la consulta y vista de plantas . . . . .	130
Figura 145	Sección 5 del formulario: satisfacción con la consulta y vista de plantas . . . . .	131
Figura 146	Sección 6 del formulario: satisfacción con la vista de crear itinerario . . . . .	131
Figura 147	Sección 7 del formulario: satisfacción con la nuevas opciones de los itinerarios al estar autenticado . . . . .	132
Figura 148	Sección 8.1 del formulario: satisfacción con las opciones de perfil de usuario I . . . . .	133
Figura 149	Sección 8.2 del formulario: satisfacción con las opciones de perfil de usuario II . . . . .	133
Figura 150	Sección 9 del formulario: satisfacción con la vista y opciones del administrador . . . . .	134
Figura 151	Sección 10 del formulario: satisfacción general con la aplicación . . . . .	135
Figura 152	Sección 11 del formulario: agradecimientos . . . . .	135

# Capítulo 1: Introducción

En este capítulo se explicará la motivación que ha llevado a desarrollar el proyecto, cuáles fueron los objetivos planteados y se describirá la estructura de la memoria.

## 1.1 Motivación

El jardín botánico de Madrid es el espacio con mayor diversidad de plantas que existe en España. Sin embargo, realizar un recorrido por el mismo puede ser bastante tedioso si no se dispone de un itinerario definido. En este sentido, el propio jardín propone un itinerario recomendado, pero este no tiene por qué satisfacer las expectativas de la persona que lo utilice.

Por otro lado, todas las plantas y árboles que contiene el jardín se encuentran etiquetadas y con cierta información acerca de cada una de ellas, como por ejemplo su procedencia o su nombre científico. Sin embargo, algo muy útil para el visitante sería la posibilidad de poder acceder a más información acerca de cada planta cuando se encuentra observándola.

Por último, otra necesidad que puede surgirle a un visitante es la búsqueda de cómo llegar a una zona del jardín o un tipo concreto de plantas desde el lugar en el que se encuentra.

Para ello, se ha implementado una aplicación móvil que facilite realizar todo lo mencionado anteriormente, la cual consta de un sistema de consulta y gestión de itinerarios, un sistema de consultas sobre los elementos del jardín, y un mapa interactivo que permite consultar la ubicación de los elementos que contiene el jardín y recibir indicaciones sobre cómo llegar a cada lugar.

Se puede acceder al código desarrollado en el siguiente enlace, disponible solo para personal de la Universidad Complutense de Madrid:  
[https://drive.google.com/file/d/1KsltpMVTOROTqYn9U0\\_RP8lFBpwaxDxC/view?usp=sharing](https://drive.google.com/file/d/1KsltpMVTOROTqYn9U0_RP8lFBpwaxDxC/view?usp=sharing)

## 1.2 Objetivos

El objetivo principal de este proyecto es desarrollar una aplicación móvil que facilite la visita del jardín y la consulta de información acerca de las plantas y árboles que contiene. Este objetivo se puede refinar en los siguientes objetivos más específicos:

- Facilitar a los visitantes la posibilidad de gestionar (crear, publicar y consultar) itinerarios definidos sobre el jardín botánico.
- Permitir a un visitante añadir comentarios a itinerarios publicados por otros usuarios.
- Facilitar a un visitante el acceso a la información acerca de las plantas y árboles del jardín mientras realiza la visita.
- Permitir la visualización de la ubicación de cada elemento del jardín mediante un mapa interactivo facilitando instrucciones sobre la forma de alcanzarlo.
- Permitir la gestión de la información de los usuarios de la aplicación así como de la información generada por los mismos.
- Facilitar la creación y exportación de análisis sobre los datos de uso y la actividad que se genera en la aplicación.
- Implementar y publicar una API REST desde la que se pueda acceder a los diferentes servicios que ofrece la aplicación.

## 1.3 Estructura de la memoria

La memoria del proyecto se estructura de la siguiente forma.

- En el capítulo 1 se describe la motivación para el trabajo, se plantean los objetivos principales y secundarios y se realiza una descripción de la estructura de la memoria.
- En el capítulo 2 se realiza una revisión de algunas herramientas con funcionalidades similares al trabajo realizado en este proyecto.

- En el capítulo 3 se lista la tecnología empleada en el desarrollo de este proyecto.
- En el capítulo 4 se describen los casos de uso del proyecto y los tres módulos por los que están compuestos: información, itinerarios y usuario.
- En el capítulo 5 se trata la arquitectura, además de la forma en la que se ha realizado la persistencia de la información del sistema creado, explicando cada uno de los campos que componen la base de datos junto a figuras para facilitar su comprensión.
- En el capítulo 6 se describen el diseño, la funcionalidad y la API REST de la aplicación.
- En el capítulo 7 se describen la evaluación de la aplicación por parte de usuarios.
- En el capítulo 8 trata de las conclusiones, los objetivos que se han cumplido y el trabajo a futuro.
- En el capítulo 9 trata de las aportaciones individuales de cada miembro del grupo.

# Chapter 1: Introduction

In this chapter the motivation that has led to the development of this project, the proposed objectives and the structure of the report will be explained.

## 1.1 Motivation

The Real Jardín Botánico de Madrid is the space with the largest diversity of plants that exists in Madrid. However, follow a route along it can be quite difficult if a defined path does not exist. In this sense, the garden itself provides a recommended route, although it doesn't have to satisfy the person's expectations.

On the other hand, every plant and tree that the garden contains are labeled and have some information about them, like their origin or their scientific name. However, something very useful for a visitor would be the possibility of being able to access more information about the plant when they are watching it.

Ultimately, another necessity that can arise to the visitor is how to get to a specific part of the garden or a certain kind of plant from where they are.

To do so, it has been implemented a mobile application that makes easy to do all the previously mentioned things, which consists of an itinerary query and management system, an information about the plants of the garden query system and an interactive map of the garden that allows to query the location of the elements that the garden has and receive indications to reach them has been developed.

The code can be accessed from the next link, available only for the personal from the Universidad Complutense de Madrid:  
[https://drive.google.com/file/d/1KsltpMVTOROTqYn9U0\\_RP8lFBpwaxDxC/view?usp=sharing](https://drive.google.com/file/d/1KsltpMVTOROTqYn9U0_RP8lFBpwaxDxC/view?usp=sharing)

## 1.2 Objectives

The main objective of this project is the development of a mobile application that facilitates the visits and the information about the plants and trees of the garden queries.

This objective can be defined with the next more specific objectives:

- Bring the possibility of manage (create, publish and consult) itineraries in the botanic garden to the visitors.
- Allow a visitor to add commentaries to the itineraries published by the other users.
- Make it easier for the visitors to access the information about the plants and trees of the garden while they are visiting it.
- Allow the visualization of each of the garden's elements through the use of an interactive map that shows directions to reach them.
- Allow the management of the users' information in the application along the information they add to the application themselves.
- Create and export statistics to analyze the usement data and activity that the application generates.
- Develop and publish an API REST from where access the services the application offers.

## 1.3 Structure of the report

The structure of this report is as follows.

- In the chapter 1 the motivation of this project is described, the main and secondary objectives are proposed and a description of the report's structure is made.
- In the chapter 2 a review of some tools with similar functionality to the work done in this project is made.

- In the chapter 3 the technology used for the development of this project is listed.
- In the chapter 4 the cases of use of this project and its 3 modules it is made of (information, itineraries and user) are described.
- In the chapter 5 it is talked about the architecture of the application and also it is explained how the persistence of the information of the system created is done, illustrating each field that composes the database along with figures to ease its understanding.
- In the chapter 6 it is described the design, the functionality and the API REST used in the application.
- In the chapter 7 it is described the evaluation of the application done by the users.
- In the chapter 8 talks about the conclusions, the objectives achieved and the future work.
- In the chapter 9 it is discussed the individual contributions of each member of the group.

## Capítulo 2: Estado del arte

En el presente capítulo se recogerán las características más importantes acerca del software semejante que hay actualmente en el mercado.

La aplicación implementada es un sistema de gestión de itinerarios. En este ámbito se han encontrado dos tipos de software: los sistemas que facilitan la identificación de plantas y el software destinado a muestra de detalles plantas.

### 2.1 Sistemas de identificación plantas

Los sistemas de identificación de plantas, ya sean flores, arboles silvestres o asilvestrados. Cuentan con módulos para la gestión de la información de las plantas

Algunos ejemplos de sistemas de identificación de plantas son:

#### 2.1.1 ArbolApp

[1] Se trata de un sistema de identificación de los árboles silvestres y asilvestrados de la Península Ibérica y de las Islas Baleares, mediante una búsqueda que puede ser guiada o abierta. La búsqueda guiada se organiza en una serie de alternativas en las cuales se debe elegir el enunciado que coincida o que más se acerque a las características del árbol que se quiere identificar. La opción elegida enviará al usuario a otra alternativa, y así sucesivamente hasta llegar a un género o una especie. La búsqueda abierta permite marcar las características del árbol observado que más llamen la atención del usuario, hasta llegar a la especie en cuestión o al menos a unas pocas seleccionadas entre las que estará la buscada. Este sistema puede ser utilizado desde cualquier dispositivo electrónico ya que los usuarios disponen, tanto de una aplicación web como de una aplicación móvil. Entre sus funcionalidades destaca:

- Búsqueda guiada
- Búsqueda abierta
- Listado de especies
- Gestionar favoritos
- Glosario de términos

#### 2.1.2 PlantNet

[2] Se trata de un sistema de identificación de plantas mediante una comparación con las imágenes ya existentes en una inmensa base de datos botánica, la cual va creciendo de manera constante gracias a las aportaciones y a las fotografías, los usuarios disponen de una aplicación web y de una aplicación móvil para poder acceder a sus funcionalidades. Entre sus funcionalidades destaca:

- Realizar una nueva observación
- Búsqueda de flora mundial por tipo de plantas
- Búsqueda de flora por género, especie o familia
- Listado de las últimas fotos de las especies compartidas

### 2.2 Sistemas de muestra de detalles de plantas

La aplicación de muestra de detalles de plantas se caracteriza porque permiten al usuario obtener información detallada de las plantas.

### **2.2.1 RJB Museo vivo**

[3] Aplicación móvil, anteriormente disponible para Android e IOS pero actualmente solo se encuentra disponible para IOS, la cual consta de un sistema que muestra cada detalle del museo vivo albergado en el parque. Entre sus funcionalidades destaca:

- Audioguía con la descripción de las plantas y los principales atractivos del jardín
- Visitas guiadas
- Información sobre plantas y arquitectura
- Plano del Real Jardín Botánico de Madrid
- Zonas de interés
- Otros servicios

### **2.2.2 iKnow Trees 2 LITE**

[4] Aplicación móvil disponible solo para Android, la cual se trata de un sistema de muestra de información e imágenes de más de 200 especies de árboles de Europa y América del norte con la cual poder identificar especies sin la ayuda de un libro. Sus principales funcionalidades son:

- Listado de árboles
- Gestionar favoritos
- Lista de los últimos árboles consultados
- Mostrar información del árbol seleccionado

## Capítulo 3: Tecnología empleada

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

### 3.1 Ionic

Ionic es un framework de desarrollo software de código abierto [5] creado por Max Lynch, Ben Sperry, y Adam Bradley de Drifty Co. en 2013. Este framework está orientado al desarrollo de aplicaciones móviles, web y web progresivas. Algunas de sus ventajas son:

- Desarrollo multiplataforma con el mismo código.
- Componentes visuales prediseñados.
- Herramientas amigables con el desarrollador.
- Permite el desarrollo con múltiples herramientas.

### 3.2 Angular

Angular es un framework para aplicaciones web [6] desarrollado en TypeScript, de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página. Su objetivo es aumentar las aplicaciones basadas en navegador con capacidad de Modelo Vista Controlador (MVC) [7], en un esfuerzo para hacer que el desarrollo y las pruebas sean más fáciles.

La biblioteca lee el HTML que contiene atributos de las etiquetas personalizadas adicionales, entonces obedece a las directivas de los atributos personalizados, y une las piezas de entrada o salida de la página a un modelo representado por las variables estándar de JavaScript.

### 3.3 HTML5 y CSS3

HTML5 [8] es la última versión del lenguaje de marcado para el desarrollo de páginas web. Esta versión establece un conjunto de nuevos elementos y atributos con significado semántico que manifiestan las nuevas necesidades de las páginas web modernas.

HTML [9] describe el contenido de la página web, pero no su funcionalidad. En este sentido usa marcado para estructurar el contenido, tales como imágenes o texto. Aunque puede describir, hasta un cierto punto, la apariencia de un documento, otras tecnologías son usadas generalmente para especificar la presentación de una página web.

CSS [10] es el lenguaje utilizado para describir la presentación de una página web. Una hoja de estilo CSS se compone de reglas que modifican la manera en que un documento, normalmente HTML, es presentado. Una regla CSS se compone de un selector, que seleccionará los elementos HTML que se actualizarán a través de una serie de propiedades.

### 3.4 Firebase y Firestore

Firebase [11] es una plataforma para el desarrollo de aplicaciones web y aplicaciones móviles lanzada en 2011 y adquirida por Google en 2014.

Es una plataforma ubicada en la nube, integrada con Google Cloud Platform [12], que usa un conjunto de herramientas para la creación y sincronización de proyectos que serán dotados de alta calidad, haciendo posible el crecimiento del número de usuarios y dando resultado también a la obtención de una mayor monetización. Sus principales ventajas son:

- Sincronización fácil de datos del proyecto.
- Compatibilidad multiplataforma.
- Uso de la infraestructura de Google, permitiendo un escalado automático.
- Creación de proyectos sin necesidad de servidor.

Cloud Firestore [13] es un servicio de almacenamiento de datos derivado de Google Cloud Platform, adaptado a la plataforma de Firebase. Al igual que Realtime Database [14], es una base de datos NoSQL, aunque presenta diversas diferencias. Se organiza en forma de documentos agrupados en colecciones, y en ellos se pueden incluir tanto campos de diversos tipos (cadenas

de texto, números, puntos geográficos, referencias a la propia base de datos, arrays, booleanos, marcas de tiempo, e incluso objetos propios) como otras subcolecciones. Ente sus limitaciones más destacadas encontramos la de no soportar las búsquedas de texto tipo "LIKE", eso es, buscar por subcadenas del texto almacenado, y la de no poder filtrar las búsquedas con condiciones que impliquen más de un campo, si no es por búsquedas por el texto exacto.

### 3.5 Leaflet

Leaflet [15] es una biblioteca JavaScript de código abierto utilizada para desarrollar aplicaciones que utilicen mapas. Su primera versión se lanzó en 2011 y soporta la mayoría de plataformas móviles y de escritorio, soportando HTML5 [8] y CSS3 [10].

Leaflet permite a los desarrolladores mostrar mapas almacenados en un servidor público fácilmente, cargar datos GEOJson [16], cambiar su estilo y crear capas interactivas (por ejemplo, marcadores que muestren información al pulsarlos).

### 3.6 Mapbox

Mapbox [17] es un proveedor de mapas en línea realizados por encargo. Mapbox es el creador, o un colaborador significativo, de algunas bibliotecas de mapeo de código abierto y aplicaciones, entre ellas la biblioteca de Javascript de Leaflet [15].

Los datos son tomados tanto de bases de datos abiertas como de bases de datos de propietarios. La tecnología está basada en Node.js [18], CouchDB [19], Mapnik [20], GDAL [21], y Leafletjs [15].

### 3.7 NodeJS

Node.js es un entorno en tiempo de ejecución multiplataforma [18], de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación JavaScript, asíncrono, con E/S de datos en una arquitectura orientada a eventos [22] y basado en el motor V8 de Google [23]. Fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables, como por ejemplo, servidores web. Fue creado por Ryan Dahl en 2009 y su evolución está apadrinada por la empresa Joyent, que además tiene contratado a Dahl en plantilla.

### 3.8 NPM

npm [24] es el sistema de gestión de paquetes por defecto para Node.js [18], un entorno de ejecución para JavaScript. Consiste en una interfaz de línea de comandos, también llamada npm, y una base de datos online con paquetes públicos y privados, llamada npm registry. NPM administra las dependencias locales de un proyecto, como también las herramientas de JavaScript que estén instaladas en el sistema.

### 3.9 Visual Studio Code

Visual Studio Code [25] es un editor de código fuente desarrollado por Microsoft para Windows, Linux y macOS. Incluye soporte para la depuración, control integrado de Git [26], resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código. También es personalizable, por lo que los usuarios pueden cambiar el tema del editor, los atajos de teclado y las preferencias. Es gratuito y de código abierto, aunque la descarga oficial está bajo software privativo e incluye características personalizadas por Microsoft.

### 3.10 Git

Git [26] un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia, la confiabilidad y compatibilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente. Su propósito es llevar registro de los cambios en archivos de computadora incluyendo coordinar el trabajo que varias personas realizan sobre archivos compartidos en un repositorio de código.

### 3.11 Gimp

GIMP (siglas en inglés de GNU Image Manipulation Program) [27] es un programa de edición de imágenes digitales en forma de mapa de bits, tanto dibujos como fotografías. Es un programa libre y gratuito. Es el programa de manipulación de gráficos disponible en más sistemas operativos.

GIMP tiene herramientas que se utilizan para el retoque y edición de imágenes, dibujo de formas libres, cambiar el tamaño, recortar, hacer fotomontajes, convertir a diferentes formatos de imagen, y otras tareas más especializadas. Se pueden también crear imágenes animadas en formato GIF e imágenes animadas en formato MPEG usando un plugin de animación.

Los desarrolladores y encargados de mantener GIMP se esfuerzan en mantener y desarrollar una aplicación gráfica de software libre, de alta calidad para la edición y creación de imágenes originales, de fotografías, de íconos, de elementos gráficos tanto de páginas web como de elementos artísticos de interfaz de usuario.

## Capítulo 4: Especificación de requisitos

En este capítulo de la memoria se van a especificar los requisitos de la aplicación desarrollada. Para ello, en primer lugar se describirán los actores que interactuarán con la aplicación y, a continuación, se desarrollarán los casos de uso en los que intervienen los actores agrupados por módulos funcionales.

### 4.1 Actores

Los actores definidos en los casos de usos son los siguientes:

- Usuario sin registrar. Son usuarios que no disponen de cuenta en la aplicación. Solo pueden consultar la información pública que ofrece la aplicación. No pueden realizar ninguna función más.
- Usuario registrado. Son usuarios que han registrado una cuenta en la aplicación. Pueden consultar la información e incluso añadir y/o modificar la suya propia.
- Administrador. Es el único usuario encargado de realizar el mantenimiento de la aplicación.

La funcionalidad de la aplicación se ha agrupado en tres módulos funcionales:

- Módulo información
- Módulo itinerarios
- Módulo usuario

En los siguientes apartados se desarrollan cada uno de estos módulos.

### 4.2 Módulo información

En este módulo se ha agrupado toda la funcionalidad referente a la gestión de la información que se muestra sobre el jardín botánico. En la figura 1 se muestra el diagrama de casos de uso asociado al módulo.

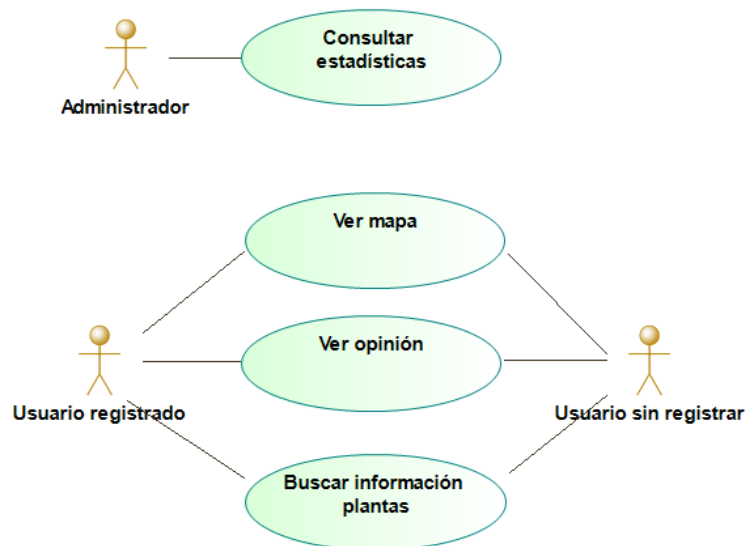


Figura 1: Diagrama de casos de uso módulo información

A continuación se van a describir los casos de uso que forman parte de este módulo.

<b>JARDÍN01</b>	<b>Consultar estadísticas</b>
Actores	Administrador
Descripción	Consulta diversas estadísticas relativas a la aplicación
Precondición	Iniciar sesión con una cuenta de administrador
Secuencia normal	<ol style="list-style-type: none"> <li>1. El administrador accede al menú de estadísticas</li> <li>2. El sistema muestra las estadísticas</li> </ol>
Postcondición	El sistema muestra los resultados de la búsqueda

<b>JARDÍN02</b>	<b>Ver mapa</b>
Actores	Usuario registrado/sin registrar
Descripción	Muestra el mapa del jardín botánico con su diversa información
Precondición	
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario pulsa en la opción para mirar el mapa</li> <li>2. El usuario navega por el mismo hasta que le da a salir o pulsa sobre algún elemento para ver su información</li> </ol>
Postcondición	Muestra la información solicitada por el usuario o sale del mismo.

<b>JARDÍN03</b>	<b>Ver opinión</b>
Actores	Usuario registrado/sin registrar
Descripción	Muestra la opinión sobre un itinerario del jardín botánico
Precondición	La opinión y el elemento al que hace referencia deben existir en la aplicación
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario accede al itinerario correspondiente</li> <li>2. En la parte inferior se muestra el usuario que ha creado dicha opinión y el texto de la misma</li> </ol>
Postcondición	Se muestra la opinión sobre el itinerario seleccionado
Excepciones	<ul style="list-style-type: none"> <li>- Paso 1: el itinerario no existe. La aplicación informa del error</li> <li>- Paso 2: el usuario que opina ha sido eliminado o lo ha sido su opinión. Se ignora que haya existido dicha opinión</li> </ul>

<b>JARDÍN04</b>	<b>Buscar información plantas</b>
Actores	Usuario registrado/sin registrar
Descripción	Muestra la información sobre una planta
Precondición	Es necesario que la información de la planta indicada se encuentre almacenada en el sistema
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario introduce el nombre de la planta a consultar</li> <li>2. El sistema devuelve la información asociada a dicha planta</li> </ol>
Postcondición	Se obtiene la información de la planta indicada
Excepciones	<ul style="list-style-type: none"> <li>- Paso 1: no existe la planta indicada. El sistema lo indica debidamente</li> </ul>

<b>JARDÍN05</b>	<b>Consultar datos</b>
Actores	Administrador
Descripción	Consulta la información almacenada
Precondición	Iniciar sesión con una cuenta de administrador
Secuencia normal	<ol style="list-style-type: none"> <li>1. El administrador accede al menú de búsqueda</li> <li>2. Introduce los valores de búsqueda</li> <li>3. El sistema muestra los resultados de la búsqueda</li> </ol>
Postcondición	El sistema muestra los resultados de la búsqueda
Excepciones	<ul style="list-style-type: none"> <li>- Paso 2: Que los valores introducidos en la búsqueda no coincidan con ningún valor almacenado. Se informa al administrador de ello</li> </ul>

### 4.3 Módulo itinerarios

En este módulo se ha agrupado la funcionalidad referente a la gestión y visionado de los itinerarios que se pueden utilizar para visitar el jardín botánico. En la figura 2 se muestra el diagrama de casos de uso asociado al módulo.

A continuación se van a describir los casos de uso que forman parte de este módulo.

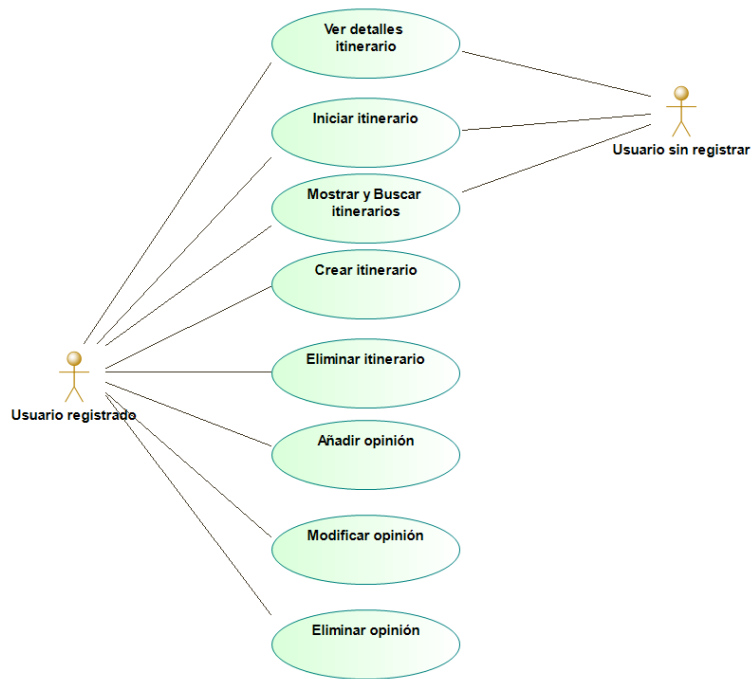


Figura 2: Diagrama de casos de uso módulo itinerarios

<b>JARDÍN06</b>	<b>Ver detalles itinerario</b>
Actores	Usuario registrado/sin registrar
Descripción	Muestra la información detallada de un itinerario
Precondición	Que el itinerario exista en la aplicación
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario pulsa sobre un itinerario existente</li> <li>2. Se muestran la información correspondiente al mismo</li> </ol>
Postcondición	Se muestra la información del itinerario seleccionada
Excepciones	- Paso 2: itinerario no existe en el sistema. Se informa del error correspondiente

<b>JARDÍN07</b>	<b>Iniciar itinerario</b>
Actores	Usuario registrado/sin registrar
Descripción	Permite comenzar el seguimiento del itinerario por el jardín
Precondición	Haber accedido a los detalles de un itinerario existente
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario pulsa sobre el botón para comenzar el itinerario</li> <li>2. Se muestra el mapa del jardín con una línea marcando el recorrido a seguir y la ubicación actual del usuario</li> </ol>
Postcondición	Muestra el mapa con el itinerario a seguir

<b>JARDÍN08</b>	<b>Mostrar y buscar itinerarios</b>
Actores	Usuario registrado/sin registrar
Descripción	Muestra una lista con los itinerarios existentes en el sistema y un menú para buscar itinerarios en concreto
Precondición	
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario accede al menú correspondiente y se muestran todos los itinerarios</li> <li>2. (Opcional) El usuario selecciona los parámetros de búsqueda que desee entre los ofrecidos y se filtran los resultados</li> </ol>
Postcondición	Se muestran los itinerarios de acuerdo a los criterios de búsqueda
Excepciones	<ul style="list-style-type: none"> <li>- Paso 1: no existen itinerarios en el sistema. Se informa debidamente de ello</li> <li>- Paso 2: no existen resultados que coincidan con los parámetros de búsqueda fijados. Se informa al usuario de ello</li> </ul>

<b>JARDÍN09</b>	<b>Crear itinerarios</b>
Actores	Usuario registrado
Descripción	Permite a un usuario registrado crear un itinerario para visitar el jardín
Precondición	El usuario debe haber iniciado sesión
Secuencia normal	<ol style="list-style-type: none"> <li>1. usuario accede al menú correspondiente</li> <li>2. Indica los distintos parámetros necesarios para añadir el itinerario al sistema</li> <li>3. Confirma la creación del itinerario</li> </ol>
Postcondición	Se añade un nuevo itinerario al sistema
Excepciones	<ul style="list-style-type: none"> <li>- Paso 2: el usuario introduce un valor erróneo. Se le informa de ello para que lo vuelva a introducir correctamente</li> <li>- Paso 3: el itinerario contiene algún valor clave repetido. Se informa al usuario de que lo cambie.</li> </ul>

<b>JARDÍN10</b>	<b>Eliminar itinerario</b>
Actores	Usuario registrado
Descripción	Se elimina el itinerario
Precondición	Que exista el itinerario, que sea del mismo usuario y que este haya iniciado sesión
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario accede al itinerario que desea eliminar</li> <li>2. Se da click en el botón eliminar (solo se mostrará si el creador es el mismo que el usuario autenticado)</li> <li>3. El sistema notifica el borrado</li> </ol>
Postcondición	El itinerario queda eliminado del sistema
Excepciones	<ul style="list-style-type: none"> <li>- Paso 1: el nombre del itinerario no coincide con ninguno ya sea porque no existe o porque existía y ha sido eliminado anteriormente</li> </ul>

<b>JARDÍN11</b>	<b>Añadir opinión</b>
Actores	Usuario registrado
Descripción	Añade una nueva opinión sobre un elemento del jardín
Precondición	El usuario debe haber iniciado sesión y el elemento debe existir en el sistema
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario accede al elemento sobre el que quiere opinar</li> <li>2. Va a la sección del mismo para añadir su opinión</li> <li>3. Introduce la puntuación que le quiera dar al elemento y, opcionalmente, un texto que la acompañe</li> </ol>
Postcondición	Se añade una nueva opinión asociada al elemento al sistema
Excepciones	<ul style="list-style-type: none"> <li>- Paso 1: el elemento no existe. Muestra un mensaje de error.</li> <li>- Paso 2: el usuario ya ha añadido una opinión. Se le informa que ya ha añadido una y que no puede añadir más</li> </ul>

<b>JARDÍN12</b>	<b>Eliminar opinión</b>
Actores	Usuario registrado
Descripción	Se eliminan la opinión
Precondición	Que exista la opinión y que sea del propio usuario autenticado
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario accede al elemento en el que aparece la opinión que desea borrar</li> <li>2. Pulsa sobre el botón de modificar la opinión (solo aparece si el usuario autenticado es el autor)</li> <li>3. En este menú pulsa sobre el botón de eliminar</li> <li>4. El sistema elimina la opinión</li> </ol>
Postcondición	La opinión asociada al elemento desaparece del sistema
Excepciones	<ul style="list-style-type: none"> <li>• Paso 1: el elemento no existe. Se le indica al usuario que ese elemento ya no existe</li> </ul>

<b>JARDÍN13</b>	<b>Modificar opinión</b>
Actores	Usuario registrado
Descripción	Se modifica la opinión
Precondición	Que exista la opinión y que sea del propio usuario autenticado
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario accede al elemento en el que aparece la opinión que desea modificar</li> <li>2. Pulsa sobre el botón de modificar la opinión (solo aparece si el usuario autenticado es el autor)</li> <li>3. Introduce los nuevos datos que desee</li> <li>4. El sistema actualiza la opinión</li> </ol>
Postcondición	La opinión asociada al elemento se actualiza
Excepciones	- Paso 1: El elemento no existe. Se le indica al usuario que ese elemento ya no existe

#### 4.4 Módulo usuario

Este módulo contiene toda la funcionalidad encargada a la gestión de la información de los propios usuarios de la aplicación. En la figura 3 se muestra el diagrama de casos de uso asociado al módulo.

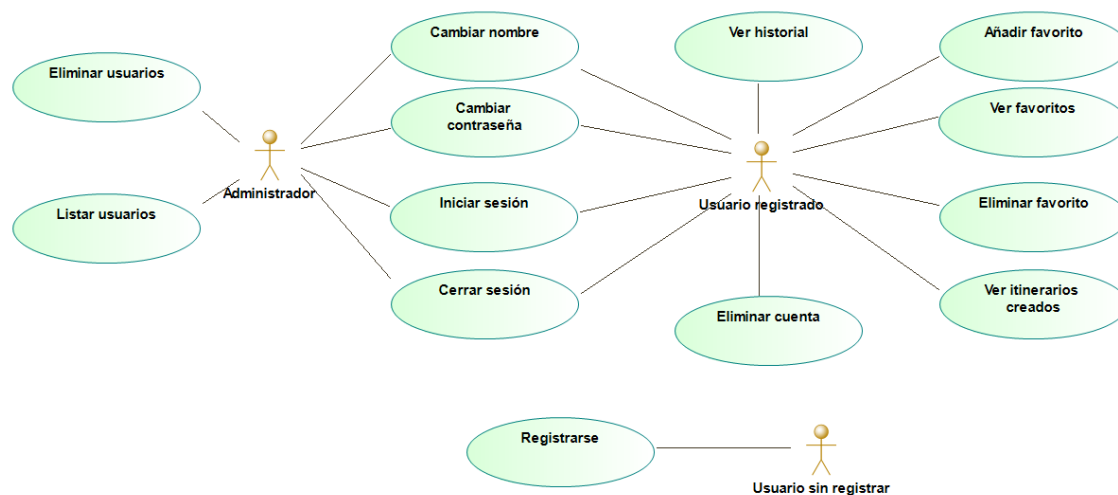


Figura 3: Diagrama de casos de uso módulo usuario

A continuación se van a describir los casos de uso que forman parte de este módulo.

<b>JARDÍN14</b>	<b>Eliminar usuarios</b>
Actores	Administrador
Descripción	Se eliminan todos los datos del usuario
Precondición	El usuario que se desea eliminar tiene que estar previamente registrado
Secuencia normal	<ol style="list-style-type: none"> <li>1. El administrador selecciona de una lista de usuarios mediante una búsqueda al usuario que desea eliminar</li> <li>2. Se da click en el botón eliminar</li> <li>3. El sistema avisa de la actualización de la lista</li> </ol>
Postcondición	El usuario queda eliminado del sistema
Excepciones	<ul style="list-style-type: none"> <li>- Paso 1: el nombre de usuario no coincide con ninguno ya sea porque no existe o porque existía y ha sido eliminado anteriormente.</li> </ul>
Comentarios	Los usuarios con más probabilidad de ser eliminados son aquellos que continuamente realicen actos indebidos

<b>JARDÍN15</b>	<b>Listar usuarios</b>
Actores	Administrador
Descripción	Muestra una lista con todos los usuarios registrados y permite filtrar los resultados
Precondición	El usuario debe ser administrador y haber iniciado sesión
Secuencia normal	<ol style="list-style-type: none"> <li>1. El administrador pulsa sobre el botón correspondiente para acceder al menú</li> <li>2. El sistema muestra todos los usuarios existentes</li> <li>3. (Opcional) El administrador introduce datos para filtrar los resultados</li> </ol>
Postcondición	El sistema muestra los resultados de la consulta
Excepciones	<ul style="list-style-type: none"> <li>- Paso 2: no existen usuarios. El sistema muestra un mensaje indicándolo</li> <li>- Paso 3: no hay usuarios que coincidan con los filtros indicados. El sistema lo notifica al administrador</li> </ul>

<b>JARDÍN16</b>	<b>Registrarse</b>
Actores	Usuario sin registrar
Descripción	Se registra al nuevo usuario en el sistema
Precondición	Que el usuario no esté registrado o que no haya iniciado sesión
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción “registrarse”</li> <li>2. Introduce los valores correctamente en los campos especificados</li> <li>3. Selecciona la opción confirmar</li> <li>4. El sistema avisa del registro del usuario</li> </ol>
Postcondición	El usuario queda registrado en el sistema
Excepciones	<ul style="list-style-type: none"> <li>- Paso 2: el valor del campo introducido no es correcto, se pedirá al usuario un nuevo valor o que el usuario ya esté registrado</li> </ul>
Comentarios	El valor a introducir no debe contener ningún carácter no válido

<b>JARDÍN17</b>	<b>Iniciar sesión</b>
Actores	Administrador, usuario registrado
Descripción	El usuario inicia sesión
Precondición	El usuario debe estar registrado
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario selecciona “Iniciar sesión”</li> <li>2. Introduce los valores requeridos y confirma</li> <li>3. El sistema avisa del inicio de sesión al usuario</li> </ol>
Postcondición	El usuario inicia la sesión
Excepciones	<ul style="list-style-type: none"> <li>- Paso 2: los valores introducidos por el usuario no coinciden con ningún usuario registrado</li> </ul>

<b>JARDÍN17</b>	<b>Iniciar sesión</b>
Actores	Administrador, usuario registrado
Descripción	El usuario inicia sesión
Precondición	El usuario debe estar registrado
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario selecciona “Iniciar sesión”</li> <li>2. Introduce los valores requeridos y confirma</li> <li>3. El sistema avisa del inicio de sesión al usuario</li> </ol>
Postcondición	El usuario inicia la sesión
Excepciones	<ul style="list-style-type: none"> <li>- Paso 2: Los valores introducidos por el usuario no coinciden con ningún usuario registrado</li> </ul>

<b>JARDÍN18</b>	<b>Cerrar sesión</b>
Actores	Administrador, usuario registrado
Descripción	El usuario cierra sesión
Precondición	El usuario debe haber iniciado sesión
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario pulsa sobre el botón de cerrar sesión</li> <li>2. Se la cierra sesión</li> </ol>
Postcondición	Se cierra la sesión del usuario

<b>JARDÍN19</b>	<b>Cambiar nombre de usuario</b>
Actores	Administrador, usuario registrado
Descripción	Se editará el campo “nombre” del usuario
Precondición	El usuario debe ser poseedor de la cuenta y previamente haber iniciado sesión
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario accede a su perfil</li> <li>2. Selecciona la opción “cambiar nombre de usuario”</li> <li>3. Selecciona el campo que desea modificar (en este caso “nombre”)</li> <li>4. Introduce un nuevo valor y selecciona confirmar.</li> <li>5. El sistema avisa de la actualización del nombre de usuario</li> </ol>
Postcondición	El nombre de usuario queda actualizado con su nuevo valor introducido
Excepciones	<ul style="list-style-type: none"> <li>- Paso 2: el valor del campo introducido no es correcto y se pedirá al usuario un nuevo valor</li> </ul>
Comentarios	El valor a introducir no debe contener ningún carácter no válido

<b>JARDÍN20</b>	<b>Cambiar contraseña</b>
Actores	Administrador, usuario registrado
Descripción	Se editará el campo “contraseña” del usuario
Precondición	El usuario debe ser poseedor de la cuenta y previamente haber iniciado sesión
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario accede a su perfil</li> <li>2. Selecciona la opción “cambiar contraseña”</li> <li>3. Selecciona el campo que desea modificar</li> <li>4. Introduce un nuevo valor y lo confirma</li> <li>5. El sistema avisa de la actualización de la contraseña de usuario</li> </ol>
Postcondición	La contraseña queda actualizada con su nuevo valor introducido
Excepciones	<ul style="list-style-type: none"> <li>- Paso 4: que la contraseña a la hora de confirmar no coincida o que los valores no sean válidos</li> </ul>
Comentarios	El valor a introducir no debe contener ningún carácter no válido

<b>JARDÍN21</b>	<b>Ver historial</b>
Actores	Usuario registrado
Descripción	Muestra el historial de itinerarios realizados por el usuario
Precondición	El usuario debe haber iniciado sesión
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario accede a su perfil</li> <li>2. Va a la sección correspondiente donde se muestra el historial</li> </ol>
Postcondición	Se muestra el historial de los itinerarios realizadas por el usuario
Excepciones	<ul style="list-style-type: none"> <li>- Paso 2: el usuario no ha realizado ningún itinerario. Se indica debidamente</li> </ul>

<b>JARDÍN22</b>	<b>Ver favoritos</b>
Actores	Usuario registrado
Descripción	Muestra una lista con los itinerarios favoritos del usuario
Precondición	El usuario debe haber iniciado sesión
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario accede a su perfil</li> <li>2. Accede a la sección del mismo donde se muestran</li> </ol>
Postcondición	Se muestra la lista con los itinerarios favoritos del usuario
Excepciones	<ul style="list-style-type: none"> <li>- Paso 2: el usuario no tiene itinerarios en favoritos. Se muestra un mensaje informando de ello</li> </ul>

<b>JARDÍN23</b>	<b>Añadir a favoritos</b>
Actores	Usuario registrado
Descripción	Añade un itinerario a la lista de itinerarios favoritos del usuario
Precondición	Haber iniciado sesión y que el itinerario exista
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario accede al itinerario que quiera añadir</li> <li>2. Pulsa sobre el botón para añadir el itinerario a favoritos</li> </ol>
Postcondición	Se añade el itinerario a la lista de favoritos del usuario
Excepciones	<ul style="list-style-type: none"> <li>- Paso 1: el itinerario no existe. Se informa del error</li> <li>- Paso 2: el itinerario ya está en favoritos. En su lugar se mostrará un botón para eliminarla de sus favoritos</li> </ul>

<b>JARDÍN24</b>	<b>Quitar de favoritos</b>
Actores	Usuario registrado
Descripción	Elimina un itinerario de la lista de itinerarios favoritos del usuario
Precondición	Haber iniciado sesión y que el itinerario exista
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario accede al itinerario que quiere quitar de favoritos</li> <li>2. Pulsa sobre el botón para eliminar el itinerario de sus favoritos</li> </ol>
Postcondición	Se elimina el itinerario de la lista de favoritos del usuario
Excepciones	<ul style="list-style-type: none"> <li>- Paso 1: el itinerario no existe. Se informa del error</li> <li>- Paso 2: el itinerario no está en favoritos. En su lugar se mostrará un botón para añadirlo a sus favoritos</li> </ul>

<b>JARDÍN25</b>	<b>Ver itinerarios creados</b>
Actores	Usuario registrado
Descripción	Muestra una lista con los itinerarios creados por el usuario
Precondición	Que el usuario haya iniciado sesión
Secuencia normal	<ol style="list-style-type: none"> <li>1. El usuario accede a su perfil</li> <li>2. Se coloca en la sección del mismo donde se muestran</li> </ol>
Postcondición	Se muestra la lista de itinerarios creados por el usuario
Excepciones	<ul style="list-style-type: none"> <li>- Paso 2: el usuario no ha creado itinerarios. Se muestra un mensaje informando de ello</li> </ul>

## Capítulo 5: Arquitectura de la aplicación y modelo de datos

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

### 5.1 Arquitectura de la aplicación

En este sistema se ha utilizado una arquitectura cliente-servidor.

La arquitectura cliente-servidor [28] es un modelo de diseño de software en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta.

En general, un servicio es una abstracción de una serie de recursos. El cliente para utilizarlos solo debe conocer el formato y contenido de la respuesta del servicio solicitado.

En la arquitectura de la aplicación se ha decidido implementar en el servidor una interfaz de programación de aplicaciones (API) [29]. La API es una capa de abstracción para acceder a un servicio. Lo que se consigue restringiendo a un formato específico la comunicación cliente-servidor es una mayor facilidad en el proceso de parseo del contenido. Además, al abstraer el acceso, se facilita el intercambio de información entre múltiples plataformas.

En la figura 4 se muestra el diseño de la arquitectura cliente-servidor del sistema. El navegador ejecuta la parte cliente de la aplicación, código HTML y TypeScript obtenido de la compilación de los distintos componentes Angular que forman la aplicación. El cliente realiza peticiones a través de HTTP a los servicios que expone la API REST, la cual sin conocimiento del cliente, procesa la petición y recupera los datos necesarios de la base de datos. Finalmente, el cliente recibe la respuesta del servicio.

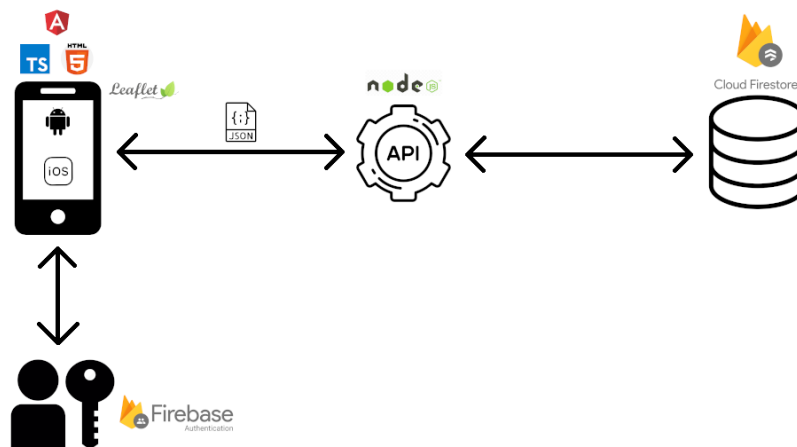


Figura 4: Arquitectura del sistema

#### 5.1.1 Elementos de la arquitectura

En este apartado se explican los distintos elementos que componen la arquitectura del sistema.

##### 5.1.1.1 Aplicación móvil

Esta parte de la arquitectura compone el denominado lado cliente. Se trata de una aplicación para dispositivos móviles desarrollada con Ionic [5] mediante Angular [6] y HTML [8]. Esta aplicación, además, utiliza Leaflet [15] para mostrar el mapa del Real Jardín Botánico.

#### **5.1.1.2 Autenticación**

El servicio de autenticación de los usuarios en la aplicación se realiza mediante el uso de Firebase Authentication [30]. La llamada a este servicio se realiza directamente desde el cliente.

#### **5.1.1.3 API REST**

El cliente se comunica con la base de datos mediante peticiones HTTP a esta API REST [29] implementada con NodeJS [18]. Este servicio se encarga de gestionar las peticiones, solicitar los datos a la base de datos y devolverlos al cliente.

#### **5.1.1.4 Base de datos**

La base de datos del sistema se encuentra almacenada en la nube de Firestore [13].

## 5.2 Modelo de datos

En este apartado se va a describir la forma en la que se ha realizado la persistencia de la información del sistema creado.

Se ha empleado como sistema de persistencia de datos Firestore [13], una base de datos no relacional de tipo documental. El esquema general de la base de datos se puede ver en la figura 5. En dicha figura cada cuadrado representa un documento de la base de datos y las flechas apuntan a los documentos que ellos mismos contienen.

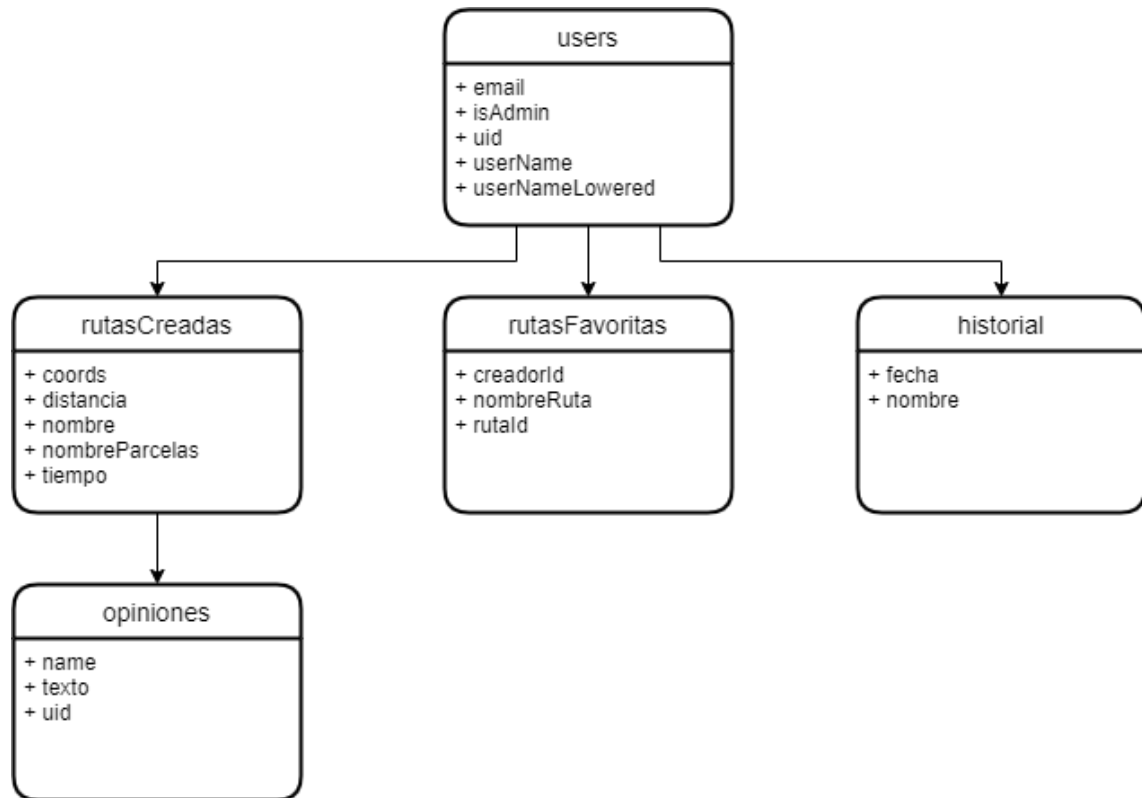


Figura 5: Diagrama ER de la BBDD

Esta base de datos incluye una colección principal, *users*, como se ve en la figura 6.

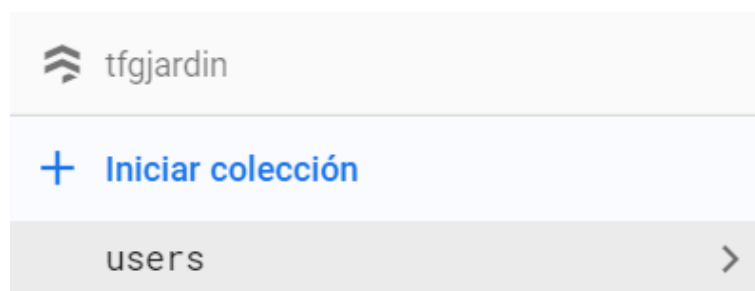


Figura 6: Colecciones de la BBDD

Esta colección almacena los datos referidos a los usuarios de la aplicación. Los campos detallados de la colección son los siguientes:

- email: correo electrónico del usuario.
- isAdmin: booleano que indica si el usuario es de tipo administrador o no.
- uid: id del usuario en Firebase.

- userName: nombre de usuario que se muestra en la aplicación.
- userNameLowered: nombre de usuario en minúsculas que se muestra en la aplicación.

#### + Agregar campo

```
email: "prueba@gmail.com"
isAdmin: true
uid: "████████████████████████████████████████"
userName: "Prueba"
userNameLowered: "prueba"
```

Figura 7: Campos de la colección *users*

No obstante, para gestionar los diversos elementos de la aplicación esta colección presenta a su vez dos subcolecciones en base a la actividad del usuario en la misma. Estas subcolecciones son las siguientes:

- rutasCreadas: subcolección que almacena los itinerarios creados por el usuario. Sus campos concretos son los siguientes:
  - coords: array con las coordenadas del centro de las parcelas que componen el itinerario.
  - distancia: distancia en kilómetros del itinerario.
  - nombre: nombre del itinerario.
  - nombreParcelas: array con el nombre de cada parcela que compone el itinerario.
  - tiempo: tiempo en minutos que se tarda en completar el itinerario.

#### + Agregar campo

```
▶ coords: [{lat: 40.41237, lng: -3.6...}
distancia: 0.38
nombre: "Itinerario de prueba"
▶ nombreParcelas: ["C.6", "E.8", "E.7", "E.7..."]
tiempo: 5
```

Figura 8: Campos de la subcolección *rutasCreadas*

Esta subcolección presenta, a su vez, una subcolección más, *opiniones*, que se describe a continuación:

- name: nombre del usuario que ha creado la opinión
- texto: texto de la opinión.

- uid: id en Firebase del usuario que ha creado la opinión.

### + Agregar campo

name: "Hola3"

texto: "Nueva opinión"

uid: "██"

Figura 9: Campos de la subcolección *opiniones*

- rutasFavoritas: subcolección que almacena los itinerarios favoritos del usuario. Sus campos concretos son los siguientes:

- creadorId: id en Firestore del usuario que ha creado el itinerario.
- nombreRuta: nombre del itinerario.
- rutaId: id en Firestore del itinerario.

### + Agregar campo

creadorId: "██"

nombreRuta: "Itinerario de prueba"

rutaId: "██"

Figura 10: Campos de la subcolección *rutasFavoritas*

- historial: subcolección que almacena los itinerarios realizados por el usuario. Sus campos concretos son los siguientes:

- fecha: día y hora en el que se comenzó el itinerario.
- nombre: nombre del itinerario realizado.

### + Agregar campo

fecha: "26 May, 2021, 10:34:57 AM"

nombre: "Mi itinerario"

Figura 11: Campos de la subcolección *historial*

## Capítulo 6: Diseño de la aplicación

En este capítulo se tratará el diseño y la implementación de las funcionalidades más significativas de la aplicación, comenzando por la aplicación móvil y terminando con la API implementada.

Los usuarios finales de esta aplicación engloban todo el espectro de personas, con todo tipo de experiencia en aplicaciones móviles. Por este motivo, el propósito ha sido hacer un diseño atractivo y simple para que el público en general encuentre todas las funcionalidades a simple vista.

Como se comentó en el capítulo tres, que trataba sobre las tecnologías utilizadas, para la implementación de la capa de presentación se ha utilizado Angular. Entre las razones principales de utilizar este framework de desarrollo se encuentran:

- La posibilidad de reducir el número de peticiones al servidor gracias a la ejecución de parte del código en el lado del cliente.
- Poder escribir código modular, reutilizable y seguir los principios básicos del desarrollo de software.
- La facilidad de comunicar la capa de presentación con la API que se ha desarrollado a través de los mecanismos que ofrece Angular.

### 6.1 Colores y tipografía

Los colores y la tipografía utilizados han sido los que se han considerado más amigables con el usuario. Al tratarse de una aplicación cuyos potenciales usuarios son todas aquellas personas que deseen visitar las instalaciones del Real Jardín Botánico de Madrid, se ha procurado ofrecer una sensación de sencillez en el uso de la aplicación utilizando tonos verdes para familiarizar al usuario con la naturaleza del Real Jardín Botánico de Madrid. En la figura 12 se muestran los códigos de color utilizados.

```
src > app > pages > routes-created > routes-created.page.scss > ...
1  ion-back-button {
2    margin-top: 10px;
3    background-color: rgba(3, 87, 16, 0.685);
4    color: white;
5    border-radius: 50%;
6    margin-bottom: 10px;
7  }
8
9  .itinerario_creado {
10   margin-top: 20px;
11   text-align: center;
12 }
```

Figura 12: Fragmento de código - Códigos de color utilizados en la aplicación móvil

### 6.2. Funcionalidad de la aplicación

En este apartado se describen las funcionalidades implementadas y las pantallas asociadas más representativas de la aplicación.

#### 6.2.1 Vista principal

##### 6.2.1.1 Home

Home es la primera vista a la que accede el usuario tras abrir la aplicación móvil. En la figura 13 se puede ver esta vista en los diferentes temas, oscuro y claro, de un usuario sin autenticar. Sin embargo, si el usuario se autentica, se mostrará por un lado, en la parte superior, su nombre de usuario acompañado del texto "Bienvenido" y por otro se añadirán funcionalidades como se muestra en la figura 14.

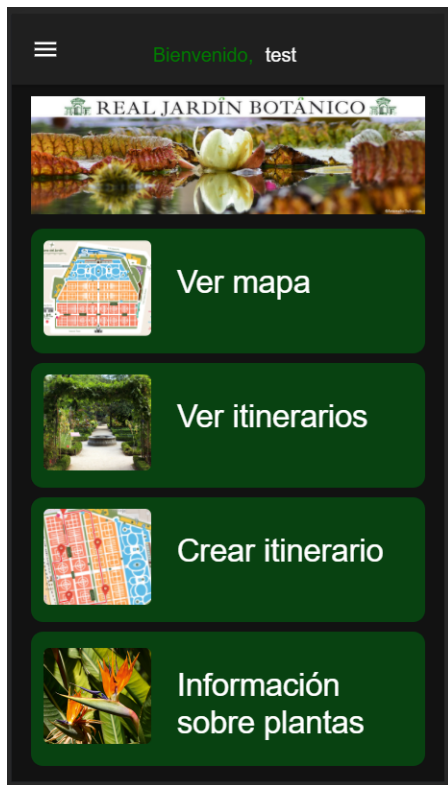


(a) Tema oscuro



(b) Tema claro

Figura 13: Vista de Home sin autenticación del usuario



(a) Tema oscuro



(b) Tema claro

Figura 14: Vista de Home con autenticación del usuario

En la figura 15 se puede ver el código html empleado.

```


<div class="btn-home" (click)="navigateURL('map')">
  <div class="home-35">
    
  </div>
  <div class="home-65">
    <h1>Ver mapa</h1>
  </div>
</div>
<div class="btn-home" (click)="navigateURL('itineraries')">
  <div class="home-35">
    
  </div>
  <div class="home-65">
    <h1>Ver itinerarios</h1>
  </div>
</div>
<div *ngIf="isLoggedIn()">
  <div class="btn-home" (click)="navigateURL('itinerary-maker')">
    <div class="home-35">
      
    </div>
    <div class="home-65">
      <h1>Crear itinerario</h1>
    </div>
  </div>
</div>
<div class="btn-home" (click)="navigateURL('plants-info')">
  <div class="home-35">
    
  </div>
  <div class="home-65">
    <h1>Información sobre plantas</h1>
  </div>
</div>

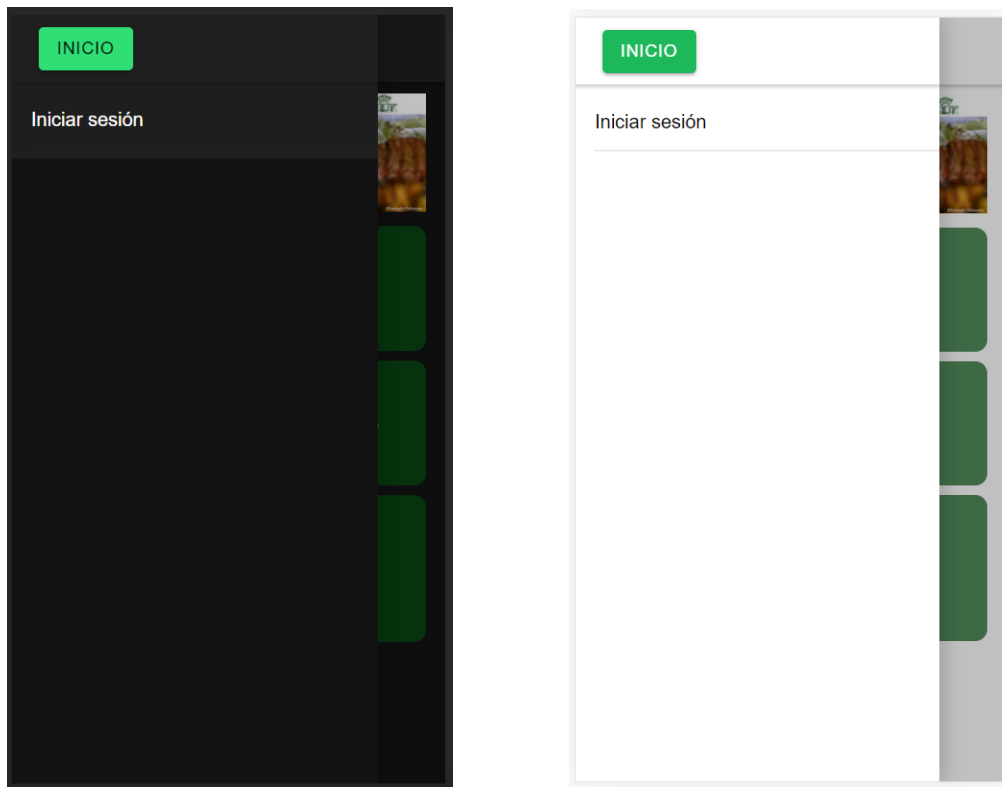
```

Figura 15: Fragmento de código - Código de la vista de Home

### 6.2.1.1 Menú lateral

En el menú lateral los usuarios disponen de un botón "INICIO" mediante el cual, en cualquier momento, pueden volver a la vista principal "Home". En caso de no estar autenticados pueden

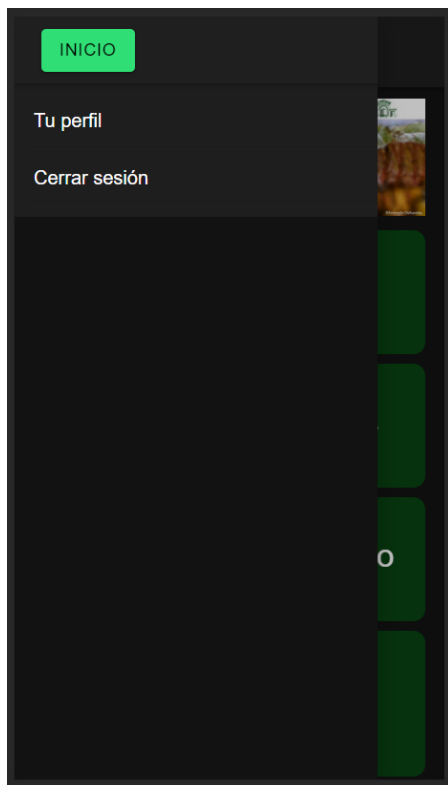
hacerlo mediante la opción de "Iniciar sesión" como se muestra en la figura 16. Una vez los usuarios se han autenticado dispondrán de dos apartados nuevos como se muestra en la figura 17: "Tu perfil" mediante el cuál pueden acceder a su perfil y la opción "Cerrar sesión" para dejar de estar autenticados en la aplicación.



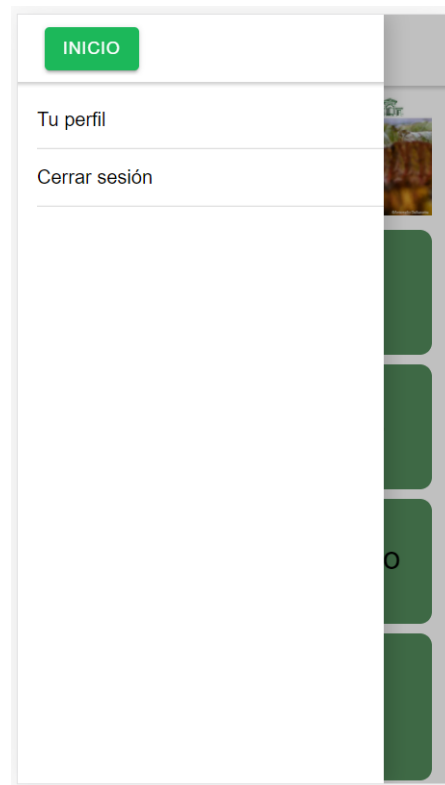
(a) Tema oscuro

(b) Tema claro

Figura 16: Vista del menú lateral sin autenticación del usuario



(a) Tema oscuro



(b) Tema claro

Figura 17: Vista del menú lateral con autenticación del usuario

En la figura 18 se puede ver el código html empleado.

```

<ion-menu side="start" menuId="first" contentId="main">
  <ion-header>
    <ion-toolbar>
      <ion-title>
        <ion-button color="success" (click)="navigateURL('home')">Inicio</ion-button>
      </ion-title>
    </ion-toolbar>
  </ion-header>
  <ion-content>
    <ion-list>
      <ion-item *ngIf="loggedIn" (click)="navigateURL('profile')">Tu perfil</ion-item>
      <ion-item *ngIf="loggedIn" (click)="logout()">Cerrar sesión</ion-item>
      <ion-item *ngIf="!loggedIn" (click)="navigateURL('login')">Iniciar sesión</ion-item>
    </ion-list>
  </ion-content>
</ion-menu>

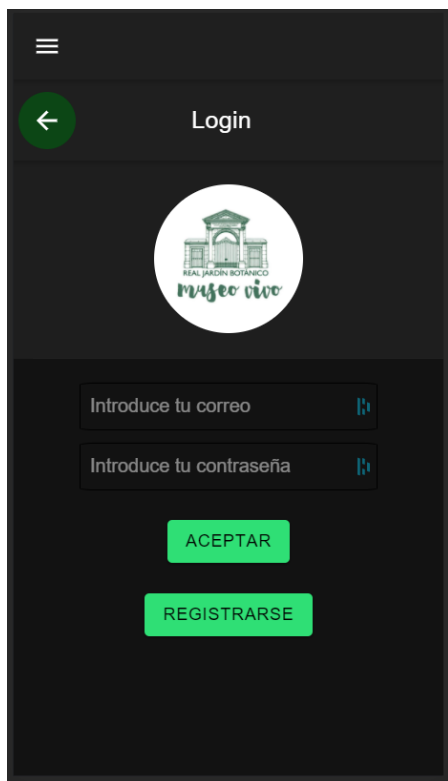
```

Figura 18: Fragmento de código - Código del menú lateral

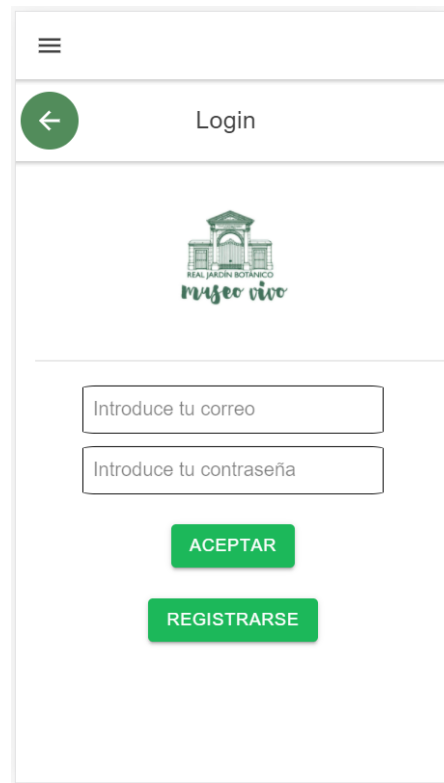
## 6.2.2 Vistas de usuario

### 6.2.2.1 Autenticación

El sistema de autenticación es uno de los pilares de la aplicación, dado que de este depende la identificación de los usuarios. En figura 19 se muestra una captura de la pantalla de "Login". Como se puede observar, se ha buscado continuar con el mismo estilo de diseño que en el resto de la aplicación.



(a) Tema oscuro



(b) Tema claro

Figura 19: Vista inicio de sesión

Una vez el usuario introduce sus credenciales, se realiza una petición a la API con los datos. En caso de que sean correctos, el usuario queda autenticado en la aplicación, en caso de error se informa al usuario de que no pudo iniciar sesión. En la figura 20 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

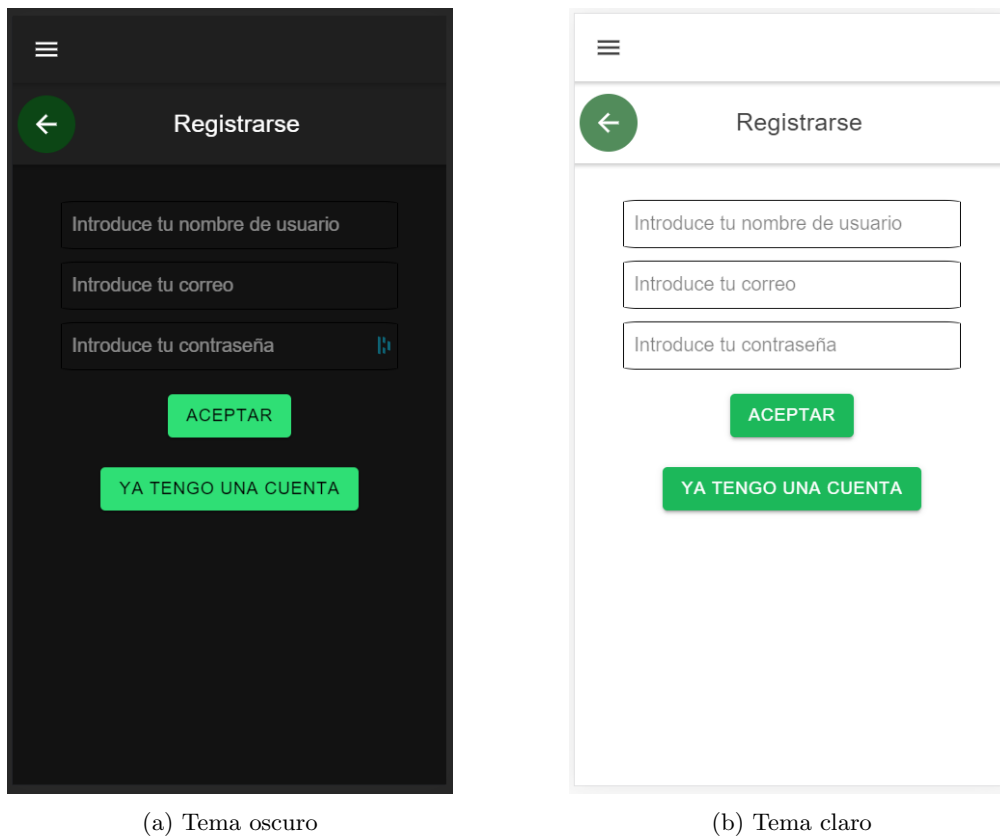
async login(email: string, password: string): Promise<user> {
  try {
    const { user } = await this.authService.signInWithEmailAndPassword(email, password);
    this.isLoggedIn = true;
    this.isAdmin = await this.isUserAdmin(user.email);
    this.userEmail = email;
    this.setUser(email);
  } catch (error) {
    return user;
  }
  switch(error.code){
    case 'auth/invalid-email': this.alertService.showAlert("Dirección de correo no válida.");then(() => { }); break;
    case 'auth/user-not-found': this.alertService.showAlert("Usuario o contraseña incorrectos.");then(() => { }); break;
    case 'auth/wrong-password': this.alertService.showAlert("Usuario o contraseña incorrectos.");then(() => { }); break;
    case 'auth/too-many-requests': this.alertService.showAlert("Demasiados intentos intentando iniciar sesión, la cuenta ha sido suspendida temporalmente.");then(() => { }); break;
    default: this.alertService.showAlert(error).then(() => { });
  }
}
}

```

Figura 20: Fragmento de código - Código de autenticación de usuario

### 6.2.2.2 Registrarse

Dado que la aplicación cuenta con más funcionalidades para usuarios que disponen de una cuenta es importante contar con un registro para que así los usuarios puedan crearse una cuenta, autenticarse y poder acceder a dichas funcionalidades. En figura 21 se muestra una captura de la pantalla de "Registrarse". Como se puede observar, se ha buscado continuar con el mismo estilo de diseño que en el resto de la aplicación.



(a) Tema oscuro

(b) Tema claro

Figura 21: Vista de registrar nuevo usuario

Una vez el usuario introduce sus datos, en caso de que sean correctos, el usuario queda dado de alta y autenticado en la aplicación, en caso de error se informa al usuario de que los datos introducidos no son correctos. En la figura 22 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

this.afs.collection('users').add({
  uid : user.uid,
  email : user.email,
  isAdmin : false,
  userName : usuario,
  userNameLowered : usuario.toLowerCase()
});

```

Figura 22: Fragmento de código - Código de registro de usuario

### 6.2.2.3 Cerrar sesión

La funcionalidad de cerrar sesión (figura 23) no requiere comunicación directa con la API. Para hacerla efectiva se borra el almacenamiento local donde se guardan los dos tipos de token y se le redirige al usuario a la vista principal de la aplicación.

```

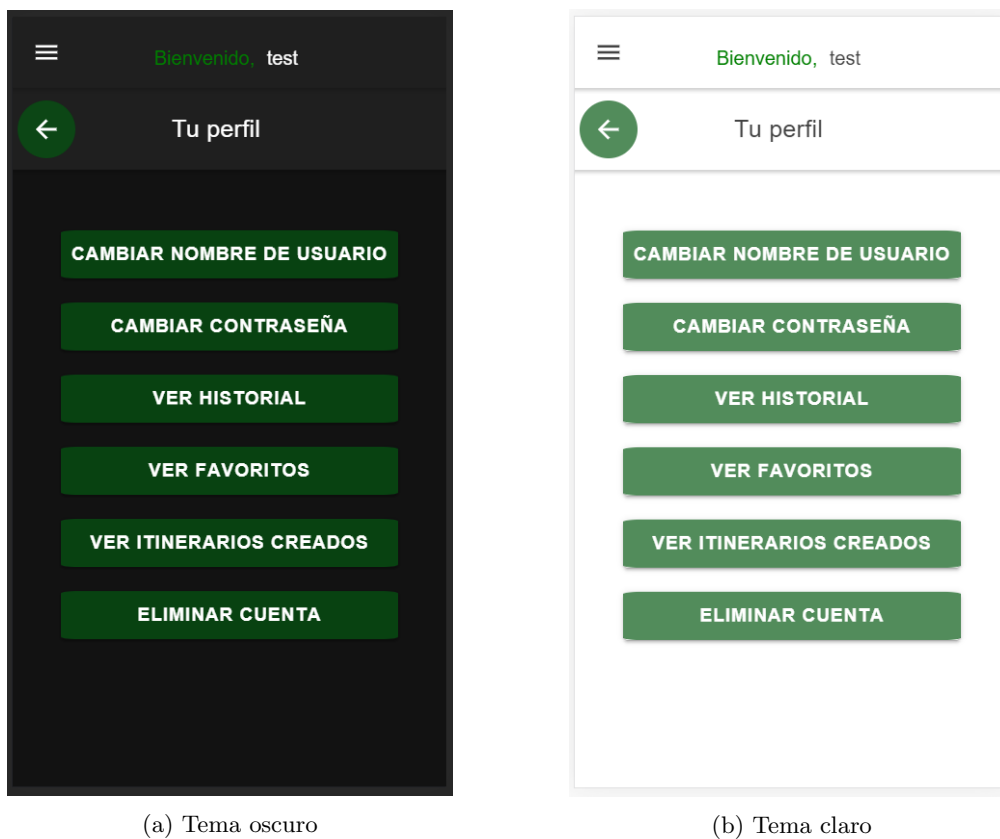
async logout(): Promise<void> {
  try {
    await this.afAuth.signOut();
    this.isLogged = false;
    this.isAdmin = false;
    this.userName = "";
    this.userEmail = "";
    this.name.next("");
    this.authenticated.next(false);
  } catch (error) {
    this.alertService.showAlert(error).then(() => { });
  }
}
}

```

Figura 23: Fragmento de código - Cerrar sesión

#### 6.2.2.4 Mi perfil

Una vez autenticado, el usuario puede acceder a su perfil en el cual dispone de varias funcionalidades como se muestra en la figura 24:



(a) Tema oscuro

(b) Tema claro

Figura 24: Vista del perfil de usuario

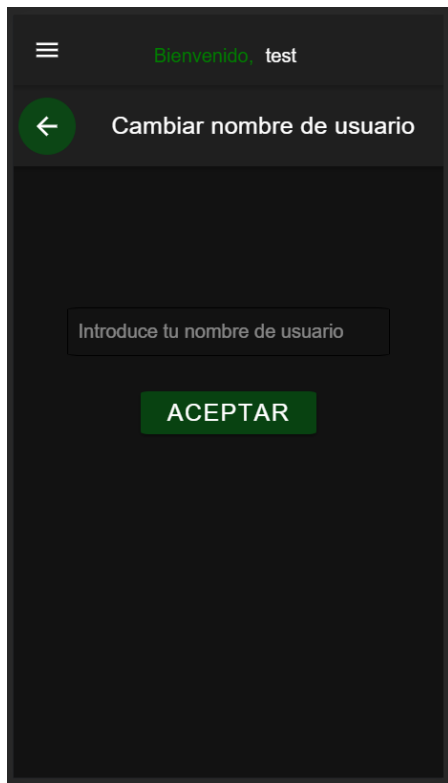
En la figura 25 se puede ver el código html empleado.

```
<ion-header>
  <ion-toolbar>
    <ion-buttons slot="start">
      <ion-back-button defaultHref="/home"></ion-back-button>
    </ion-buttons>
    <ion-title>tu perfil</ion-title>
  </ion-toolbar>
</ion-header>
<ion-content>
  <div class="perfil-content">
    <div class="btn">
      <ion-button class="btn-user" color="tint" (click)="navigateURL('changeUsername')">Cambiar nombre de usuario</ion-button>
    </div>
    <div class="btn">
      <ion-button class="btn-user" color="tint" (click)="navigateURL('changePassword')">Cambiar contraseña</ion-button>
    </div>
    <div *ngIf="!this.auth.isAdmin">
      <div class="btn">
        <ion-button class="btn-user" color="tint" (click)="navigateURL('record')">Ver historial</ion-button>
      </div>
      <div class="btn">
        <ion-button class="btn-user" color="tint" (click)="navigateURL('favorites')">Ver favoritos</ion-button>
      </div>
      <div class="btn">
        <ion-button class="btn-user" color="tint" (click)="navigateURL('routesCreated')">Ver itinerarios creados</ion-button>
      </div>
      <div class="btn">
        <ion-button class="btn-user" color="tint" (click)="eliminarCuenta()">Eliminar cuenta</ion-button>
      </div>
    </div>
  </div>
</ion-content>
```

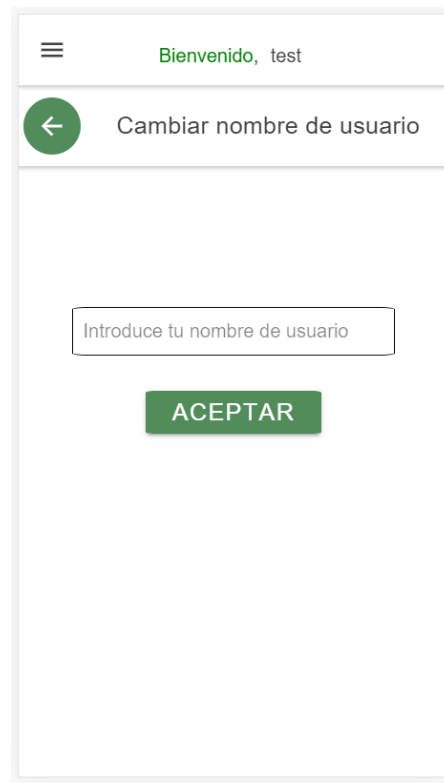
Figura 25: Fragmento de código - Código del perfil de usuario

### 6.2.2.5 Cambiar nombre de usuario

En esta vista que se muestra en la figura 26, los usuarios pueden cambiar su nombre de usuario si así lo desean, solo necesita introducir el nuevo nombre elegido y dar a aceptar.



(a) Tema oscuro



(b) Tema claro

Figura 26: Vista de cambiar nombre de usuario

En la figura 27 se puede ver el código html empleado.

```

<ion-header>
  <ion-toolbar>
    <ion-buttons slot="start">
      <ion-back-button (click)="navigateTo('/profile')"></ion-back-button>
    </ion-buttons>
    <ion-title>Cambiar nombre de usuario</ion-title>
  </ion-toolbar>
</ion-header>

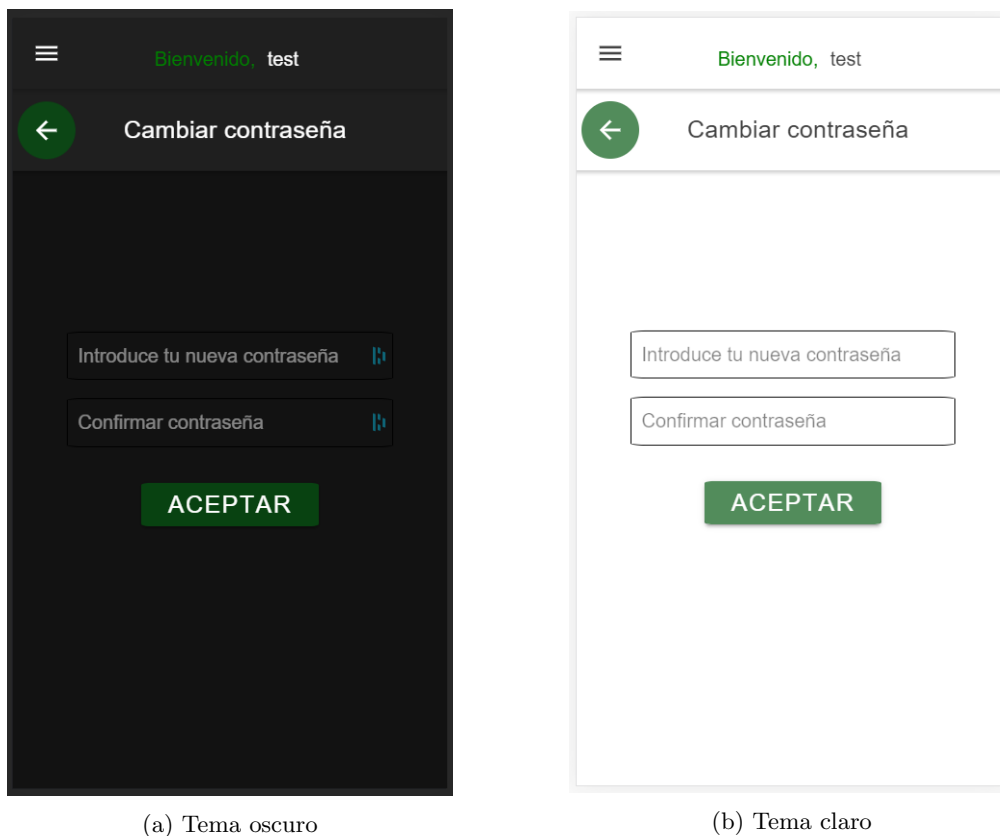
<ion-content>
  <div class="content">
    <br>
    <div class="text-input">
      <ion-input type="text" #usuario required placeholder="Introduce tu nombre de usuario"></ion-input>
    </div>
    <div class="btn-changen">
      <ion-button class="aceptar" color="tint" (click)="changeName(usuario)">Aceptar</ion-button>
    </div>
  </div>
</ion-content>

```

Figura 27: Fragmento de código - Código de cambiar nombre de usuario

### 6.2.2.6 Cambiar contraseña

En esta vista el usuario puede cambiar su contraseña ya sea porque la seguridad de su cuenta ha sido vulnerada o simplemente desea cambiar su contraseña por motivos ajenos a la seguridad, para ello solo tiene que introducir una nueva contraseña y confirmarla, como se muestra en la figura 28.



(a) Tema oscuro

(b) Tema claro

Figura 28: Vista de cambiar contraseña del usuario

En la figura 29 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

var user = this.afAuth.currentUser;
(await user).updatePassword(newPass).then(()=>{});
this.alertService.showAlert("Contraseña actualizada correctamente").then(()=>{});
return true;
} catch(error){
  this.alertService.showAlert(error).then(() => { });
}

```

Figura 29: Fragmento de código - Código de cambiar contraseña

### 6.2.2.7 Ver historial

En esta vista el usuario puede consultar todos los itinerarios que ha recorrido, como se muestra en la figura 28.

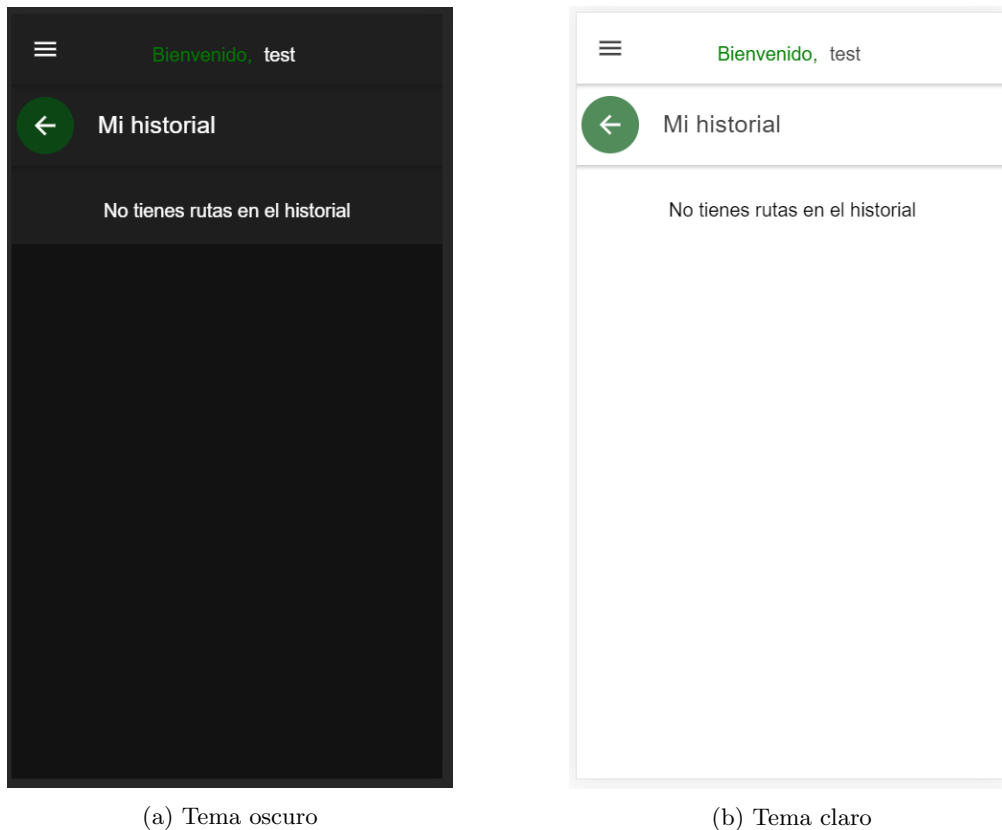


Figura 30: Vista de ver historial

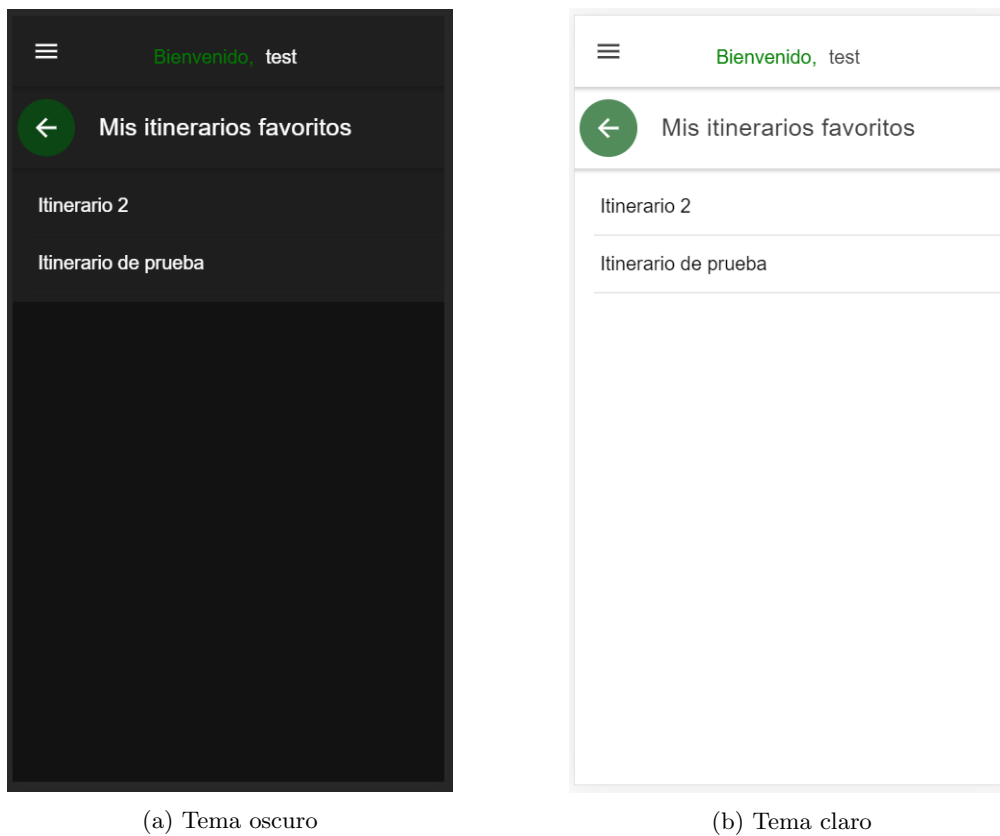
Para que el usuario pueda ver su historial, se realiza una petición a la API que devuelve todos los itinerarios que el usuario ha realizado, En la figura 31 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```
async fetchHistorial(email){  
  return await this.http.post<any[]>(`${environment.apiUrl}/getHistorialList`, {email: email}).toPromise();  
}
```

Figura 31: Fragmento de código - Código del historial

### 6.2.2.8 Ver favoritos del usuario

En esta vista el usuario puede consultar todos sus itinerarios favoritos, como se muestra en la figura 28.



(a) Tema oscuro

(b) Tema claro

Figura 32: Vista de ver favoritos

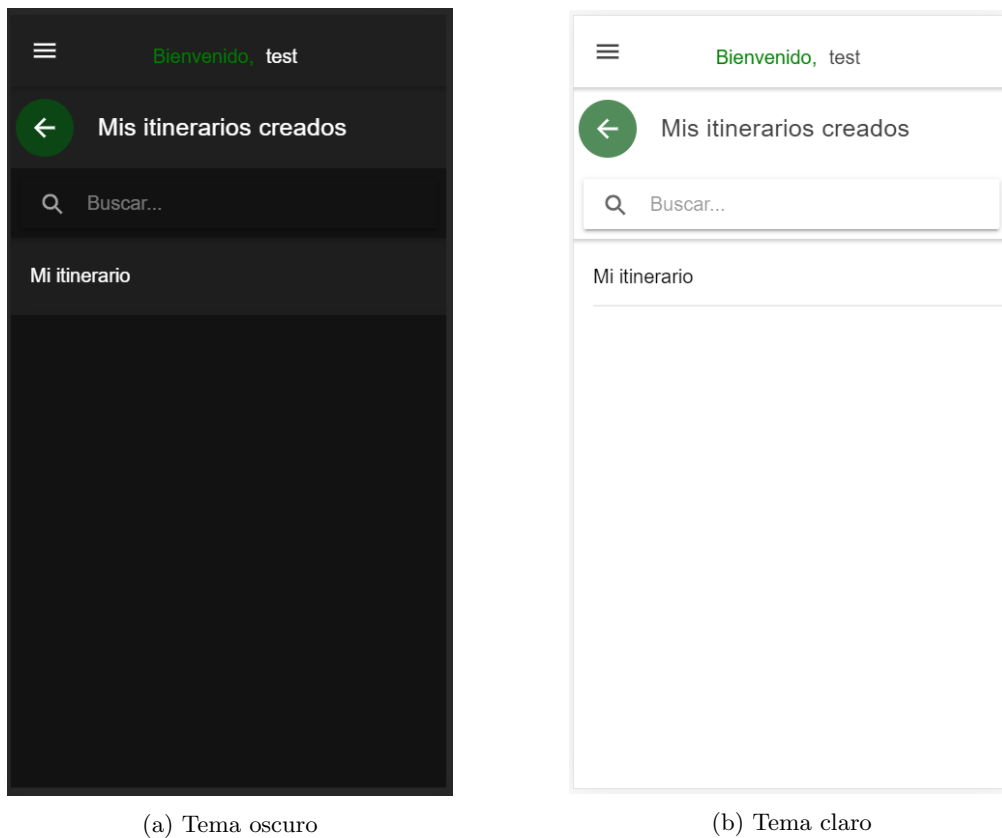
Para que el usuario pueda ver sus favoritos, se realiza una petición a la API que devuelve todos los itinerarios favoritos del usuario, En la figura 33 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```
async fetchFavorites(uid){
  return await this.http.post<any[]>(`${environment.apiUrl}/getFavorites`, {userId:uid}).toPromise();
}
```

Figura 33: Fragmento de código - Código del historial

#### 6.2.2.9 Ver itinerarios creados por el usuario

En esta vista el usuario puede consultar todos sus itinerarios creados, como se muestra en la figura 28.



(a) Tema oscuro

(b) Tema claro

Figura 34: Vista de itinerarios creados por el usuario

Para que el usuario pueda ver sus itinerarios creados, se realiza una petición a la API que devuelve todos los itinerarios creados del usuario, En la figura 35 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

public getRutasUsuario(){
    return this.datasrv.getRutasUsuario(this.authsrv.getUserName());
}

```

Figura 35: Fragmento de código - Código de itinerarios del usuario

#### 6.2.2.10 Eliminar cuenta

El usuario, si así lo desea puede eliminar su cuenta y darse de baja en la aplicación, como se muestra en la figura 36

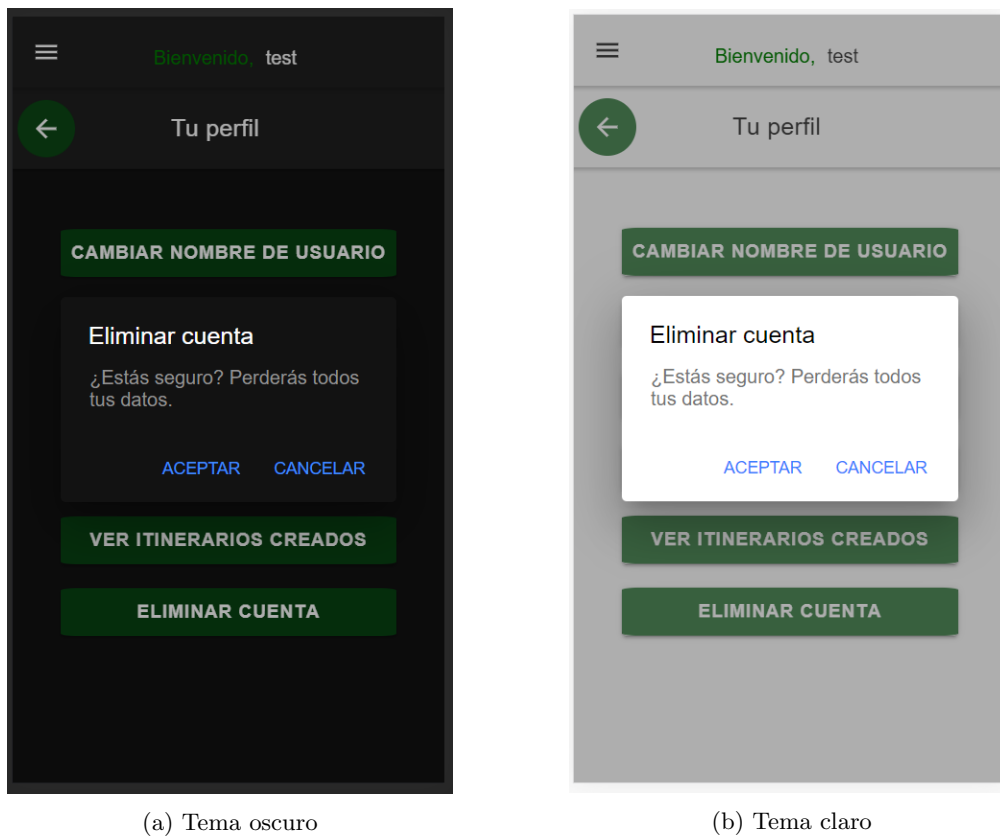


Figura 36: Vista de eliminar cuenta del usuario

Cuando el usuario selecciona la opción de "Eliminar cuenta", se cierra sesión y se realiza una solicitud a la API que se encarga de eliminar todos los datos de dicho usuario de la aplicación. En la figura 37 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```
private async eliminarCuentaHandler(){
  const name = this.auth.getUserName();

  this.auth.logout();
  this.data.deleteUsers(name);

  this.navigateURL('home');

  const alert = await this.alertController.create({
    header: "Cuenta eliminada",
    buttons:[
      {
        text:"Aceptar"
      }
    ]
  });
  await alert.present();
}
```

Figura 37: Fragmento de código - Código de eliminar cuenta de usuario

## 6.2.3 Vistas de itinerarios

### 6.2.3.1 Buscador de itinerarios

En esta vista los usuarios disponen de un buscador mediante el cual puede consultar todos los itinerarios creados por todos los usuarios y acceder a ellos si desean consultar o realizar el

itinerario seleccionado, como se muestra en la figura 38.

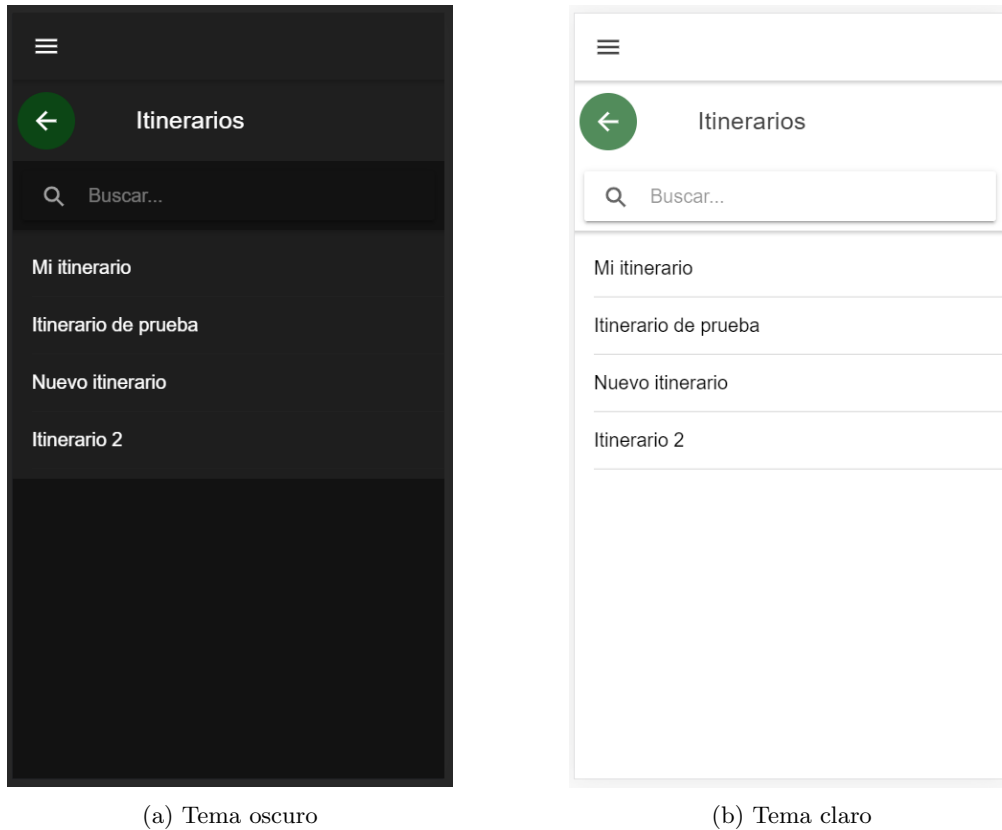


Figura 38: Vista de buscador de itinerarios

En la figura 39 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```
async fetch() {
  this.rutas = await this.http.get<any[]>(`${environment.apiUrl}/getRoutesList`).toPromise();
}

<div class="${this.hasRutas() ? 'no-empty' : 'empty'}">
  <ion-list>
    <div *ngon="let itine of this.getRutas()">
      <div *ngon="let i of itine.rutas | filter:rutas:buqueda">
        <ion-item [(click)="navigateTo('ruta', {nombre:i.data.nombre})]" [(ion-1)comp
        </div>
      </ion-list>
    </div>
  </ion-list>
</div>
<ng-template empty>
  <ion-list>
    <div class="Itinerario_creado"><No hay ningún Itinerario/fin>
  </ion-list>
</ng-template>
</ion-content>
```

Figura 39: Fragmento de código - Código de buscador de itinerarios

### 6.2.3.2 Crear itinerario

En esta vista los usuarios pueden crear un itinerario añadiendo las parcelas que consideren y en caso de equivocación eliminar la parcela seleccionada, para ello solo necesitan señalar las parcelas que desean recorrer, dar un nombre a dicho itinerario y seleccionar "CREAR", como se muestra en la figura 40.



(a) Tema oscuro



(b) Tema claro

Figura 40: Vista crear itinerario

En la figura 41 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

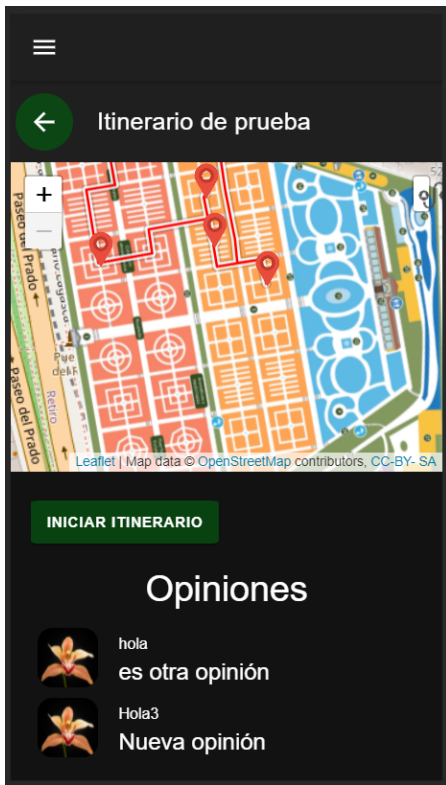
this.dataService.addRuta(nombre, this.checkPointsCoords, this.checkPoints,
this.authService.getUId(), this.authService.getColuid(),
this.authService.getUserName(), distancia, tiempo);

```

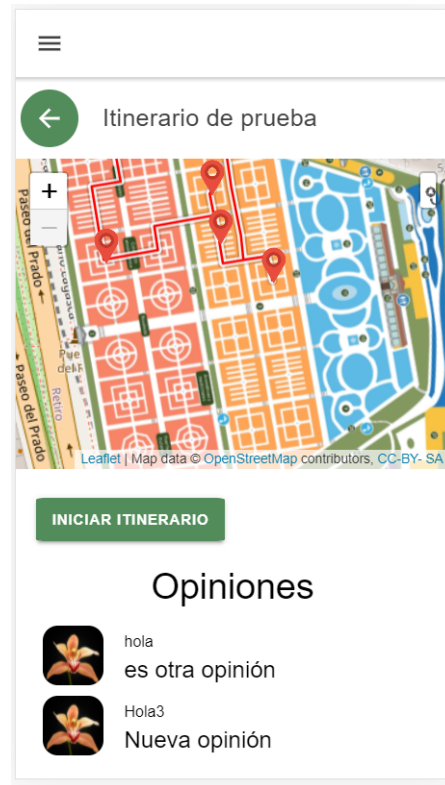
Figura 41: Fragmento de código - Código de crear itinerario

### 6.2.3.3 Ver itinerario

En esta vista los usuarios pueden ver el itinerario seleccionado, en caso de no estar autenticado la vista que tienen los usuarios se muestra en la figura 42, en caso de estar autenticado la vista a la que acceden los usuarios es igual a la que acceden los que no están autenticados salvo que se añades opciones de las que los usuarios pueden hacer uso, como se muestra en la figura 43

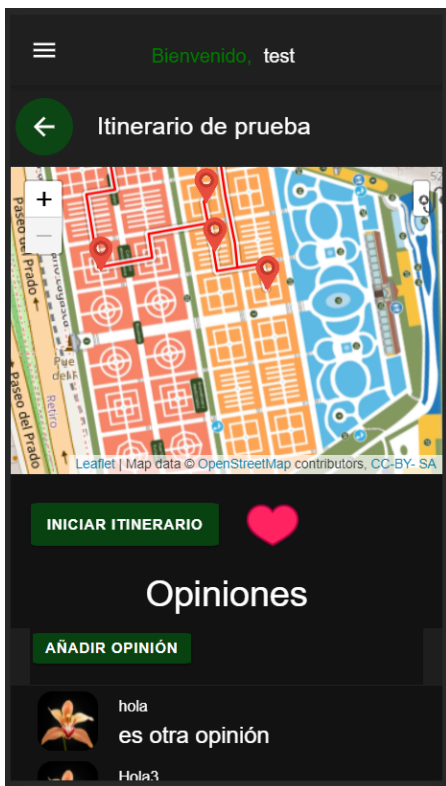


(a) Tema oscuro

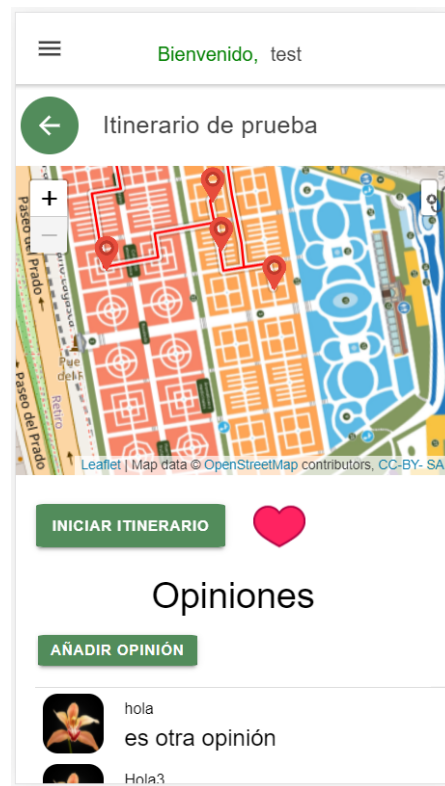


(b) Tema claro

Figura 42: Vista de ver itinerario de usuario no autenticado



(a) Tema oscuro



(b) Tema claro

Figura 43: Vista de ver itinerario de usuario autenticado

En la figura 44 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

<div class="content-it">
  <ion-title *ngIf="ruta">{{ruta.data.nombre}}</ion-title>
</div>
</ion-toolbar>
</ion-header>
<ion-content>
  <div id="mapId" style="width: 100%; height: 50%;></div>
  <div class="scrol" *ngIf="this.exists">
    <div class="content-iter">

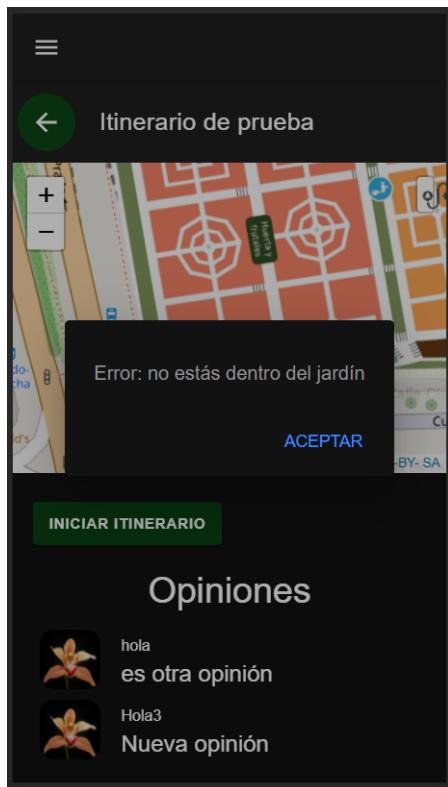
      <div class="col-50">
        <ion-button class="btn-start" color="tint" (click)="this.iniciarRuta(ruta.data.nombre)">Iniciar itinerario</ion-button>
      </div>
      <div>
        <div class="content-heart" *ngIf="isUser();">
          <div class="col-25">
            
          </div>
          <ng-template class="col-25" #noFav>
            
          </ng-template>
          <div class="col-25" *ngIf="this.userEsCreador()">
            
          </div>
        </div>
      </div>
    </div>
  <div class="opinions-content">
    <div class="opinions">

```

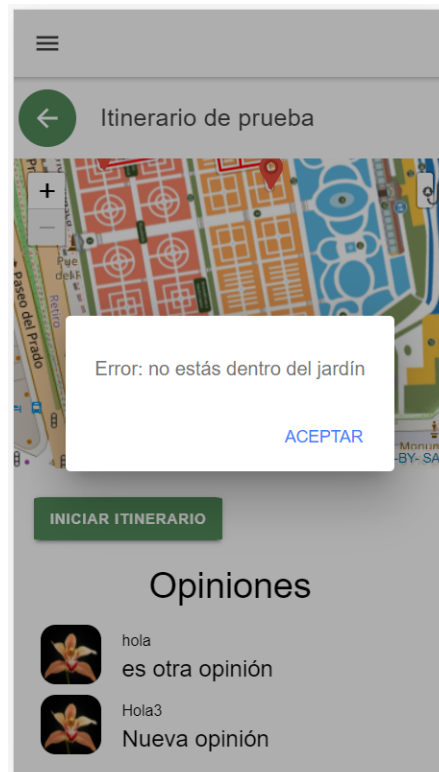
Figura 44: Fragmento de código - Código de ver itinerario

### 6.2.3.4 Iniciar itinerario

Si el usuario se encuentra en el Real Jardín Botánico puede seleccionar la opción de "Iniciar Itinerario" para poder realizar el itinerario seleccionado, en caso de no estar en el Real Jardín Botánico se mostrará un error avisando que no se encuentra dentro del Real Jardín Botánico por lo que no se puede iniciar el itinerarios, como se muestra en la figura 45.



(a) Tema oscuro



(b) Tema claro

Figura 45: Vista de iniciar itinerario

En la figura 46 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

async iniciarRuta(nombreRuta){
  await this.loading.present();
  this.map.locate({
    setView:true,
    watch:true,
    enableHighAccuracy:true
  })
  .on('locationfound', async e => {
    if(!this.map.getBounds().contains(e.latlng)){
      await this.loading.dismiss();
      this.map.stopLocate();

      if(this.marker){
        this.map.removeLayer(this.marker);
      }
      const alert = await this.alertController.create({message:"Error: no estás dentro del jardín", buttons:[{text:"Aceptar"}]});
      await alert.present();

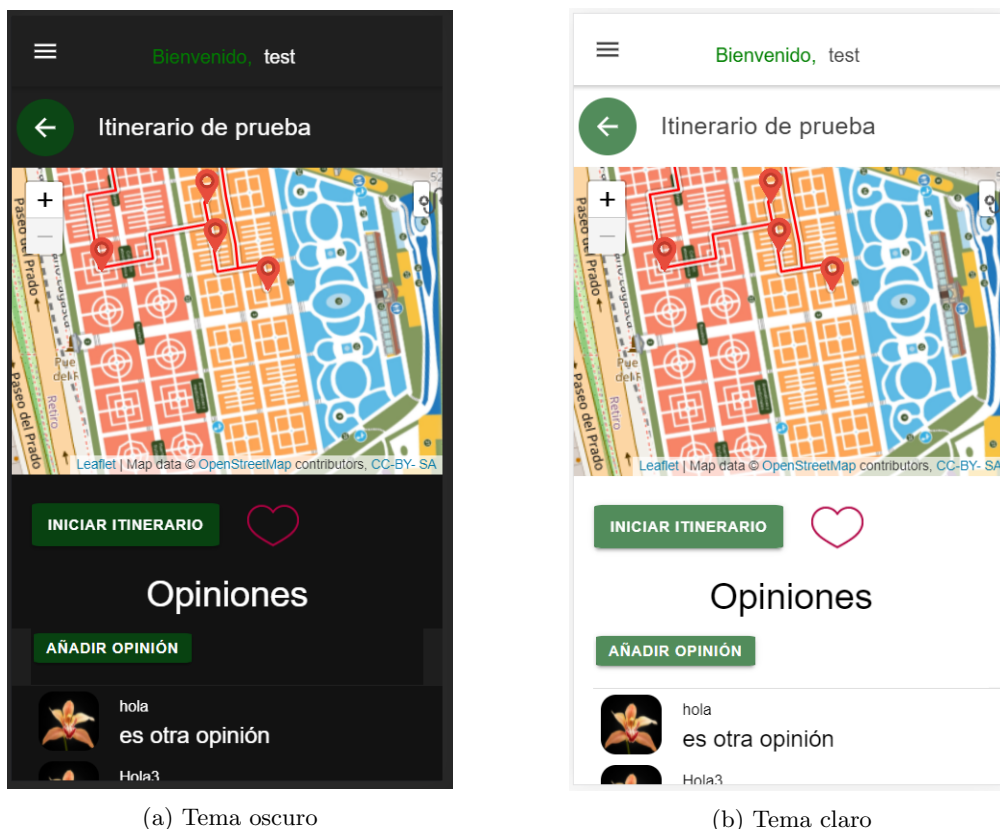
      return;
    }else{
      if(!this.marker){
        this.marker = L.marker(e.latlng).addTo(this.map).bindPopup("Estás aquí").openPopup();
        this.addHistorial(nombreRuta);
      }else{
        this.marker.setLatLng(e.latlng);
      }
    }
  })
  .on('locationerror', async e => {
    await this.loading.dismiss();
    this.map.stopLocate();
    if(this.marker){
      this.map.removeLayer(this.marker);
      this.marker = undefined;
    }
  })
}

```

Figura 46: Fragmento de código - Código de iniciar itinerario

### 6.2.3.5 Añadir itinerario a favorito

Los usuarios disponen de un botón con forma de corazón para añadir un itinerario a favoritos si así lo desean y así poder encontrarla con facilidad sin necesidad de realizar una búsqueda exhaustiva, como se muestra en la figura 47.



(a) Tema oscuro

(b) Tema claro

Figura 47: Vista de añadir itinerario a favorito

En la figura 48 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

async addFav(uid, rid, nombreRuta, creador){
  let aux = await this.http.post<any[]>(`${environment.apiUrl}/addFav`, {uid:uid, rid:rid, nombreRuta:nombreRuta, creador:creador}).toPromise();
  return aux;
}

```

Figura 48: Fragmento de código - Código de añadir itinerario a favorito

### 6.2.3.6 Eliminar itinerario de favorito

Los usuarios disponen de un botón con forma de corazón para eliminar un itinerario de sus favoritos si así lo desean, como se muestra en la figura 49.

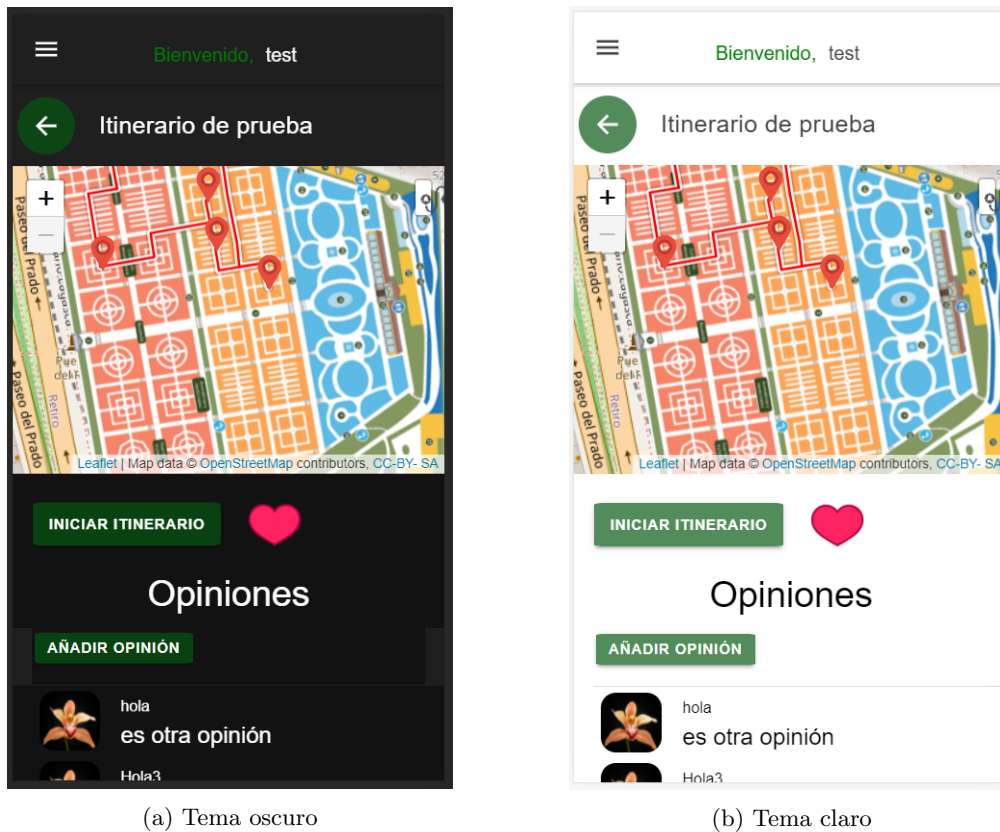


Figura 49: Vista de eliminar itinerario de favorito

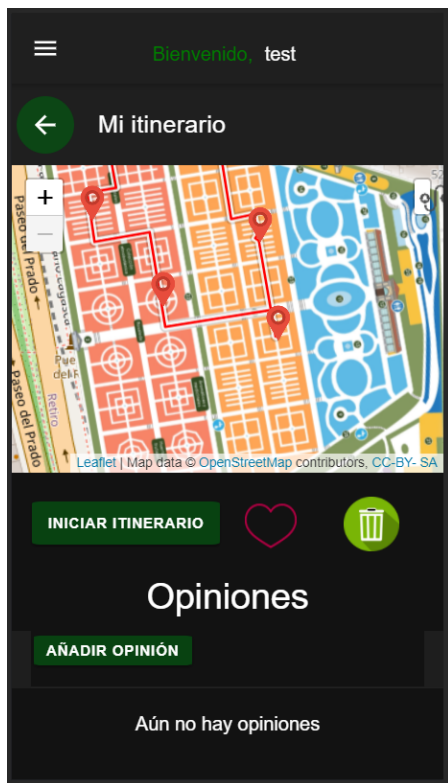
En la figura 50 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```
async deleteFav(uid, rid){
  return await this.http.post<any[]>(`${environment.apiUrl}/deleteFav`, {uid:uid, rid:rid}).toPromise();
}
```

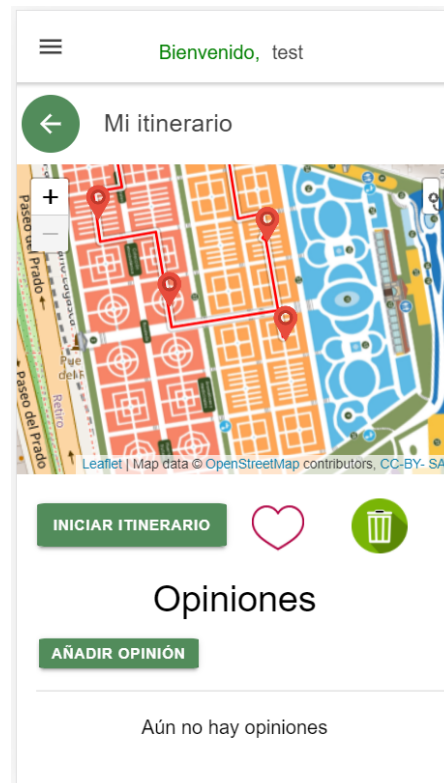
Figura 50: Fragmento de código - Código de eliminar itinerario de favorito

### 6.2.3.7 Eliminar itinerario

El usuario disponen de un botón para poder eliminar un itinerario previamente creado por el mismo, como se muestra en la figura 51, el botón no aparecerá en caso de no ser el creador de dicho itinerario.



(a) Tema oscuro



(b) Tema claro

Figura 51: Vista de eliminar itinerario

En la figura 52 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

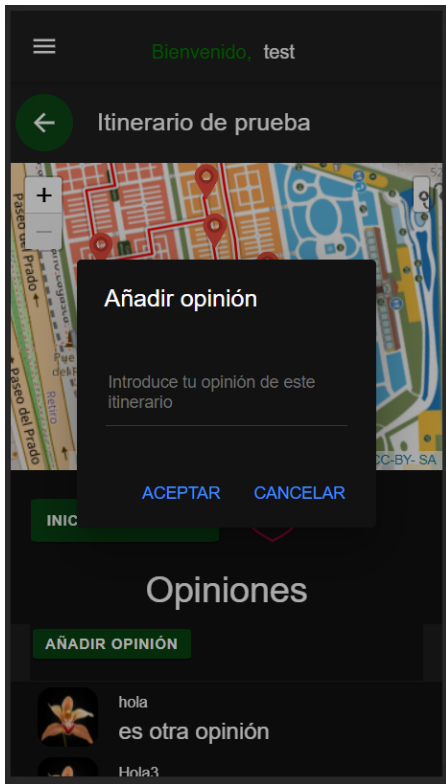
return await this.http.post(`${environment.apiUrl}/deleteRuta`,
{
  uid:ruta.creador,
  rid:ruta.id,
  nombreRuta:ruta.data.nombre
})
).toPromise().then(async () => {
  const alert = await this.alertController.create({
    header:"Itinerario eliminado",
    buttons:[
      {
        text:"Aceptar"
      }
    ]
  });
  await alert.present();
});

```

Figura 52: Fragmento de código - Código de eliminar itinerario

### 6.2.3.8 Añadir/eliminar/modificar opinión

Los usuarios disponen de la opción de añadir opinión en cualquier itinerario ya sea propio o ajeno, como se muestra en la figura 53, en caso de ya haber opinado tiene la opción de modificar o eliminar dicha opinión, como se muestra en la figura 54.

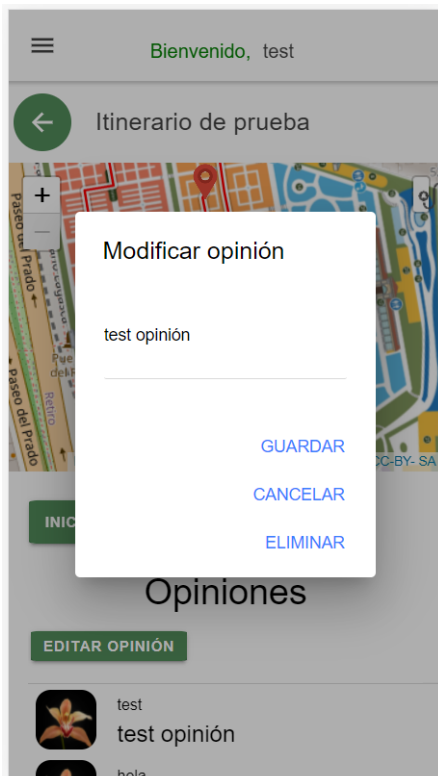


(a) Tema oscuro



(b) Tema claro

Figura 53: Vista de añadir opinión



(a) Tema oscuro



(b) Tema claro

Figura 54: Vista de modificar o eliminar opinión

En la figura 55 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```
async addOpinion(creadorid, rid, uid, userName, texto){
  return await this.http.post(`${environment.apiUrl}/addOpinion`,
    {
      creadorid:creadorid,
      rid:rid,
      uid:uid,
      userName:userName,
      texto:texto
    }
  ).toPromise();
}

async deleteOpinion(cid, rid, uid){
  return await this.http.post(`${environment.apiUrl}/deleteOpinion`,
    {
      creadorid:cid,
      rid:rid,
      uid:uid
    }
  ).toPromise();
}

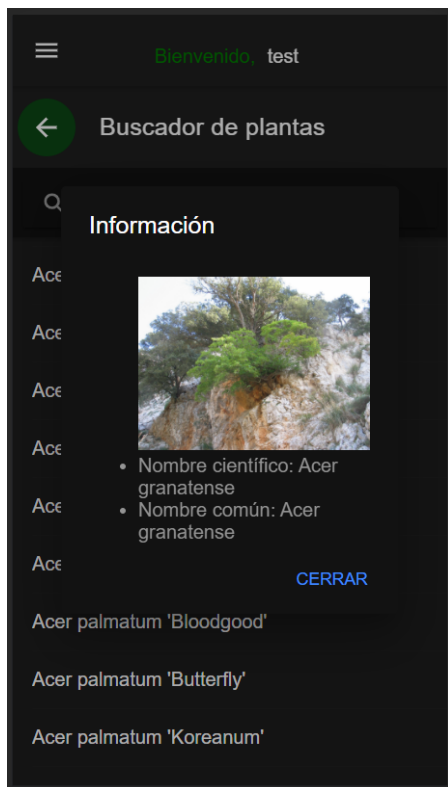
async updateOpinion(cid, rid, uid, texto){
  return await this.http.post(`${environment.apiUrl}/updateOpinion`,
    {
      creadorid:cid,
      rid:rid,
      uid:uid,
      texto:texto
    }
  ).toPromise();
}
```

Figura 55: Fragmento de código - Código de añadir/modificar/eliminar opinión

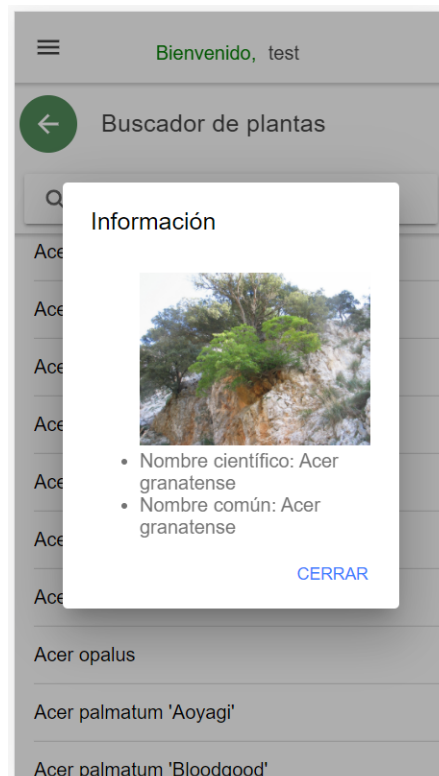
**6.2.4 Vistas de información**

**6.2.4.1 Ver información sobre plantas**

Los usuarios disponen de un buscador el cual te permite filtrar los elementos mostrados. En el que pueden consultar toda la flora de la cual dispone el Real Jardín Botánico, mostrando información como: una imagen del elemento buscado, su nombre científico, nombre común, una pequeña descripción y su ubicación dentro del Real Jardín Botánico, como se muestra en la figura 56.



(a) Tema oscuro



(b) Tema claro

Figura 56: Vista de eliminar cuenta del usuario

En la figura 57 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

<ion-header>
  <ion-toolbar>
    <ion-buttons slot="start">
      <ion-back-button defaultHref=".." /></ion-back-button>
    </ion-buttons>
    <ion-title>Buscador de plantas</ion-title>
  </ion-toolbar>
  <ion-searchbar placeholder="Buscar..." animated (ionchange)="buscar($event)" debounce="100"></ion-searchbar>
</ion-header>

<ion-content>
  <ion-list>
    <ion-item *ngFor="let plant of this.infoPlantasArray | filtroPlantas: busqueda" (click)="mostrarInfo(plant)">
      {{plant.nombre}}
    </ion-item>
  </ion-list>
</ion-content>

```

Figura 57: Fragmento de código - Código de ver información de plantas

```

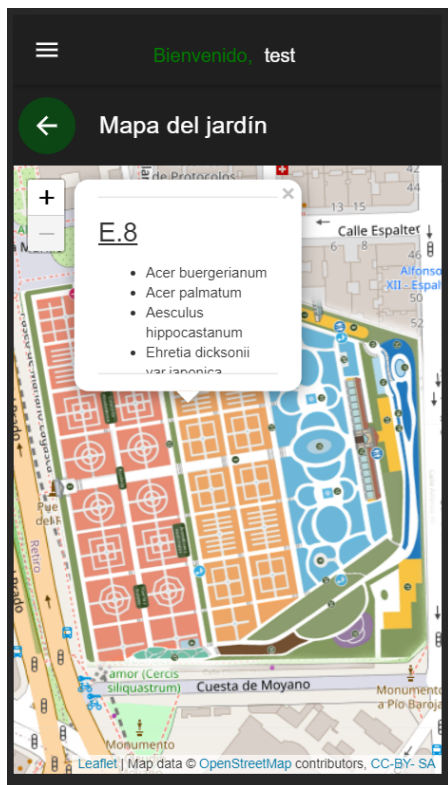
arboles.csv
src > assets > scraping > arboles.csv
1 Nombre|ubicación|covid|foto|descripción
2 Abies alba|E-1.5|Abeto blanco|https://upload.wikimedia.org/wikipedia/commons/2/24/Abies_alba_Pirenees-Orientales.jpg|Abies alba, el abeto común o abeto blanco es una especie arbórea de la familia de las pináceas, originaria de las regiones montañosas de Europa. De porte piramidal, tamaño medio o elevado, entre 20 y 50 metros, puede alcanzar los 60 metros de altura, su tronco es derecho y columnar, desprovisto de ramas en su parte inferior, de hasta 6 metros de circunferencia, con corteza cenicienta a blanquecina, lisa y con vesículas resinosas; se oscurece y resquebraja en los ejemplares viejos.
3 Abies bracteata|E-1.4|Abeto de Santa Lucía|https://upload.wikimedia.org/wikipedia/commons/8/82/Abies_bracteata_02_Hutterer22%20cap.jpg|Abies bracteata, el abeto de Santa Lucía o gigantea, es una especie poco común de abeto que está confinado a las laderas y profundidades de los cañones rocosos de las Montañas de Santa Lucía, en la costa central de California, Estados Unidos.
4 Abies cephalonica|TOL-3|Abeto griego|https://upload.wikimedia.org/wikipedia/commons/5/5b/Abies_ceph.Enos.JPG|Abies cephalonica, abeto griego o abeto de Cefalonia es una especie arbórea de la familia de las pináceas, originaria de las regiones montañosas de Grecia, fundamentalmente en el Peloponeso y en la isla de Cefalonia.
5 Abies concolor|E-1.4|Abeto del Colorado|https://upload.wikimedia.org/wikipedia/commons/f/f7/Abies_concolor_subsp._concolor_crown.jpg|Abies concolor o abeto del Colorado es una especie arbórea de la familia de las pináceas, originaria de las montañas occidentales de Norteamérica, a altitudes de 900 a 3.400 msnm.

```

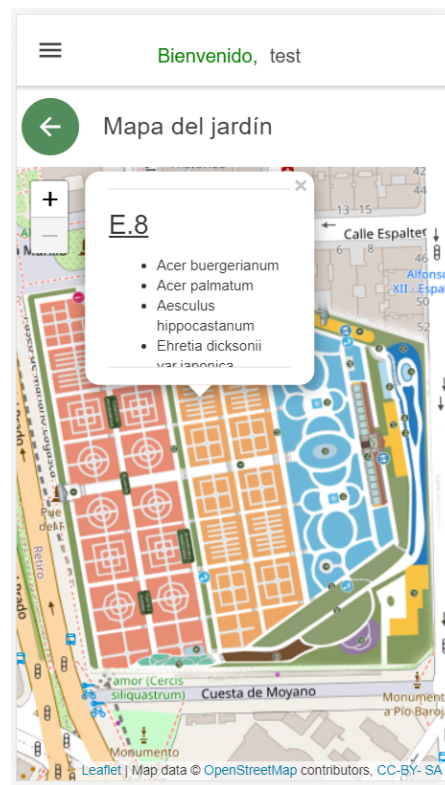
Figura 58: CSV con la información de las plantas

### 6.2.4.2 Ver mapa

Los usuarios disponen de un mapa, en el cual puede consultar el nombre de las parcelas del Real Jardín Botánico y las plantas que contiene dicha parcela. La visualización del mapa se ha obtenido gracias a [15], el mapa gracias y la división de sus parcelas gracias al plano proporcionado por el Real Jardín botánico.



(a) Tema oscuro



(b) Tema claro

Figura 59: Vista ver mapa

En la figura 60 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

this.map = new L.Map("mapId", {maxBoundsViscosity: 1.0}).setView([40.4111469579515, -3.6911385189502726], 17);
L.tileLayer('https://s.tile.openstreetmap.org/{z}/{x}/{y}.png',
{
  attribution: 'Map data © <a href="https://www.openstreetmap.org/">OpenStreetMap</a> contributors, <a href="https://creativecommons.org/licenses/by-sa/2.0/">CC-BY-SA</a>',
  minZoom: 17,
  maxZoom: 19
})
.addTo(this.map);

```

Figura 60: Fragmento de código - Código de ver mapa

### 6.2.5 Vistas de administrador

Los usuario de tipo administrador en su vista dispone de dos funcionalidades, como se muestra en la figura 61.

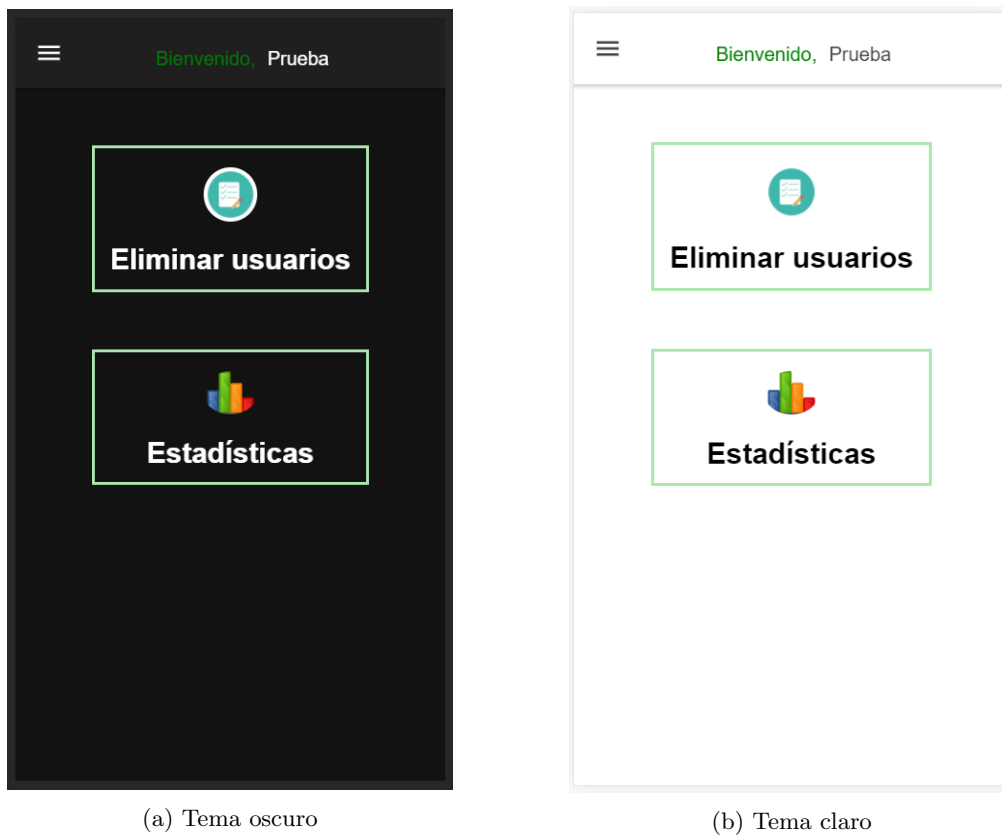
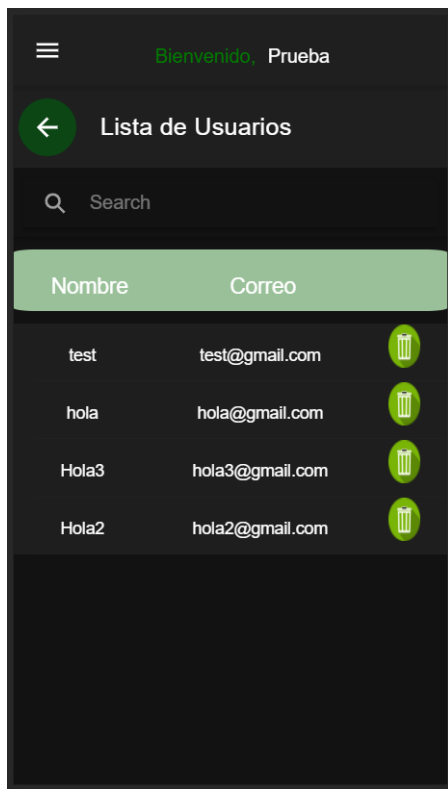


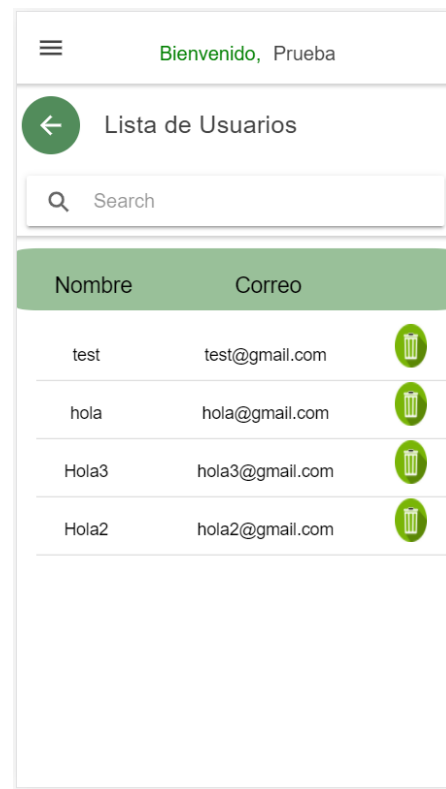
Figura 61: Vista de administrador

#### 6.2.5.1 Eliminar usuarios

Los usuarios de tipo administrador disponen de una lista de usuarios mediante la cual pueden eliminar a usuarios si así lo desean, para ello disponen de un botón con forma de papelera, como se muestra en la figura 62.



(a) Tema oscuro



(b) Tema claro

Figura 62: Vista de eliminar usuarios

En la figura 63 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

async deleteUsers(uid:string){
  this.deleteOpinionsOfUser(uid);
  return await this.http.post<any[]>(`${environment.apiUrl}/deleteUser`, {uid: uid}).toPromise();
}

```

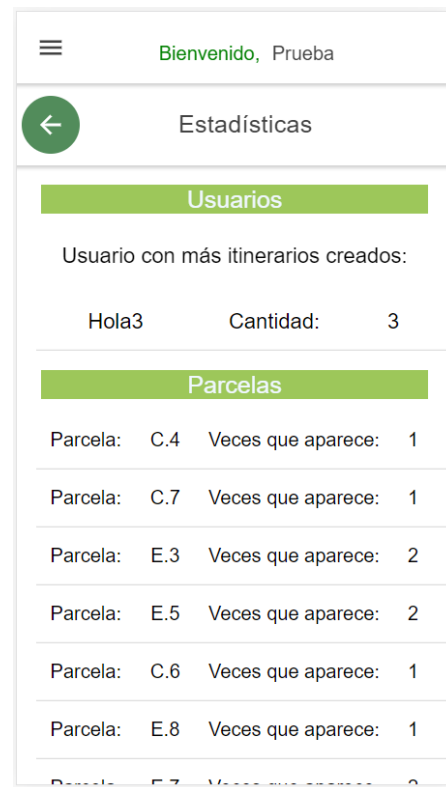
Figura 63: Fragmento de código - Código de eliminar usuarios

### 6.2.5.1 Ver estadísticas

Los usuarios de tipo administrador disponen de una vista en la cual se muestran las estadísticas relacionadas con los usuarios con más itinerarios creados y las parcelas mas visitadas del Real Jardín Botánico, como se muestra en la figura 64.



(a) Tema oscuro



(b) Tema claro

Figura 64: Vista de ver estadísticas

En la figura 65 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

<div>
  <h4>Usuarios</h4>
  <div class="noparcelas">Usuario con más itinerarios creados: </div>
  <ion-item>
    <div class="name-stats">{{this.getMaxCreadorName()}}</div>
    <div class="amount-stats">Cantidad: </div>
    <div class="num-stats">{{this.getMaxCreadorAmmount()}}</div>
  </ion-item>
</div>

<div>
  <h4>Parcelas</h4>
  <div *ngIf="this.hayParcelas(), else sinParcelas">
    <ion-item *ngFor="let p.. of this.getCantidadParcelas()">
      <div class="parcelas-25">
        Parcela:
      </div>
      <div class="parcelas-15">
        {{p.nombre}}
      </div>
      <div class="parcelas-50">
        Veces que aparece:
      </div>
      <div class="parcelas-10">
        {{p.cant}}
      </div>
    </ion-item>
  </div>
  <ng-template #sinParcelas>
    <div class="noparcelas">
      No hay parcelas añadidas aún.
    </div>
  </ng-template>
</div>

```

Figura 65: Fragmento de código - Código de ver estadísticas

## 6.3 API REST

En este apartado se especifica el diseño de la API REST implementada.

Se ha decidido implementar esta API REST con NodeJS debido a la facilidad con la que esta tecnología permite crear esta clase de servicio a la vez que garantiza seguridad en el mismo.

Esta API REST dispone de diversos endpoints en base al servicio solicitado por la aplicación móvil que le proporcionarán la información requerida. Dichos endpoints, accesibles mediante el método HTTP POST, con la información que proporcionan y la información que necesitan para devolverla, vienen descritos a continuación.

### 6.3.1 getRoutesList

Este endpoint se encuentra accesible en la ruta `/api/getRoutesList/`. No necesita ningún parámetro en la petición y devuelve la información sobre los itinerarios almacenados en la base de datos, esta es, los itinerarios creados, sus opiniones, etc. En la figura 66 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

let ruta = await admin.firestore()
  .collection('users')
  .doc(user.id)
  .collection('rutasCreadas')
  .get()
  .then(snapR => {
    if (snapR != null) {
      snapR.forEach(r => {
        rutasUser.push({ id: r.id, data: r.data(), creador:user.id });
      })
    }
  })
colRutas.push({ user: user, rutas: rutasUser });

```

Figura 66: Fragmento de código - Código del endpoint getRoutesList

### 6.3.2 getFavorites

Este endpoint se encuentra accesible en la ruta `/api/getFavorites/`. Su función consiste en obtener la lista de itinerarios favoritos de un usuario concreto. Para ello necesita que se le envíe el id del usuario a consultar en la petición POST en el parámetro "userId". En la figura 67 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

var favsUser = await admin.firestore()
  .collection('users')
  .doc(String(userDoc))
  .collection('rutasFavoritas')
  .get()
  .then(snap => {
    if (snap != null) {
      snap.forEach(f => {
        favs.push({ id: f.id, data: f.data() });
      })
    }
    response.status(200).json(favs);
  })

```

Figura 67: Fragmento de código - Código del endpoint getFavorites

### 6.3.3 deleteFav

Mediante este endpoint, accesible en la ruta `/api/deleteFav/`, la API elimina un itinerario de los favoritos del usuario. Para ello necesita los parámetros "uid", con el id del usuario del que se quiere eliminar el itinerario, y "rid", con el id del itinerario a eliminar. En la figura 68 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```
var deleteFav = await admin.firestore()
  .collection('users')
  .doc(String(userDoc))
  .collection('rutasFavoritas')
  .doc(String(rutaDoc))
  .delete()
  .then(res => {
    response.status(200).json(res);
  })
```

Figura 68: Fragmento de código - Código del endpoint deleteFav

#### 6.3.4 addFav

Este endpoint, ubicado en la ruta /api/addFav/, añade un itinerario a la lista de itinerarios favoritos del usuario. Para realizar esta acción se necesitan los siguientes parámetros:

- uid: id del usuario al que se le quiere añadir el itinerario a favoritos.
- rid: id del itinerario a añadir.
- nombreRuta: nombre del itinerario a añadir.
- creador: id del usuario creador del itinerario.

En la figura 69 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

var ruta = await admin.firestore()
  .collection('users')
  .doc(String(userDoc))
  .collection('rutasFavoritas')
  .add({
    creadorId: String(request.body['creador']),
    rutaId: String(request.body['rid']),
    nombreRuta: String(request.body['nombreRuta'])
  }).then(res => {
    res.get().then(ref => {
      const docs = [];
      docs.push(ref.id);
      docs.push(ref.data());
      response.status(200).json(docs);
    })
  });

```

Figura 69: Fragmento de código - Código del endpoint addFav

### 6.3.5 getHistorialList

Usando este endpoint, ubicado en la ruta `/api/getHistorialList/`, se obtiene el historial de itinerarios realizados por un usuario concreto. Para ello se necesita el parámetro "email" que contiene el correo electrónico del usuario a consultar. En la figura 70 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

const fav = await admin.firestore().collection('users').doc(String(idUserg)).collection('historial').get().then(snapR => {
  snapR.forEach(r => {
    reth.push(r.data());
  });
});

response.status(200).json(reth);

```

Figura 70: Fragmento de código - Código del endpoint getHistorialList

### 6.3.6 addHistorial

Este endpoint permite añadir un itinerario al historial de un usuario. Se encuentra ubicado en la ruta `/api/addHistorial/` y necesita los siguientes parámetros:

- email: correo electrónico del usuario al que se le quiere añadir el itinerario a su historial.
- nombreRuta: nombre del itinerario a añadir.
- fecha: fecha en la que se ha iniciado el itinerario.

En la figura 71 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

const historial = await admin.firestore()
  .collection('users')
  .doc(String(docUser))
  .collection('historial')
  .add({
    nombre:String(request.body['nombreRuta']),
    fecha:String(request.body['fecha'])
  });

```

Figura 71: Fragmento de código - Código del endpoint addHistorial

### 6.3.7 addRuta

Este endpoint permite añadir un nuevo itinerario a la base de datos. Para ello se debe enviar la petición a la ruta /api/addRuta/ con los siguientes parámetros:

- uid: id del usuario que crea el itinerario.
- nombre: nombre del itinerario.
- coords: array con las coordenadas de las parcelas que contiene el itinerario.
- nombreParcelas: array con el nombre de las parcelas que contiene el itinerario.
- distancia: longitud en kilómetros del itinerario.
- tiempo: tiempo necesario para completar el itinerario.

En la figura 72 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

var ruta = await admin.firestore()
  .collection('users')
  .doc(String(userDoc))
  .collection('rutasCreadas')
  .add({
    nombre: String(request.body['nombre']),
    coords: request.body['coords'],
    distancia: request.body['distancia'],
    tiempo: request.body['tiempo'],
    nombreParcelas: request.body['nombreParcelas']
  }).then(res => {
    res.get().then(ref => {
      response.status(200).json({id:ref.id, data:ref.data()});
    })
  });

```

Figura 72: Fragmento de código - Código del endpoint addRuta

### 6.3.8 userList

Utilizando este endpoint, ubicado en la ruta /api/userList, se obtiene una lista con los usuarios registrados en la aplicación. Esta petición no necesita ningún parámetro para realizarse. En la figura 73 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

exports.userList = functions.https.onRequest((request, response) => {
  response.set('Access-Control-Allow-Origin', '*');
  response.set('Access-Control-Allow-Methods', 'POST');
  response.set('Access-Control-Allow-Headers', 'Content-Type');

  var ret;
  var prueba = admin.firestore()
    .collection('users')
    .get()
    .then(snapshot => {
      const docs = [];
      snapshot.forEach(doc => {
        docs.push(doc.data());
      });
      response.json(docs);
    })
  })
})

```

Figura 73: Fragmento de código - Código del endpoint userList

### 6.3.9 deleteUser

Este endpoint permite eliminar un usuario del sistema junto a sus itinerarios creados, opiniones, itinerarios favoritos, historial, etc. Ubicado en la ruta /api/deleteUser/ necesita el parámetro "uid", que contiene el id del usuario a eliminar. En la figura 74 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

let deleteU = admin.firestore().collection('users').doc(String(idUser)).delete().then(() => {
  response.status(200).json(docs);
});

```

Figura 74: Fragmento de código - Código del endpoint deleteUser

### 6.3.10 changeUsername

Usando este endpoint se actualiza el nombre de usuario de un usuario junto al nombre en las opiniones que este haya publicado. Se ubica en la ruta /api/changeUsername/ y necesita los parámetros "oldName" y "newName", que contienen el nombre de usuario antiguo y el nuevo respectivamente del usuario a actualizar. En la figura 75 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```

var aux = await admin.firestore()
  .collection('users')
  .doc(String(idUser))
  .set({
    userName: String(docs[1]),
    userNameLowered: String(docs[1]).toLowerCase()
  }, { merge: true })

```

Figura 75: Fragmento de código - Código del endpoint changeUsername

### 6.3.11 addOpinion

Con este endpoint se permite añadir una opinión a un itinerario. Se encuentra en la ruta `/api/addOpinion/` y requiere de los siguientes parámetros:

- creadorid: id del creador del itinerario.
- rid: id del itinerario.
- uid: id del usuario que crea la opinión.
- userName: nombre de usuario del usuario que crea la opinión.
- texto: texto con la opinión del usuario.

En la figura 76 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```
let op = await admin.firestore()
  .collection('users')
  .doc(String(request.body['creadorid']))
  .collection('rutasCreadas')
  .doc(String(request.body['rid']))
  .collection('opiniones')
  .add(
    {
      uid: String(request.body['uid']),
      name: String(request.body['userName']),
      texto: String(request.body['texto'])
    }
  ).then(res =>{
    res.get().then(ref => {
      const docs = [];
      docs.push(ref.id);
      docs.push(ref.data());
      response.status(200).json(docs);
    })
  })
})
```

Figura 76: Fragmento de código - Código del endpoint addOpinion

### 6.3.12 deleteOpinion

Con este endpoint se permite eliminar una opinión de un itinerario. Se encuentra en la ruta `/api/deleteOpinion/` y requiere de los siguientes parámetros:

- creadorid: id del creador del itinerario.
- rid: id del itinerario.
- uid: id del usuario que ha creado la opinión.

En la figura 77 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```
let deleteOpinion = await admin.firestore()
  .collection('users')
  .doc(String(request.body['creadorid']))
  .collection('rutasCreadas')
  .doc(String(request.body['rid']))
  .collection('opiniones')
  .doc(String(docOpinion))
  .delete();
```

Figura 77: Fragmento de código - Código del endpoint deleteOpinion

### 6.3.13 updateOpinion

Con este endpoint se permite modificar una opinión de un itinerario. Se encuentra en la ruta /api/updateOpinion/ y requiere de los siguientes parámetros:

- creadorid: id del creador del itinerario.
- rid: id del itinerario.
- uid: id del usuario que ha creado la opinión.
- texto: texto con la opinión modificada del usuario.

En la figura 78 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```
let updateOpinion = await admin.firestore()
  .collection('users')
  .doc(String(request.body['creadorid']))
  .collection('rutasCreadas')
  .doc(String(request.body['rid']))
  .collection('opiniones')
  .doc(String(docOpinion))
  .set({
    texto:String(request.body['texto'])
  }, { merge: true })
```

Figura 78: Fragmento de código - Código del endpoint updateOpinion

### 6.3.14 deleteRuta

Este endpoint permite eliminar un itinerario del sistema. Además, elimina el itinerario de la lista de favoritos y/o del historial de los otros usuarios que lo tuvieran en ellas. Es accesible mediante la ruta /api/deleteRuta/ y necesita los siguientes parámetros:

- uid: id del usuario que ha creado el itinerario.
- rid: id del itinerario.
- nombreRuta: nombre del itinerario.

En la figura 79 se puede observar el fragmento de código relacionado con la funcionalidad descrita.

```
const deleteRuta = await admin.firestore()  
  .collection('users')  
  .doc(String(request.body['uid']))  
  .collection('rutasCreadas')  
  .doc(String(request.body['rid']))  
  .delete();
```

Figura 79: Fragmento de código - Código del endpoint deleteRuta

## Capítulo 7: Evaluación

En este capítulo se observan los resultados obtenidos de la evaluación realizada sobre la aplicación desarrollada. Se ha creado un formulario mediante Google Forms [32] y se ha compartido con usuarios de cualquier ámbito y/o edad ya que cualquier tipo de persona puede visitar el Real Jardín Botánico de Madrid.

### 7.1 Resultados de la encuesta

A continuación se listan los resultados obtenidos de la encuesta. Salvo las preguntas del subapartado 7.1.1, que vienen explicadas en su sección correspondiente, el rango de valores proporcionado para dar una respuesta se encuentra entre 1 y 10, siendo 1 el valor mínimo de satisfacción y 10 el valor máximo de satisfacción.

La muestra consiste de un conjunto de 20 personas con 12 hombres y 8 mujeres entre 22 y 47 años de edad.

#### 7.1.1 Datos generales del encuestado

En esta pregunta se le pide al encuestado que indique su edad mediante un campo de texto breve. En la figura 80 se observa que la edad de los usuarios encuestados es bastante variada, abarcando desde los 22 años a los 47. El 70% de los encuestados presenta una edad entre los 22 y los 33 años, mientras que el 30% restante tiene una edad entre los 34 y los 47 años.

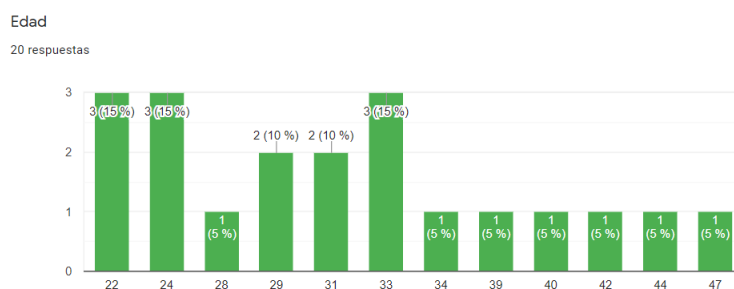


Figura 80: Respuestas de la primera pregunta del cuestionario: edad

En cuanto al sexo de los encuestados en la figura 81 se observa que el 60% son hombres mientras que el 40% son mujeres. Las opciones disponibles para responder consisten en un campo de selección con las siguientes opciones: hombre, mujer, prefiero no decirlo y otro, este último pudiendo indicar por texto una respuesta.

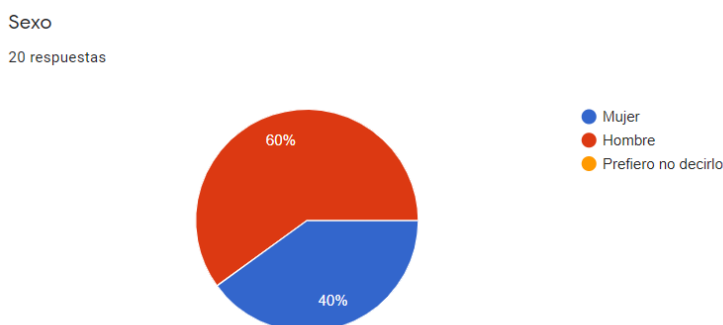


Figura 81: Respuestas de la segunda pregunta del cuestionario: sexo

En esta pregunta se le solicita al encuestado que indique si ha visitado en alguna ocasión o no el jardín, indicando el grado en que lo ha hecho o no, pudiendo indicar que lo visita habitualmente, que lo ha hecho en alguna ocasión, que no lo ha hecho pero tendría interés o que no lo ha hecho y que no le interesa. Respecto a sus respuestas la figura 82 nos muestra como el 60% no lo ha visitado nunca pero tendría interés mientras que el 40% lo ha visitado alguna vez.

¿Has visitado alguna vez el Real Jardín Botánico de Madrid?

20 respuestas

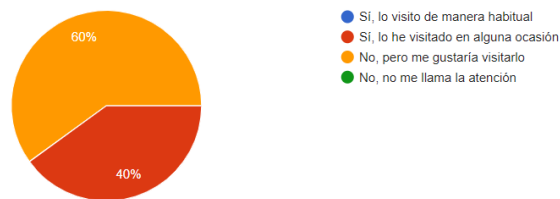


Figura 82: Respuestas de la tercera pregunta del cuestionario: visitas al jardín

### 7.1.2 Vista del mapa

En esta pregunta se le solicita al encuestado que indique su nivel de satisfacción con la vista del mapa del jardín. Esta fue valorada por los encuestados con una nota entre 9 y 10 por el 75% de los mismos. El 25% restante la valoró con un 8 (ver figura 83).

Indica tu nivel de satisfacción con la vista del mapa

20 respuestas

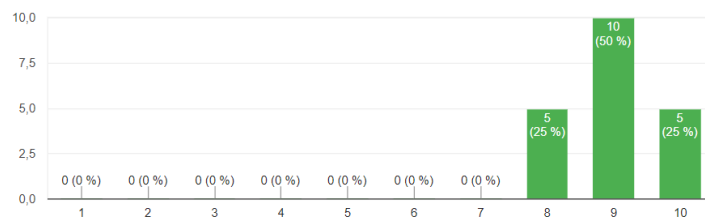


Figura 83: Respuestas de la cuarta pregunta del cuestionario: vista del mapa

Respecto a esta pregunta, donde se solicita que se valore la satisfacción con la información ofrecida de las parcelas, el 55% de los encuestados le otorgó una nota entre 6 y 7, mientras que el 45% la valoró con una nota entre 8 y 9 (ver figura 84).

Indica tu nivel de satisfacción con la información de las parcelas

20 respuestas

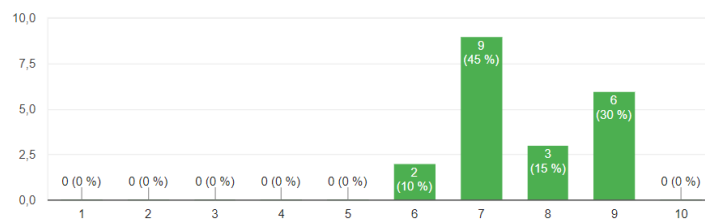


Figura 84: Respuestas de la quinta pregunta del cuestionario: información de las parcelas

### 7.1.3 Consulta de itinerarios

En esta pregunta se pide una valoración de la vista de la lista de itinerarios. Fue valorada con una nota entre 8 y 9 por el 85% de los encuestados, mientras que el 15% le dio un 7 (ver figura 85).

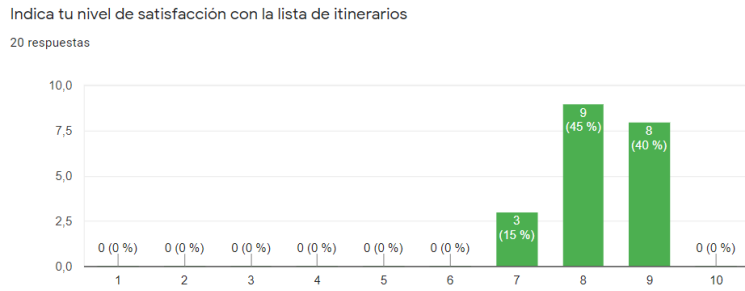


Figura 85: Respuestas de la sexta pregunta del cuestionario: lista de itinerarios

Al preguntar sobre la satisfacción con el buscador de itinerarios este alcanzó una nota de 9 o 10 por el 65% de los encuestados. El 35% lo valoró con una nota entre 7 y 8 (ver figura 86).

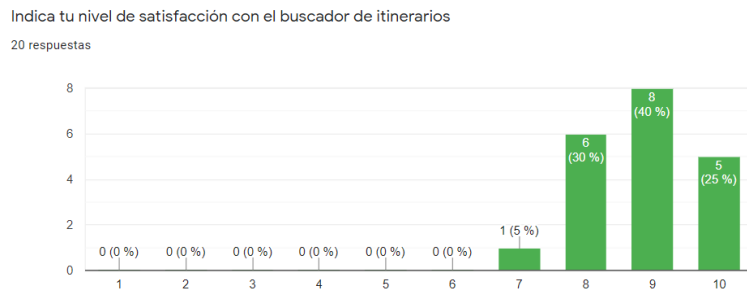


Figura 86: Respuestas de la séptima pregunta del cuestionario: buscador de itinerarios

Solicitando la valoración sobre la vista detallada de los itinerarios esta obtuvo una valoración entre 8 y 10 por el 75% de los encuestados, mientras que el 25% la valoró con una nota entre 6 y 7 (ver figura 87).

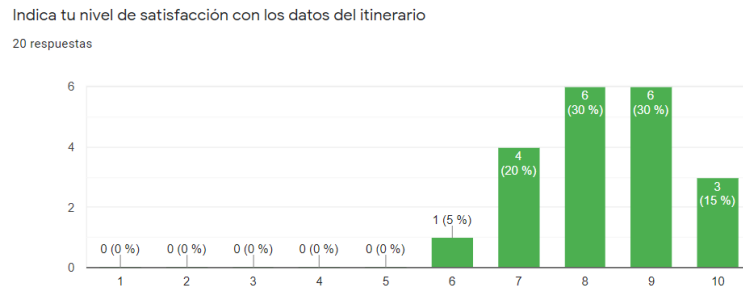


Figura 87: Respuestas de la octava pregunta del cuestionario: datos de los itinerarios

#### 7.1.4 Consulta de información sobre plantas

Preguntando sobre la valoración de la lista de plantas esta ha recibido una valoración entre 9 y 10 por el 30% de los encuestados, mientras que el 70% la valoró con una nota entre 6 y 8 (ver figura 88).

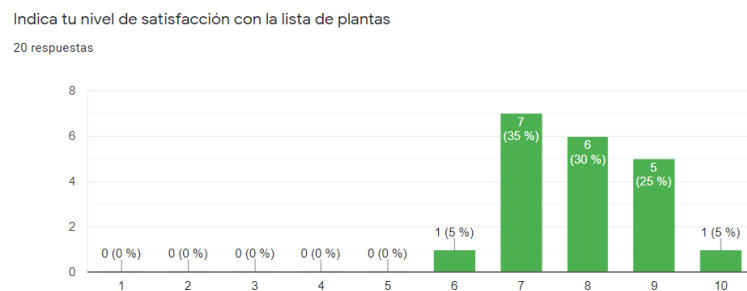


Figura 88: Respuestas de la novena pregunta del cuestionario: lista de plantas

Al solicitar la valoración del buscador de plantas este ha sido valorado con una nota entre 9 y 10 por el 75% de los encuestados. El 25% restante lo valoró con una nota entre 7 y 8 (ver figura 89).

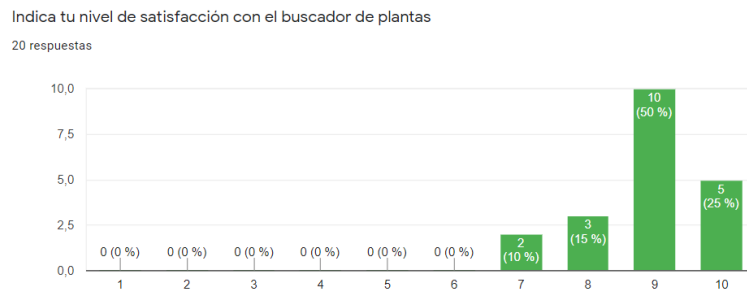


Figura 89: Respuestas de la décima pregunta del cuestionario: buscador de plantas

En esta pregunta se pide la valoración de la información ofrecida sobre las plantas, que ha recibido una nota entre 7 y 9 por el 75% de los encuestados. El otro 25% la valoró con una nota entre 5 y 6 (ver figura 90).

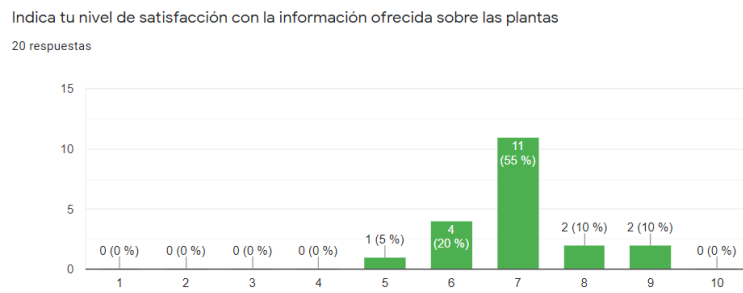


Figura 90: Respuestas de la undécima pregunta del cuestionario: información sobre plantas

### 7.1.5 Menú lateral, inicio de sesión y registro

En esta pregunta se pide una valoración del menú lateral de la aplicación, que ha recibido una valoración entre 9 y 10 por el 60% de los encuestados. El 40% restante le ha dado una valoración entre 7 y 8 (ver figura 91).

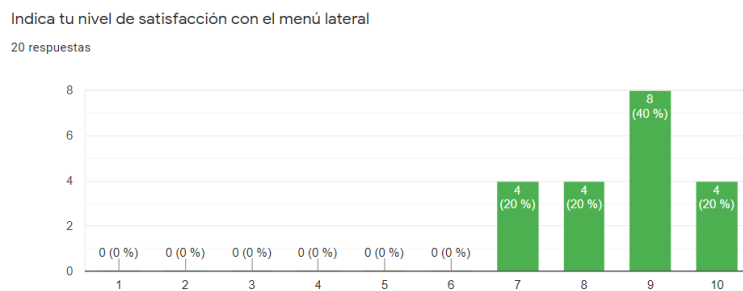


Figura 91: Respuestas de la duodécima pregunta del cuestionario: menú lateral

La valoración solicitada de la pantalla de inicio de sesión ha obtenido una nota entre 9 y 10 por el 80% de los encuestados, mientras que el otro 20% le ha otorgado una nota entre 7 y 8 (ver figura 92).

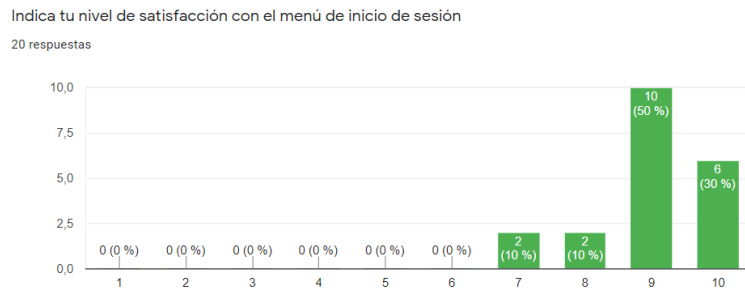


Figura 92: Respuestas de la decimotercera pregunta del cuestionario: inicio de sesión

Al pedir la valoración del menú de registro de usuarios este ha recibido por el 80% de los encuestados una valoración entre 9 y 10, mientras que el 20% restante lo ha valorado con una nota entre 7 y 8 (ver figura 93).

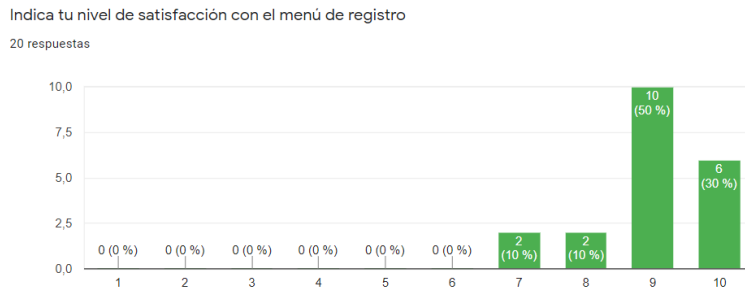


Figura 93: Respuestas de la decimocuarta pregunta del cuestionario: registro sobre plantas

### 7.1.6 Crear itinerario

Esta pregunta solicita la valoración de la vista de creación de itinerarios, que ha obtenido una valoración entre 9 y 10 por el 65% de los encuestados. Una nota entre 7 y 8 ha sido otorgada por el 35% restante (ver figura 94).

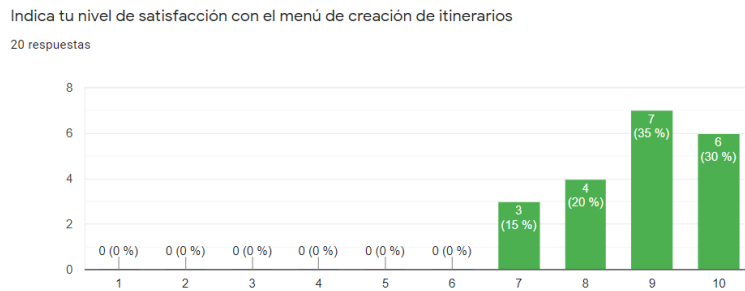


Figura 94: Respuestas de la decimoquinta pregunta del cuestionario: creación de itinerarios

### 7.1.7 Nuevas opciones en los itinerarios

Al solicitar la valoración sobre el botón de añadir y/o eliminar de favoritos este ha sido valorado con una nota entre 8 y 10 por el 70% de los encuestados, mientras que el 30% restante lo ha valorado con una nota entre 5 y 7 (ver figura 95).

Indica tu nivel de satisfacción con el botón para añadir/eliminar de favoritos

20 respuestas

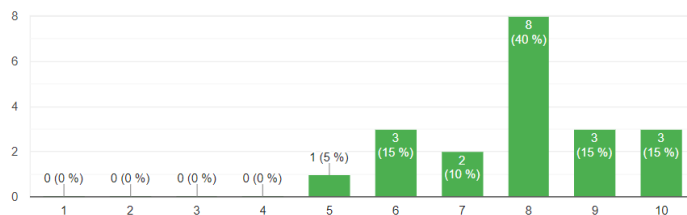


Figura 95: Respuestas de la decimosexta pregunta del cuestionario: botón de favoritos

Respecto a la valoración solicitada sobre el botón para modificar las opiniones en los itinerarios el 70% le ha otorgado una nota entre 8 y 10, recibiendo una nota entre 6 y 7 por el otro 30% de los encuestados (ver figura 96).

Indica tu nivel de satisfacción con el botón para añadir/editar/eliminar opiniones de los itinerarios

20 respuestas

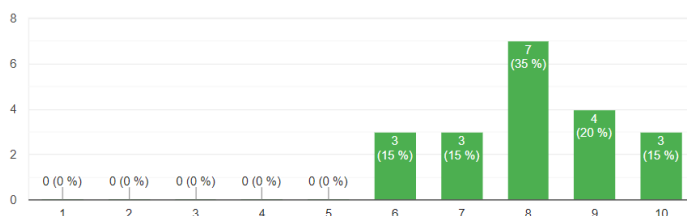


Figura 96: Respuestas de la decimoséptima pregunta del cuestionario: botón de comentarios

### 7.1.8 Opciones de perfil de usuario

La valoración pedida de la vista "Tu perfil" ha obtenido por el 55% de los usuarios una puntuación entre 6 y 7. El otro 45% la ha valorado con una nota entre 8 y 10 (ver figura 97).

Indica tu nivel de satisfacción con la vista "Tu perfil"

20 respuestas

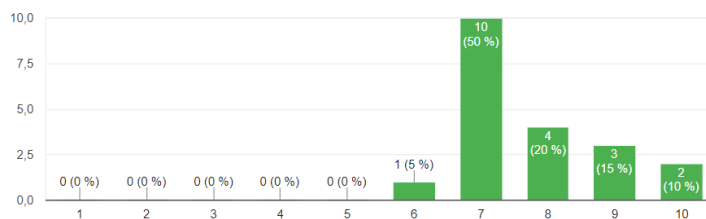


Figura 97: Respuestas de la decimooctava pregunta del cuestionario: tu perfil

Al preguntar sobre la valoración para el menú para cambiar el nombre de usuario este ha obtenido una nota entre 6 y 7 por la mitad de los encuestados, siendo valorado entre 8 y 9 por la otra mitad (ver figura 98).

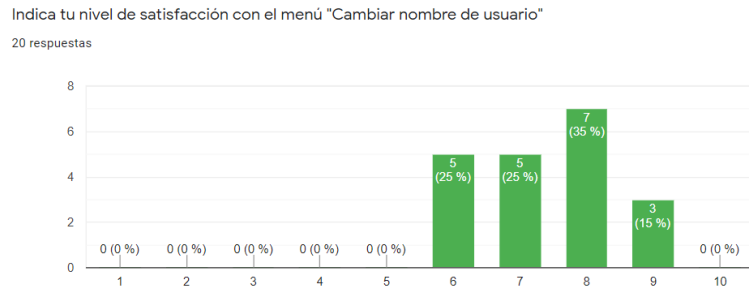


Figura 98: Respuestas de la decimonovena pregunta del cuestionario: cambiador nombre de usuario

Al solicitar la valoración de la vista para cambiar la contraseña esta ha recibido una nota entre 9 y 10 por el 55% de los encuestados. El otro 45% la ha valorado con una nota entre 7 y 8 (ver figura 99).

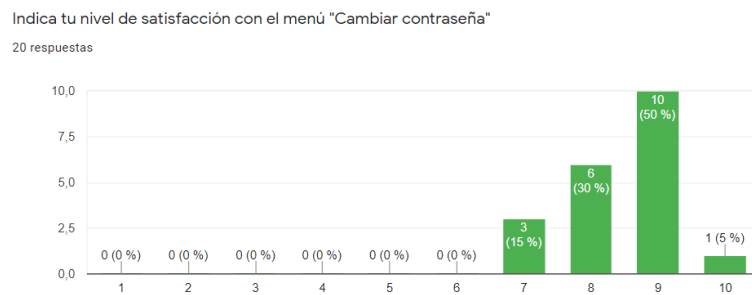


Figura 99: Respuestas de la vigésima pregunta del cuestionario: cambiar contraseña

En esta pregunta se pide la valoración sobre la visualización del historial de itinerarios, que ha obtenido una valoración entre 6 y 8 por el 65% de los encuestados, recibiendo entre 9 y 10 de valoración por el otro 35% (ver figura 100).

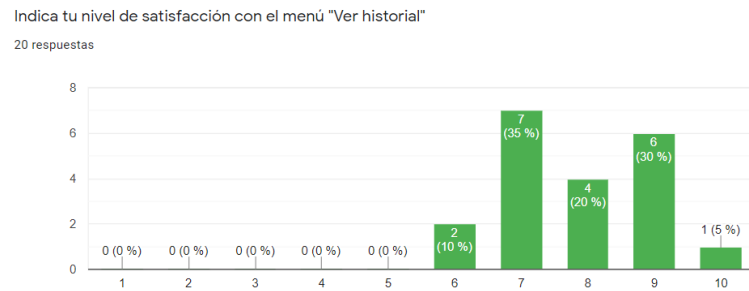


Figura 100: Respuestas de la vigésimo primera pregunta del cuestionario: ver historial

Solicitando la valoración sobre la lista de itinerarios favoritos del usuario esta ha recibido por el 65% de los encuestados una nota entre 9 y 10, siendo valorada por el otro 35% con notas entre 6 y 8 (ver figura 101).

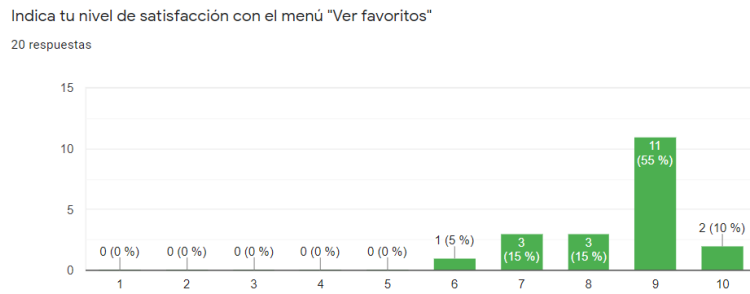


Figura 101: Respuestas de la vigésimo segunda pregunta del cuestionario: ver favoritos

La valoración requerida sobre la vista del menú para ver los itinerarios creados por el usuario ha obtenido por el 70% de los encuestados una nota entre 9 y 10, mientras que el otro 30% le ha dado una valoración entre 6 y 8 (ver figura 102).

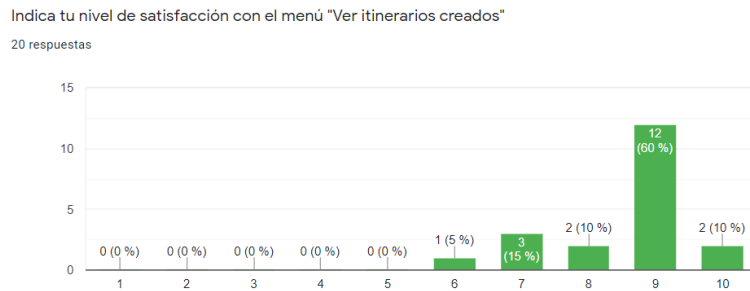


Figura 102: Respuestas de la vigésimo tercera pregunta del cuestionario: ver itinerarios creados

### 7.1.9 Administrador

Al preguntar sobre la valoración sobre la vista del menú principal de administrador el 70% de los encuestados lo ha valorado con una nota entre 8 y 10, mientras que el otro 30% lo ha valorado con una nota entre 6 y 7 (ver figura 103).

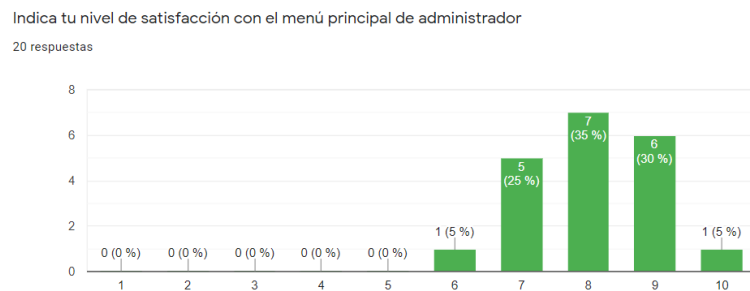


Figura 103: Respuestas de la vigésimo cuarta pregunta del cuestionario: menú principal de administrador

En esta pregunta se solicita la valoración sobre la lista de usuarios, que ha sido valorada por el 80% de los encuestados con una nota entre 8 y 10. Una valoración entre 6 y 7 le ha sido otorgada por el 20% restante (ver figura 104).

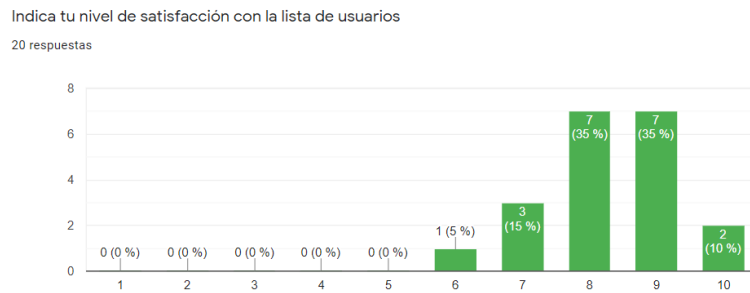


Figura 104: Respuestas de la vigésimo quinta pregunta del cuestionario: lista de usuarios

Al pedir la valoración sobre la vista de estadísticas esta ha recibido una valoración entre 6 y 7 por el 70% de los encuestados, recibiendo por el otro 30% una valoración entre 8 y 9 (ver figura 105).

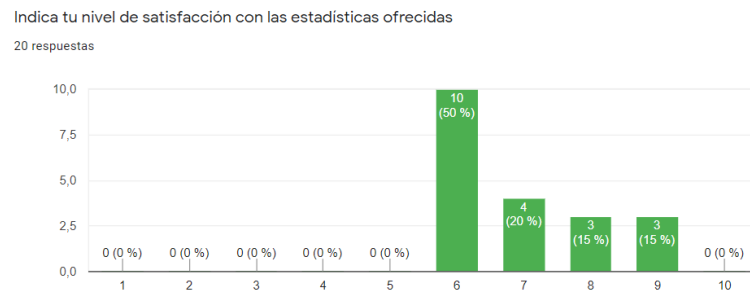


Figura 105: Respuestas de la vigésimo sexta pregunta del cuestionario: estadísticas

### 7.1.10 Satisfacción general con la aplicación

El 55% de los encuestados ha respondido a esta pregunta, en la que se pide la valoración sobre el diseño general de la aplicación, con una nota entre 9 y 10, mientras que el otro 45% le ha dado una valoración de 8 (ver figura 106).

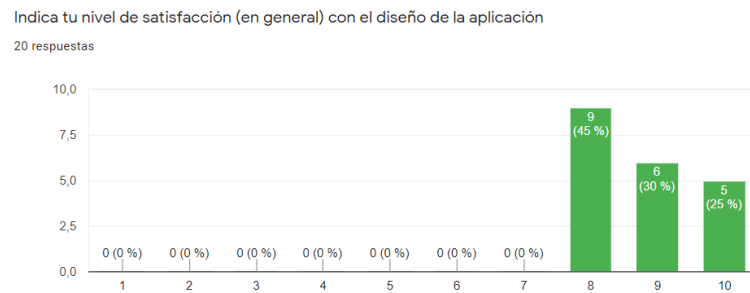


Figura 106: Respuestas de la vigésimo séptima pregunta del cuestionario: diseño general de la aplicación

Al preguntar la valoración otorgada a la fluidez de la aplicación esta ha sido valorada con una nota entre 8 y 10 por el 55% de los encuestados. El otro 45% la ha valorado con una nota entre 6 y 7 (ver figura 107).

Indica tu nivel de satisfacción (en general) con la fluidez de la aplicación

20 respuestas

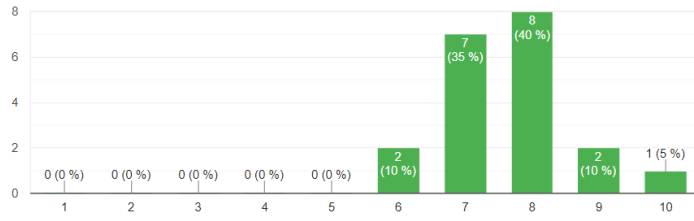


Figura 107: Respuestas de la vigésimo octava pregunta del cuestionario: fluidez de la aplicación

## Capítulo 8: Conclusiones y trabajo futuro

### 8.1 Conclusiones

En este trabajo se ha desarrollado una herramienta que facilita a los visitantes del Real Jardín Botánico de Madrid realizar visitas al mismo mediante la generación de diversos itinerarios. A su vez se ofrecen los servicios adicionales de consulta de la ubicación dentro del propio jardín y de consulta de información sobre las plantas y árboles que tiene el mismo. Se ha procurado que la mayor parte de las tareas sea automática y hacer lo más sencillas e intuitivas aquellas que requieren interacción del usuario.

Gracias al uso de Leaflet [15] se ha podido gestionar la visualización del mapa y el uso de los servicios de geolocalización, que son el objetivo principal para poder ofrecer el servicio de localización dentro del Real Jardín Botánico. Del mismo modo se ha conseguido el que los usuarios puedan compartir los itinerarios creados por ellos mismos.

En la realización de este trabajo se han utilizado principalmente tecnologías de código abierto para el desarrollo de aplicaciones híbridas. Del mismo modo, también se han puesto en práctica conceptos de ingeniería de software, de desarrollo de sistemas interactivos y de gestión de bases de datos entre otros. Todas estas cosas han permitido un desarrollo del proyecto bastante cómodo una vez que se ha superado la dificultad inicial de adaptarse a dichas tecnologías.

### 8.2 Trabajo futuro

Pese a que la mayor parte de los objetivos planteados al inicio del proyecto han sido alcanzados a lo largo del mismo han surgido aspectos que se proponen como trabajo futuro:

- **Caminos alternativos en los itinerarios.** El Real Jardín Botánico de Madrid presenta algunos caminos a través del mismo que podrían no ser accesibles para todos los visitantes del mismo. Por ello sería conveniente presentar una opción para que los visitantes puedan elegir el camino estándar o un camino alternativo que cubra las necesidades que precisen.
- **Elegir idioma de la aplicación.** El Real Jardín Botánico de Madrid es un punto de gran interés turístico de la ciudad de Madrid, por lo que es visitado por una gran cantidad de turistas que no tienen por qué saber castellano. Es por esto que la posibilidad de elegir el idioma en el que se muestra la aplicación ayudaría a estas personas a utilizarla.
- **Mostrar ubicación de los lavabos.** Un aspecto importante a la hora de realizar una visita al jardín es poder disponer de un lavabo en caso de que se necesite. Por tanto indicar en el mapa la ubicación de los mismos junto a instrucciones para alcanzarlos resultaría de gran ayuda para aquellos visitantes que no sepan dónde se encuentran los mismos.
- **Conectar el menú de información de las plantas con el mapa.** Al pulsar sobre una parcela del mapa se muestran las plantas que esta contiene. Sin embargo, solo se pueden visualizar sus nombres científicos sin ninguna información adicional. En este sentido convendría permitir que al pulsar sobre una planta concreta se mostrara la misma información que se muestra en el menú correspondiente de información de plantas. Del mismo modo, en este menú también se podría mostrar el mapa al pulsar sobre el nombre de la parcela en la que aparece la planta o el árbol correspondiente.
- **Mostrar ubicación más precisa de las plantas y árboles.** Actualmente, la ubicación de las plantas y árboles del jardín viene indicada por el nombre de las parcelas en las que aparece. No obstante, debido al tamaño de las mismas y a que hay más elementos en ellas puede ser complicado localizar el elemento concreto que se busca. Es por ello que indicar de manera concreta en qué parte de la parcela se ubica la planta o árbol correspondiente facilitaría la localización del mismo.
- **Más información sobre los elementos del jardín.** La información que se muestra actualmente sobre las plantas y árboles del jardín se ha obtenido mediante una técnica de *web scraping* [31] debido a no disponer de otra fuente de información. Debido a esto dicha información presenta lagunas y no es consistente en todos sus elementos. Por tanto, una mejora para este servicio consiste en obtener una mayor cantidad de información sobre los elementos anteriormente mencionados.

- **Mejoras y correcciones.** Durante el desarrollo software una de las partes más importantes consiste en su mantenimiento. Durante este desarrollo y gracias a la retroalimentación obtenida por usuarios de la aplicación se espera corregir errores de la misma y plantear mejoras. Del mismo modo, también se pueden realizar optimizaciones de código y/o de seguridad.

## Chapter 8: Conclusions and future work

### 8.1 Conclusions

In this work it has been developed a tool that eases the visitors of the Real Jardín Botánico de Madrid to do visits to itself with the generation of various itineraries. At the same time, it is offered the additional services of consulting the plants' ubication within the garden itself and the information query about the plants and trees the garden is composed of. It has been sought that most part of the taks are done automatically and the ones that require user's interaction do them in the easiest and most intuitive way.

Thanks to the use of Leaflet [15] it has been possible to manage the map's display and the use of the geolocalization services, which is the main objective of the project to offer the service of localization withing the Real Jardín Botánico. In the same way, it has also been achieved the sharing of the itineraries created by the users.

In the making of this project it has mainly been used open sourced code techologies for the development of hybrid applications. In the same way, it also has been put into practice software engineering concepts, interactive system development concepts and concepts about database managment among others. All these things have allowed to develop the project in an easy way once the difficulty of learning these technologies have been overcome.

### 8.2 Future work

Although most part of the objectives proposed at the begining of the project have been reached, along the development of itself there have been raised new aspects that are proposed as future work:

- **Alternative paths in the itineraries.** The Real Jardín Botánico de Madrid has some paths along itself that could be not accesible to all the visitors. Because of that it would be convenient to show an option so the visitors can choose between the standard path or an alternative way that covers all the needs they require.
- **Choose the application's language.** The Real Jardín Botánico de Madrid is a point of great tourist interest, which is why it is visited by a large amount of tourists that don't have to know spanish. It is because of this that the possibility of choosing the application's language would be really helpful for these visitors to use it.
- **Show lavatories' ubication.** An important aspect when visiting the garden is to have a lavatory available in case of need. Therefore, showing in the map the ubication of them along instructions to reach them would be really helpful for all the visitors that don't know where to find them.
- **Connect the plant's information menu with the map.** When tapping on a parcel of the map it shows the plants it contains. However, it can only be seen their scientific names without any additional information. In this sense it would be convenient to allow that when tapping over a plant's name it showed the same information that the plant's information menu shows. In the same way, in this menu it could also show the map when tapping over the parcel's name where this plant or tree is placed.
- **Show a more precise ubication of the plants and trees.** At the moment, the ubication of the plants and trees of the garden is showed by the name of the parcels they appear. However, because of the size of the parcels along with the other elements in it it can be difficult to find the precise element that is wanted. It is because of this that showing the exact part within the parcel where the element is placed would ease the localization of it.
- **More information about the elements in the garden.** The information that is currently shown about the plants and trees of the garden has been obtained with a *web scraping* [31] technique because of not having another information source available. Because of this said information has gaps and is not consistent in all of its elements. Therefore, an improvement in this service consists on obtaining more information about the previously said elements.

- **Improvements and corrections.** During the software development one of the most important parts consists on its maintenance. During this development and thanks to the feedback obtained from the application's users it is expected to correct errors in it and to plan improvements. In the same way, code/security upgrades can also be done.

## Capítulo 9: Aportaciones individuales

En este capítulo se explican las tareas y aportaciones individuales de cada miembro del grupo en el/al proyecto.

### 9.1 Francisco García Navarrete

La aportación de Francisco al proyecto se define en los siguientes apartados:

#### Ingeniería de requisitos y selección de tecnologías

Una parte esencial de este trabajo ha sido la ingeniería de requisitos. Al principio resultó algo costoso determinar correctamente el alcance que debía tener la aplicación para este Trabajo de Fin de Grado así como las funcionalidades que debía tener. Por ello, fueron necesarias reuniones con el tutor junto a Fabricio para así acotar las funcionalidades necesarias.

Del mismo modo, determinar con qué tecnologías debía implementarse este trabajo ha sido también determinante para llegar a la completitud del mismo. Se ha necesitado de investigación y comparativas para alcanzar un consenso que permitiera un desarrollo eficaz y a la vez cómodo que llevase a cumplir con los objetivos marcados.

#### Diseño de la base de datos

En la realización del diseño de la base de datos se estudiaron junto a Fabricio varios diagramas de distintos tipos en busca del más adecuado para la aplicación considerando a su vez la tecnología escogida. Este diseño a su vez ha sido dinámico ya que a lo largo del desarrollo han ido apareciendo diversas particularidades que han llevado a ello.

#### Diseño e implementación de la API

Para poder acceder a la base de datos desde la aplicación se ha diseñado e implementado una API REST mediante NodeJS. Esta API se encarga de gestionar las peticiones enviadas desde la aplicación y de devolver los datos requeridos o de efectuar cambios en la base de datos.

#### Gestión y acceso a la base de datos

A la hora de gestionar y acceder a los datos fue necesaria una labor de estudio debido a que al tratarse de una base de datos no relacional los alumnos no estaban tan familiarizados con sus características. Del mismo modo, requirió bastante tiempo diseñar las consultas necesarias para obtener los datos necesarios así como las sentencias que modifican dichos datos en caso de ser necesario.

#### Acceso a geolocalización, diseño del mapa e itinerarios

Para poder utilizar servicios de geolocalización en la aplicación y visualizar el mapa se tuvo que investigar qué servicio era más adecuado para la aplicación. Tras sopesar distintas opciones se optó por utilizar Leaflet [15] por ser gratuito y además de código abierto, sumado a la amplia documentación encontrada en la red.

De igual manera, se tuvo que realizar un trabajo de diseño para determinar la manera más adecuada de visualizar el mapa del jardín, parte vital del proyecto. Utilizando Gimp [27] y el mapa que aparece en la página web del Real Jardín botánico de Madrid ([http://www.rjb.csic.es/jardinbotanico/ficheros/documentos/pdf/didactica/plano\\_rjb\\_21.02.20\\_print.pdf](http://www.rjb.csic.es/jardinbotanico/ficheros/documentos/pdf/didactica/plano_rjb_21.02.20_print.pdf)) se logró alcanzar una visualización del mapa que fuera sencilla para los usuarios. Respecto a obtener el camino a seguir por el itinerario se ha utilizado el servicio de encaminamiento de Mapbox [17].

#### Pruebas y corrección de errores

Durante el desarrollo del proyecto fue necesario realizar pruebas para verificar el correcto funcionamiento del mismo, así como subsanar los errores que aparecían en el mismo.

### **Redacción de la memoria**

La redacción de la memoria ha sido realizada junto con Fabricio y con el apoyo del tutor del Trabajo de Fin de Grado.

## **9.2 Fabricio Bryan Torres Samueza**

La aportación de Fabricio al proyecto se define en los siguientes apartados:

### **Ingeniería de requisitos y selección de tecnologías**

Una parte esencial de este trabajo ha sido la ingeniería de requisitos. Al principio resultó algo costoso determinar correctamente el alcance que debía tener la aplicación para este Trabajo de Fin de Grado así como las funcionalidades que debía tener. Por ello, fueron necesarias reuniones con el tutor junto a Francisco para así acotar las funcionalidades necesarias.

Del mismo modo, determinar con qué tecnologías debía implementarse este trabajo fue también determinante para llegar a la completitud del mismo. Se ha necesitado de investigación y comparativas para alcanzar un consenso que permitiera un desarrollo eficaz y a la vez cómodo que llevase a cumplir con los objetivos marcados.

### **Diseño de la base de datos**

En la realización del diseño de la base de datos se estudiaron junto a Francisco varios diagramas de distintos tipos en busca del más adecuado para la aplicación considerando a su vez la tecnología escogida. Este diseño a su vez ha sido dinámico ya que a lo largo del desarrollo han ido apareciendo diversas particularidades que han llevado a ello.

### **Web scraping**

Nuestra aplicación tienen varias vistas en la que necesitamos mostrar información de la flora que posee el Real Jardín Botánico, para obtener esta información de manera sencilla y rápida, realice scraping (técnica para extraer la información que necesitamos de una página web de forma automatizada.) de la página web del Real Jardín Botánico.

Para poder recopilar esta información utilice javascript junto con Cheerio que es una biblioteca que nos permite analizar un documento html y poder extraer su información, una vez obtenido la información la guardé en un documento CSV para así poder tratarla y mostrarla en nuestra aplicación.

### **Estilos de la aplicación**

A la hora de diseñar la parte visual de la aplicación me centre: por una parte en que los colores estén acordes al tema de nuestra aplicación (para ello utilice colores que recuerden a la naturaleza y así no desentonar con el entorno que te ofrece el Real Jardín Botánico) y en que sea intuitiva y amigable con todos los usuarios para que así se lleven una buena experiencia a la hora de utilizar nuestra aplicación.

### **Estudio del estado del arte**

Para poder realizar la aplicación, realice un estudio general de aplicaciones semejantes a la que estábamos realizando para así poder dar un valor añadido a nuestra aplicación y diferenciarla de las ya existentes.

### **Redacción de la memoria**

La redacción de la memoria ha sido realizada junto con Francisco y con el apoyo del tutor del Trabajo de Fin de Grado.

## Bibliografía

- [1] Arbolapp. [En línea]. Disponible en: <https://www.arbolapp.es/>
- [2] PlantNet. [En línea]. Disponible en: <https://identify.plantnet.org/es>
- [3] RJB Museo vivo. [En línea]. Disponible en: <https://apps.apple.com/es/app/rjb-museo-vivo/id1136045846?l=en>
- [4] iKnow Trees 2 LITE. [En línea]. Disponible en: <https://iknow-trees-2-lite.es.aptoide.com/app>
- [5] Ionic Framework: Cross-Platform Mobile App Development. [En línea]. Disponible en: <https://ionicframework.com/>
- [6] Angular. [En línea]. Disponible en: <https://angular.io/>
- [7] Modelo-vista-controlador. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/Modelo-vista-controlador>
- [8] HTML5. [En línea]. Disponible en: <https://developer.mozilla.org/es/docs/Web/Guide/HTML/HTML5>
- [9] HTML. [En línea]. Disponible en: <https://developer.mozilla.org/es/docs/Web/HTML>
- [10] CSS3. [En línea]. Disponible en: <https://developer.mozilla.org/es/docs/Web/CSS>
- [11] Firebase. [En línea]. Disponible en: <https://firebase.google.com/>
- [12] Google Cloud Platform. [En línea]. Disponible en: <https://cloud.google.com/>
- [13] Firestore. [En línea]. Disponible en: <https://firebase.google.com/docs/firestore>
- [14] Realtime Database. [En línea]. Disponible en: <https://firebase.google.com/docs/database>
- [15] Leaflet. [En línea]. Disponible en: <https://leafletjs.com/>
- [16] GEOJson. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/GeoJSON>
- [17] Mapbox. [En línea]. Disponible en: <https://www.mapbox.com/>
- [18] NodeJS. [En línea]. Disponible en <https://nodejs.org/es/>
- [19] CouchDB. [En línea]. Disponible en: <https://couchdb.apache.org/>
- [20] Mapnik. [En línea]. Disponible en: <https://mapnik.org/>
- [21] GDAL. [En línea]. Disponible en: <https://gdal.org/>
- [22] Arquitectura orientada a eventos. [En línea]. Disponible en: [https://es.wikipedia.org/wiki/Programacion\\_dirigida\\_por\\_eventos](https://es.wikipedia.org/wiki/Programacion_dirigida_por_eventos)
- [23] Motor V8 de Google. [En línea]. Disponible en: [https://es.wikipedia.org/wiki/V8\\_\(Ingenieria\\_JavaScript\)](https://es.wikipedia.org/wiki/V8_(Ingenieria_JavaScript))
- [24] Node Package Manager. [En línea]. Disponible en: <https://www.npmjs.com/>
- [25] Visual Studio Code. [En línea]. Disponible en: <https://code.visualstudio.com/>
- [26] Git. [En línea]. Disponible en: <https://git-scm.com/>
- [27] GIMP. [En línea]. Disponible en: <https://www.gimp.org/>
- [28] Modelo cliente-servidor. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/Cliente-servidor>
- [29] Interfaz de programación de aplicaciones. [En línea]. Disponible en: [https://es.wikipedia.org/wiki/Interfaz\\_de\\_programacion\\_de\\_aplicaciones](https://es.wikipedia.org/wiki/Interfaz_de_programacion_de_aplicaciones)
- [30] Firebase Authentication. [En línea]. Disponible en: <https://firebase.google.com/docs/auth>
- [31] Web scraping. [En línea]. Disponible en: [https://es.wikipedia.org/wiki/Web\\_scraping](https://es.wikipedia.org/wiki/Web_scraping)
- [32] Google Forms. [En línea]. Disponible en <https://www.google.es/intl/es/forms/about/>

# Anexos

## Anexo I: Guía de uso

En este anexo se explica cómo utilizar la aplicación para los tres tipos de usuarios que la pueden usar. Primero se explican las funcionalidades que pueden realizar los usuarios no autenticados, luego la de los usuarios normales autenticados y para finalizar las de los usuarios autenticados de tipo administrador.

### 1. Menú principal sin autenticar

Cuando un usuario sin autenticar accede a la aplicación podrá ver el menú principal, ubicado en la ruta "home", tal y como se ve en la figura 108. En ella aparecen las 3 secciones principales a la que pueden acceder estos usuarios, representadas con 3 botones con sus respectivos nombres, y un botón en la parte superior izquierda que despliega el menú lateral, explicado más adelante.



Figura 108: Menú principal de la aplicación sin autenticar

### 2. Ver mapa

Pulsando sobre el botón "Ver mapa" ubicado en el menú principal se accede al mapa del jardín, ubicado en la ruta "map", como se puede observar en la figura 109. En este mapa aparece la ubicación del usuario siempre y cuando este se encuentre dentro de los confines del mismo.



Figura 109: Mapa del jardín

Pulsando sobre las parcelas del mapa se muestran las plantas que esta contiene, indicada por su nombre científico (ver figura 110).



Figura 110: Información sobre las parcelas del jardín

### 3. Lista de itinerarios

Para acceder a la lista de itinerarios disponibles se debe pulsar en el botón "Ver itinerarios" en el menú principal. Una vez hecho eso se dirige a la ruta "itineraries" donde se muestra la lista de itinerarios, evidenciado por la figura 111. En la parte superior se dispone de un buscador en el que introduciendo texto se filtran los resultados en base a la coincidencia con el nombre del itinerario.

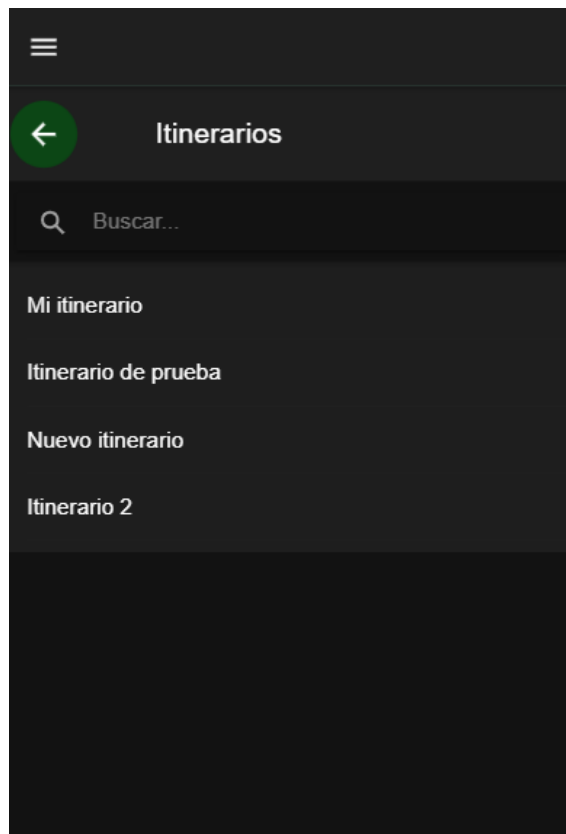


Figura 111: Lista de itinerarios

#### 4. Detalles itinerario sin autenticar

Pulsando sobre alguno de los itinerarios de la lista se podrá acceder a la vista detallada del mismo (ver figura 112), ubicada en la ruta "ruta", donde se podrá consultar el camino a recorrer, ver las opiniones de usuarios registrados en la aplicación e iniciar el camino pulsando sobre el botón "Iniciar itinerario". Al hacerlo obtendrá la ubicación del usuario y si este se encuentra dentro del jardín mostrará un marcador en el mapa indicando su ubicación. En caso contrario le notificará que no está dentro del jardín y no mostrará nada.

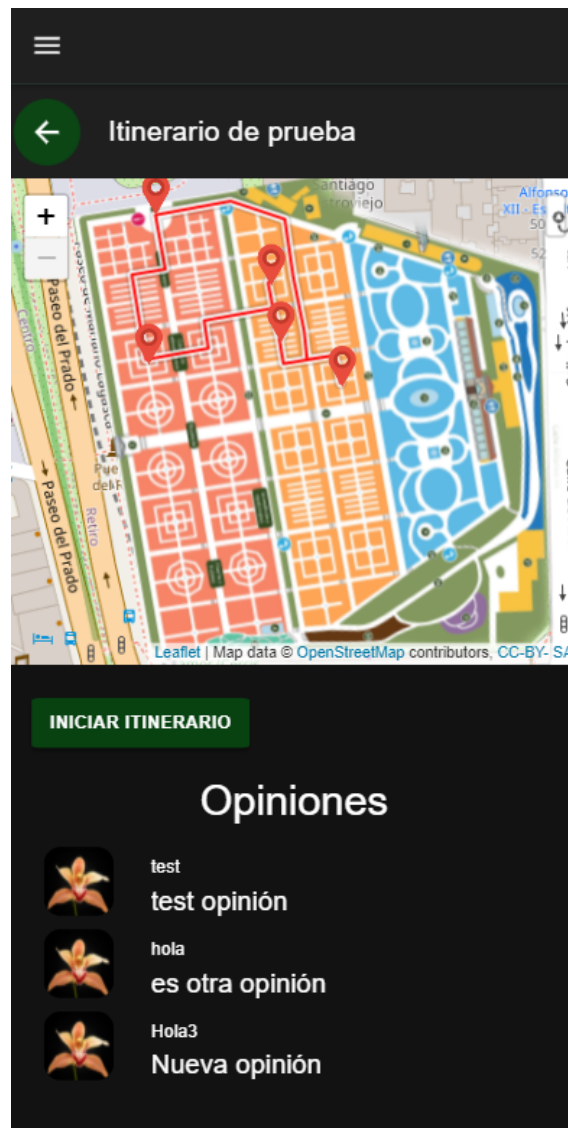


Figura 112: Vista detallada de un itinerario

Pulsando en el cuadrado blanco ubicado en la esquina superior derecha de la vista del mapa se desplegarán las instrucciones para realizar el itinerario junto a información sobre la distancia y el tiempo para completarlo (ver figura 113).



Figura 113: Instrucciones de un itinerario

## 5. Lista de plantas

Para consultar la lista de plantas del Real Jardín Botánico de Madrid se debe pulsar sobre el botón "Información sobre plantas" ubicado en el menú principal. Una vez hecho esto se accederá a la ruta "plants-info" y mostrará la lista de plantas que tiene el jardín (ver figura 114). En la parte superior de la lista se encuentra un buscador en el que al introducir texto filtra los resultados mostrando aquellos elementos cuyo nombre coincida con el texto introducido.

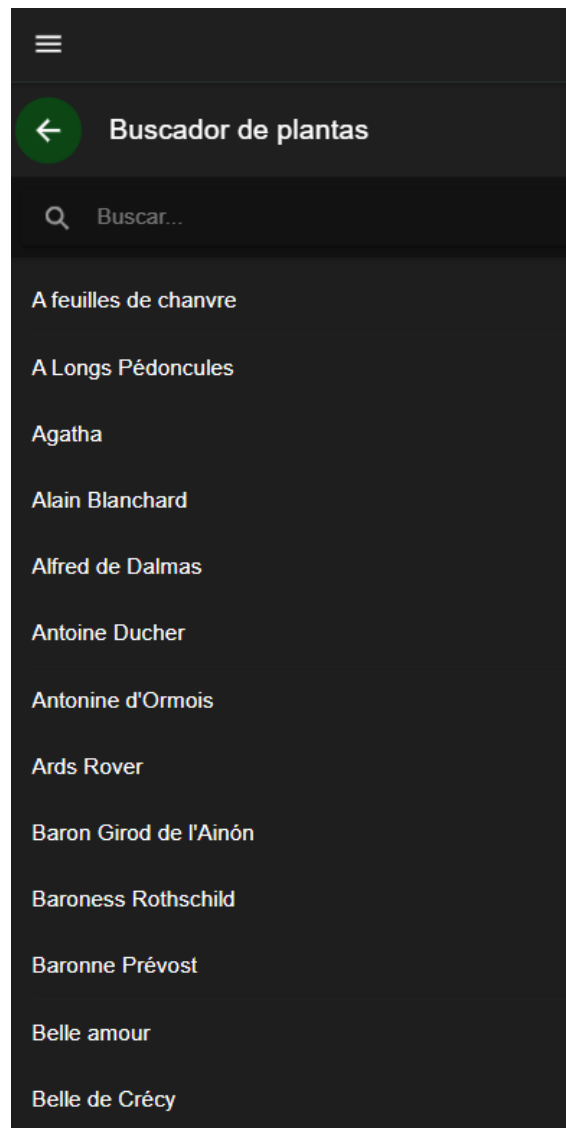


Figura 114: Lista de plantas

## 6. Información planta

Al pulsar sobre uno de los elementos de la lista de plantas se mostrará una ventana con información detallada sobre esta planta (ver figura 115).

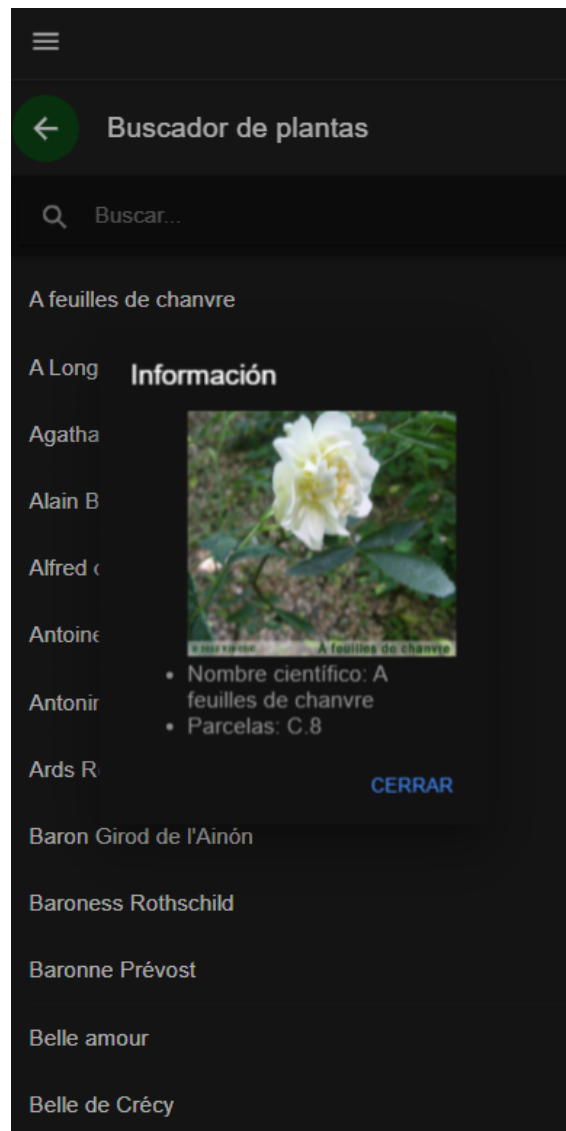


Figura 115: Información detallada de una planta

## 7. Menú lateral sin autenticar

Pulsando sobre el botón ubicado en la parte superior izquierda desde cualquier punto de la aplicación se despliega el menú lateral que, si el usuario no está autenticado, permite acceder al menú de inicio de sesión (ver figura 116). A su vez presenta un botón "Inicio" que al pulsarlo redirige al usuario al menú principal.

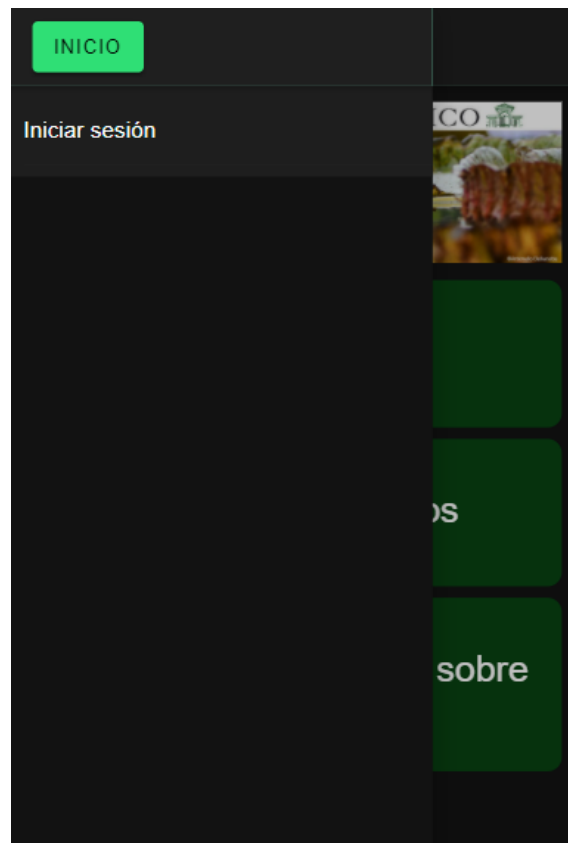


Figura 116: Menú principal de la aplicación sin autenticar

## 8. Inicio de sesión y registro

Al pulsar, desde el menú lateral y estando el usuario sin autenticar, en el botón "Iniciar sesión" se accede a la pantalla de inicio de sesión, ubicada en la ruta "login" (ver figura 117). En ella el usuario sin autenticar introduce sus credenciales (correo electrónico y contraseña) e inicia sesión en la aplicación, siendo redirigido al menú principal tras hacerlo.

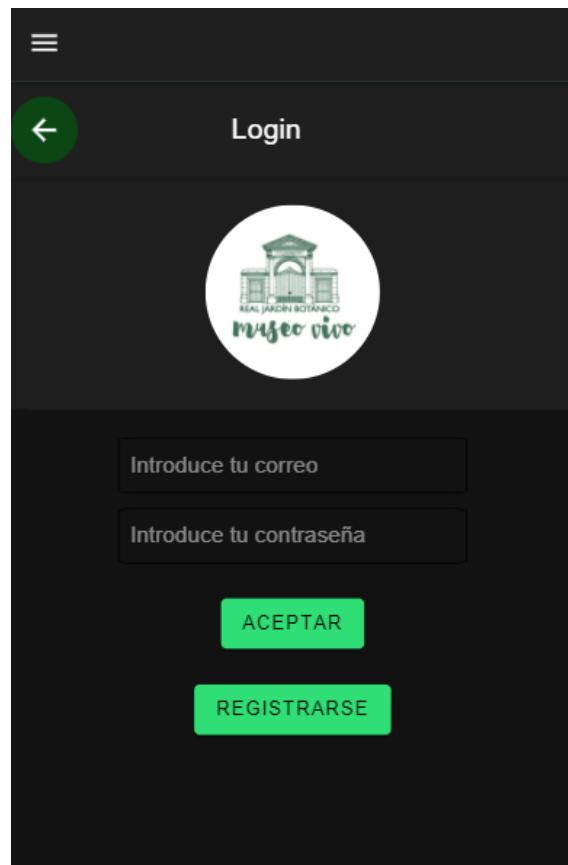


Figura 117: Inicio de sesión

Para acceder al menú de registro de usuario se debe pulsar sobre el botón "Registrarse" ubicado en el menú de inicio de sesión. Una vez pulsado se accede a la ruta "register" y se muestra este menú (ver figura 118) donde el usuario sin autenticar introduce los campos requeridos para darse de alta en la aplicación. Si los datos son válidos (nombre de usuario no repetido y con una longitud entre 5 y 12 caracteres, correo electrónico no registrado en la aplicación y contraseña de al menos 6 caracteres) el usuario es dado de alta en la aplicación, autenticado en la misma y redirigido al menú principal.

The image shows a mobile application registration screen. At the top left, there is a hamburger menu icon. Below it is a back arrow icon in a green circle. The title 'Registrarse' is centered at the top. The form consists of three input fields: 'Introduce tu nombre de usuario', 'Introduce tu correo', and 'Introduce tu contraseña'. Below the fields are two buttons: a green 'ACEPTAR' button and a red 'YA TENGO UNA CUENTA' button.

Figura 118: Registro de usuario

## 9. Menú principal autenticado

Los usuarios autenticados en la aplicación disponen de información adicional en el menú principal de la misma, siendo esta el nuevo botón "Crear itinerario" y un mensaje de bienvenida ubicado en la parte superior de la pantalla (ver figura 119).

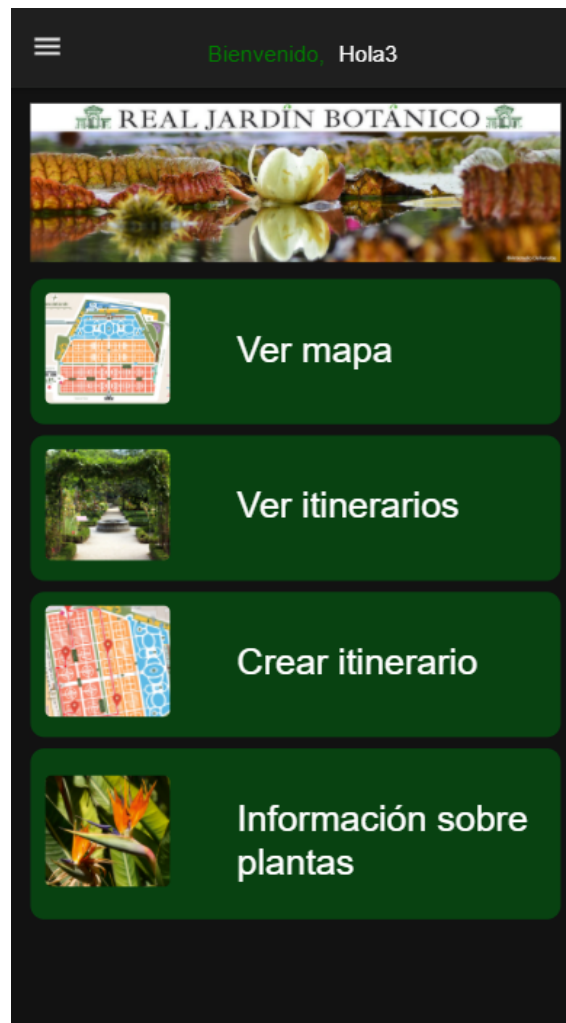


Figura 119: Menú principal de la aplicación autenticado

## 10. Detalles itinerario autenticado

Un usuario autenticado dispone de información adicional en la vista detallada de los itinerarios. Esta información adicional, como se ve en la figura 120, consiste en un botón para añadir el itinerario a favoritos, un botón para añadir/editar la opinión del usuario en el itinerario y, si el usuario autenticado es el mismo que el que ha creado el itinerario, un botón para eliminarlo.

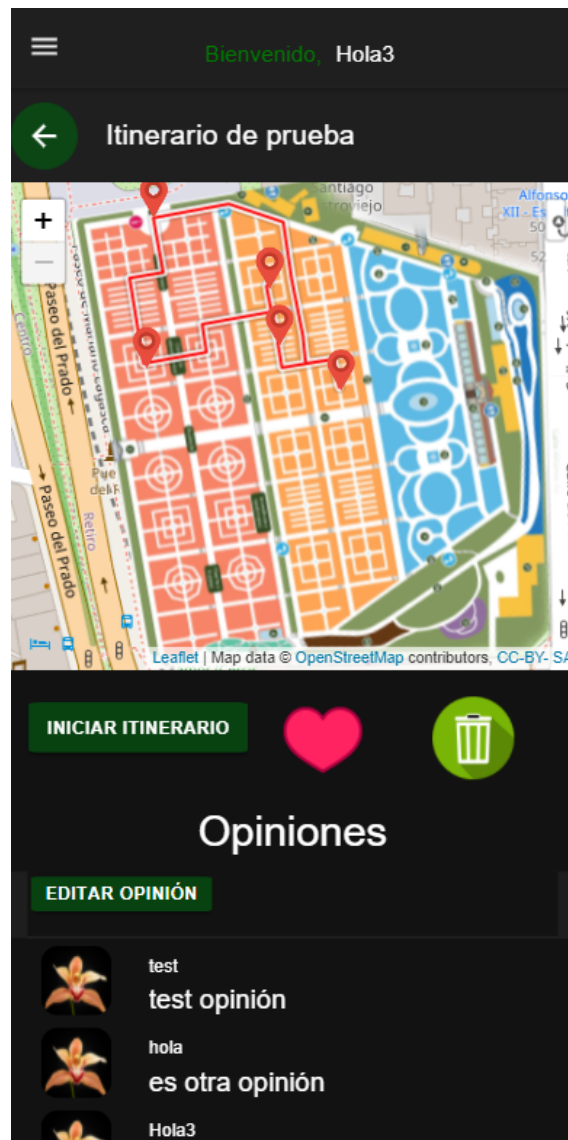


Figura 120: Vista detallada de un itinerario al estar autenticado I

En la figura 121 se muestra la vista descrita anteriormente en los casos en el que el usuario no sea el creador, no tenga el itinerario en favorito y no haya añadido una opinión aún.

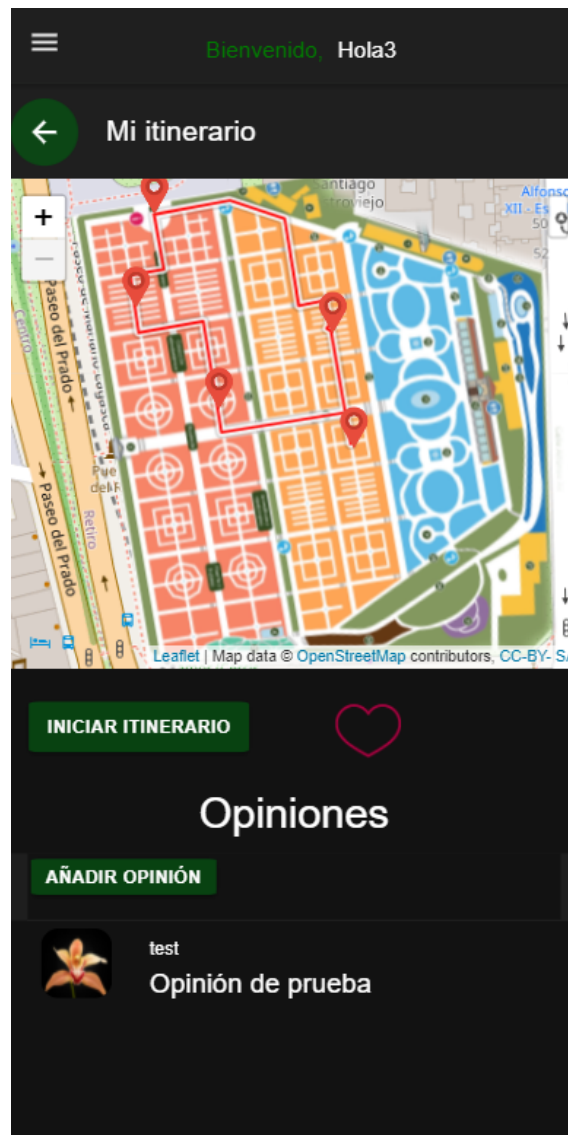


Figura 121: Vista detallada de un itinerario al estar autenticado II

En las figuras 122 y 123 se muestran las vistas para añadir una opinión y la comprobación para eliminar el itinerario respectivamente.

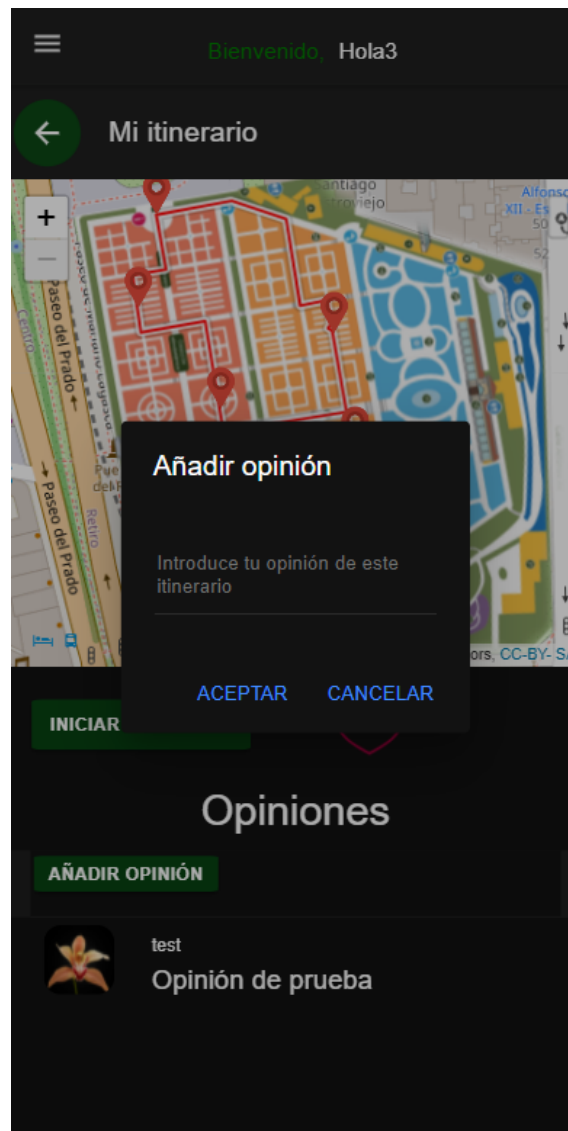


Figura 122: Añadir opinión

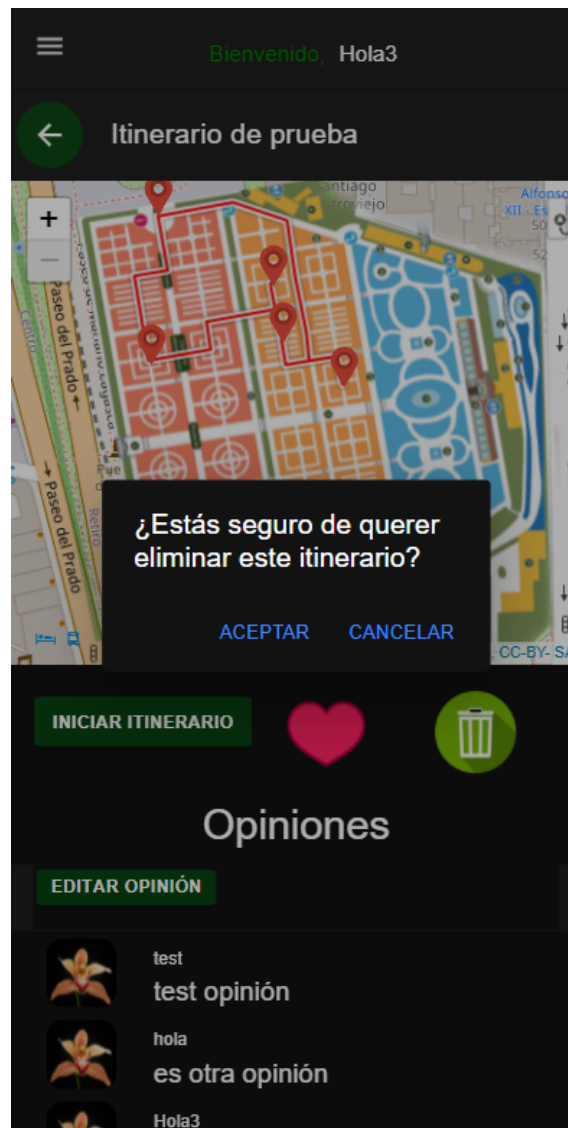


Figura 123: Eliminar itinerario

## 11. Crear itinerario

Al estar autenticado en la aplicación un usuario tiene la opción de crear itinerarios. Para acceder a ella debe pulsar en el botón "Crear itinerario" situado en el menú principal, desde donde se le redirige a la ruta "itinerary-maker" y se muestra la vista de crear itinerarios tal y como se ve en la figura 124.

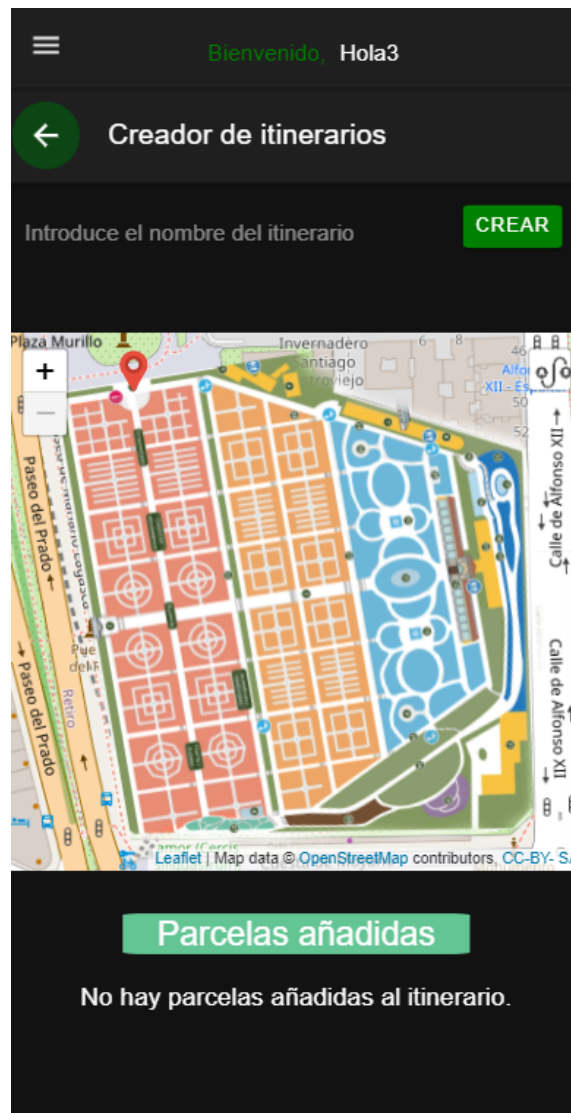


Figura 124: Menú de creación de itinerarios I

En esta vista al pulsar sobre cualquier parcela se despliega un menú en el que se muestra un botón para añadir dicha parcela al itinerario que se está creando, como se ve en la figura 125.

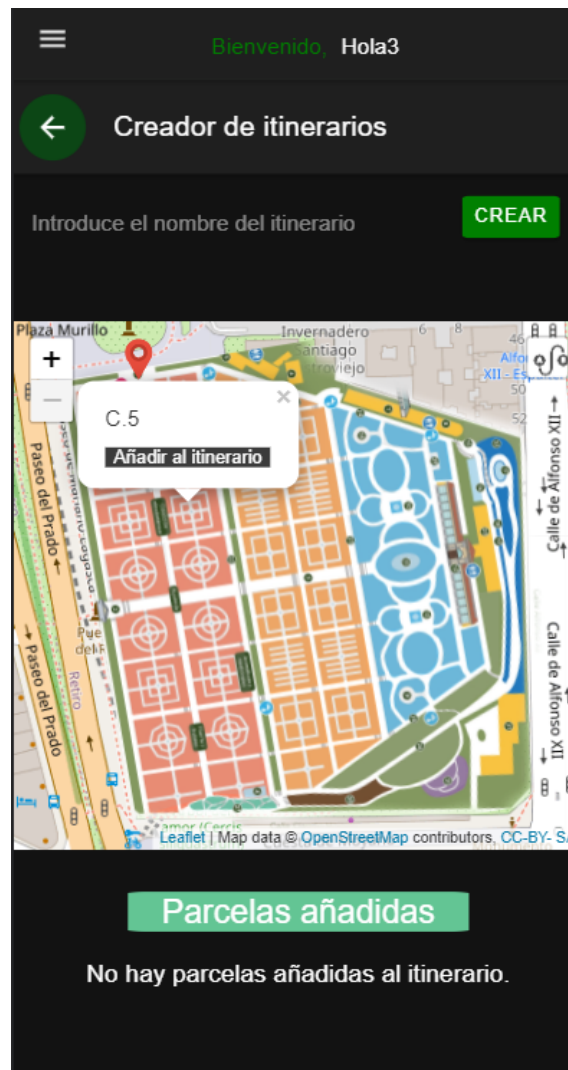


Figura 125: Menú de creación de itinerarios II

Según se van añadiendo parcelas al itinerario en el mapa se muestra el camino a seguir, mientras que en la parte inferior de la pantalla se muestra una lista con las parcelas incluidas (ver figura 126).

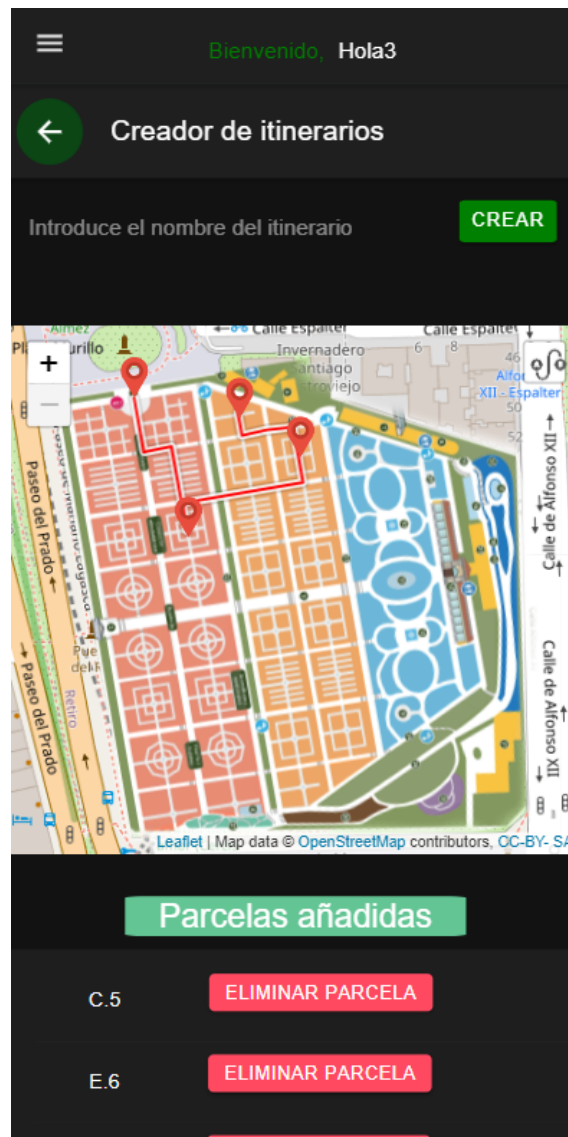


Figura 126: Menú de creación de itinerarios III

Pulsando sobre cualquiera de las parcelas añadidas se muestra un menú donde se visualiza el nombre de la parcela y un botón para eliminar dicha parcela del itinerario que se está creando. Esta acción también se puede realizar pulsando directamente sobre el botón "Eliminar parcela" situado en la lista inferior (ver figura 127).

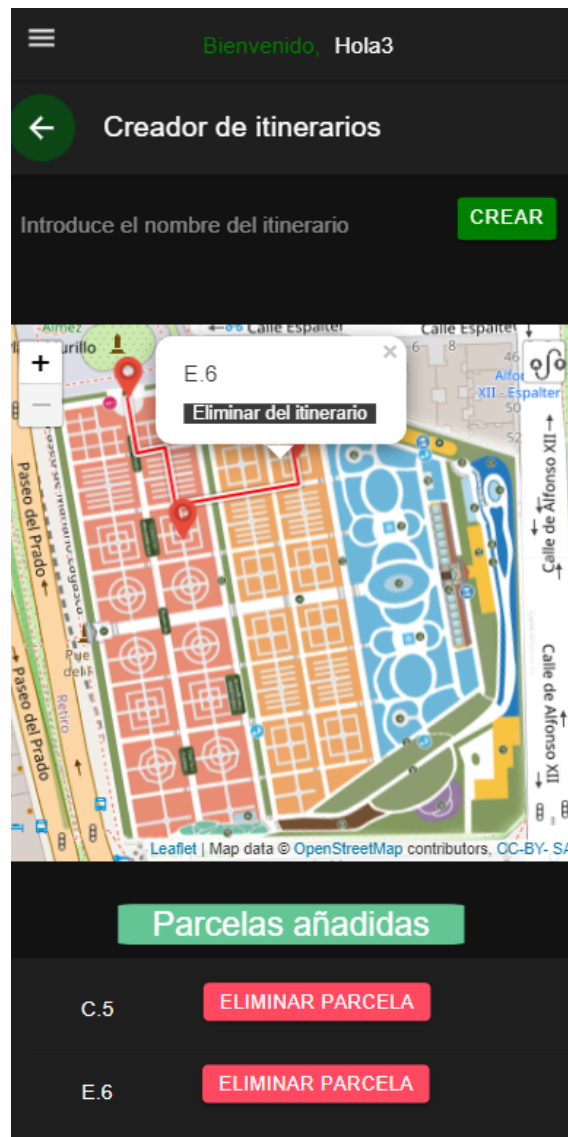


Figura 127: Menú de creación de itinerarios IV

El camino a seguir es recalculado automáticamente según se añaden y/o eliminan parcelas, como se puede ver en la figura 128.

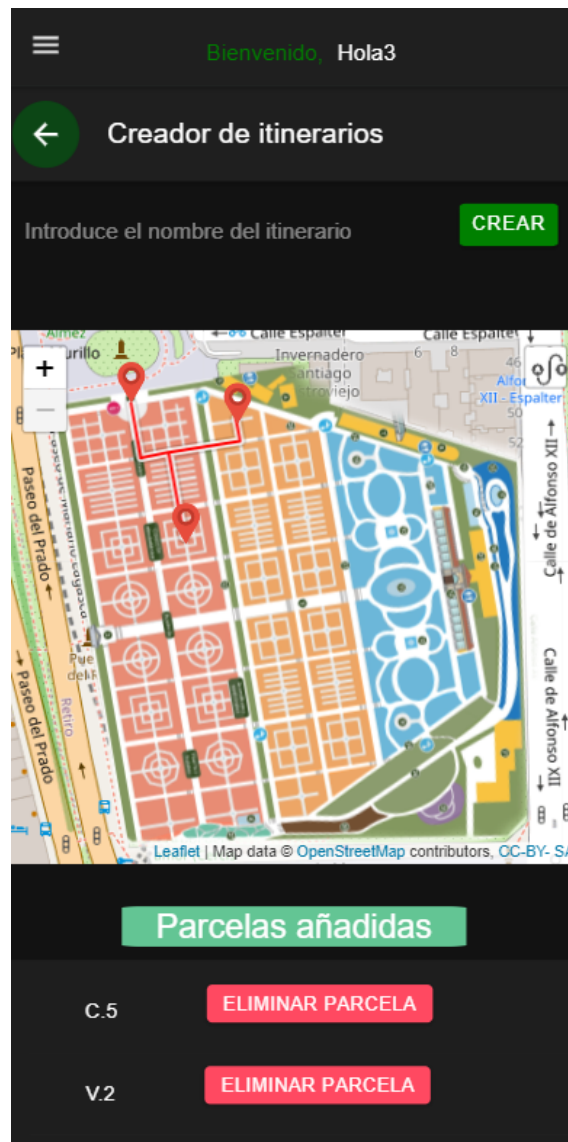


Figura 128: Menú de creación de itinerarios V

Una vez añadidas al menos 2 parcelas al itinerario y elegido un nombre para el mismo entre 1 y 20 caracteres se puede pulsar sobre el botón "Crear" situado en la parte superior derecha, se creará el itinerario y se redirigirá al usuario al menú principal donde se mostrará un mensaje de confirmación una vez añadido el itinerario al sistema, tal y como se ve en la figura 129.

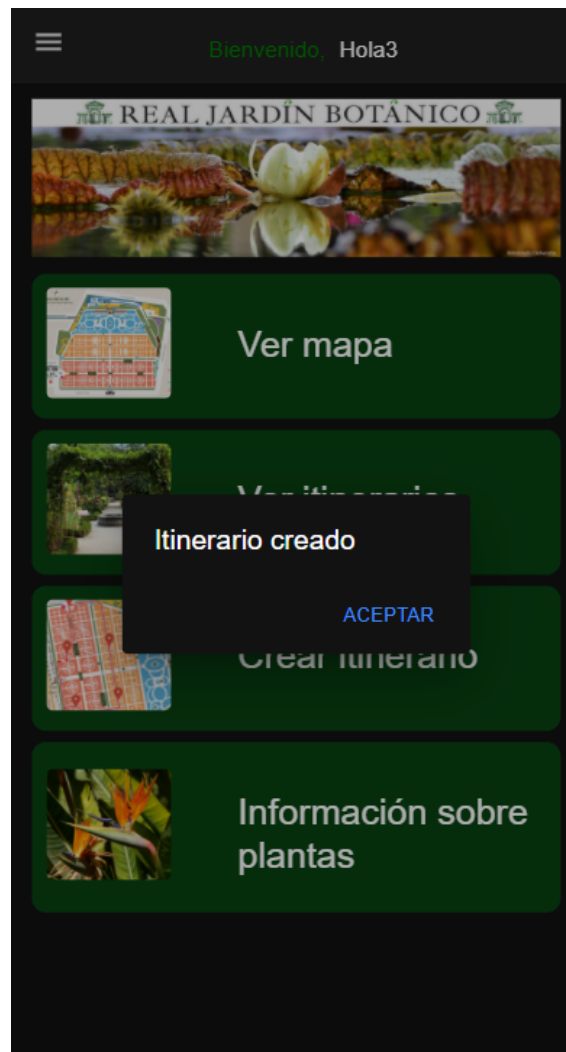


Figura 129: Menú de creación de itinerarios VI

## 12. Menú lateral autenticado y vista de perfil

Cuando un usuario se encuentra autenticado en la aplicación la vista del menú lateral cambia, mostrando el botón "Inicio" como hacía anteriormente pero cambiando el botón "Iniciar sesión" por los botones "Tu perfil" y "Cerrar sesión" (ver figura 130). Al pulsar sobre este último el usuario deja de estar autenticado en la aplicación y se le redirige al menú principal.

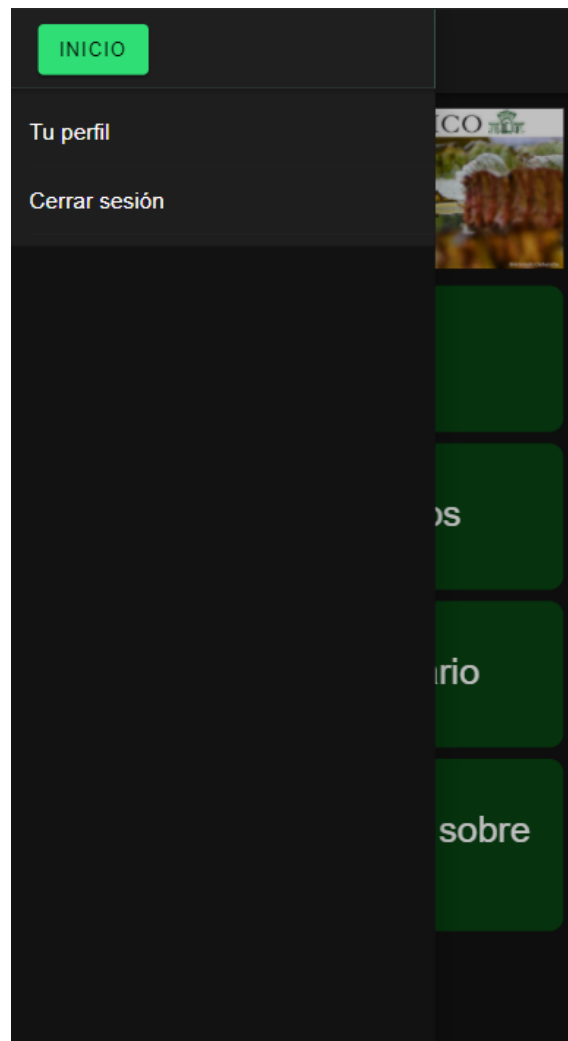


Figura 130: Menú lateral autenticado

Pulsando sobre el botón "Tu perfil" el sistema carga la ruta "profile" y muestra al usuario la vista correspondiente, donde puede observar una serie de botones que le otorgan acceso a diversas opciones para modificar y/o visualizar sus datos en la aplicación (ver figura 131).

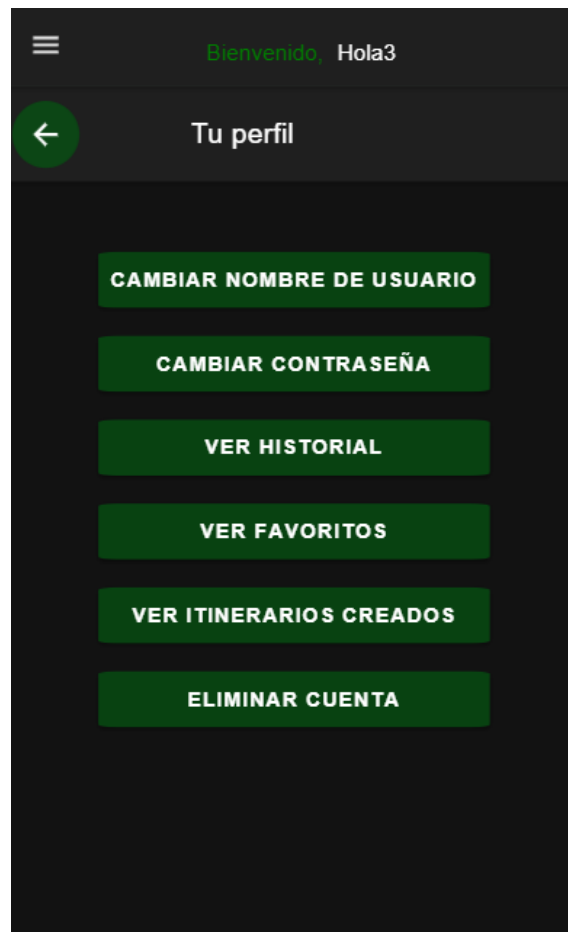


Figura 131: Vista de perfil

### 13. Cambiar nombre de usuario y/o contraseña

Desde la vista de perfil y pulsando sobre el botón "Cambiar nombre de usuario" se accede al formulario para modificar el nombre del usuario autenticado en la aplicación (ver figura 132), ubicado en la ruta "changeUsername". En este formulario se solicita el nuevo nombre del usuario y, si cumple las mismas condiciones que requería en el registro, se actualiza y se redirige al usuario al menú principal.

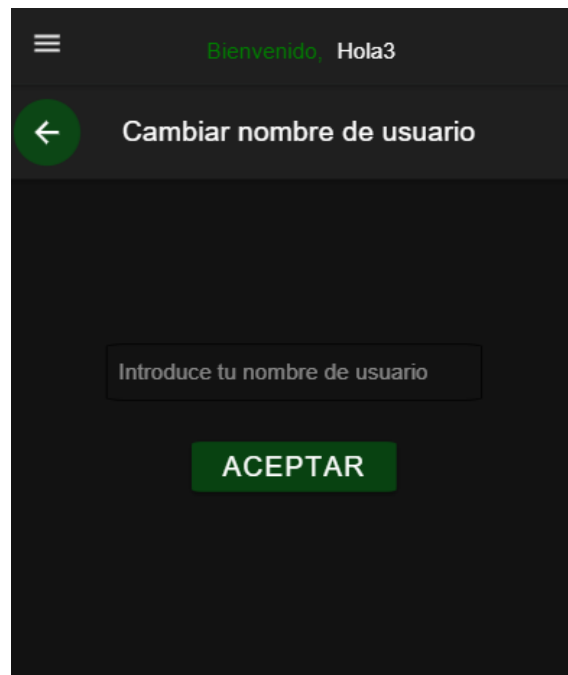


Figura 132: Cambiar nombre de usuario

Pulsando sobre el botón "Cambiar contraseña", ubicado en la vista de perfil, se accede al formulario para modificar la contraseña del usuario autenticado en la aplicación (ver figura 133), ubicado en la ruta "changePassword". Este formulario solicita la contraseña actual del usuario y la nueva contraseña elegida. Si la contraseña actual es correcta y la nueva cumple las mismas condiciones requeridas en el registro se actualiza la contraseña y se redirige al usuario al menú principal.

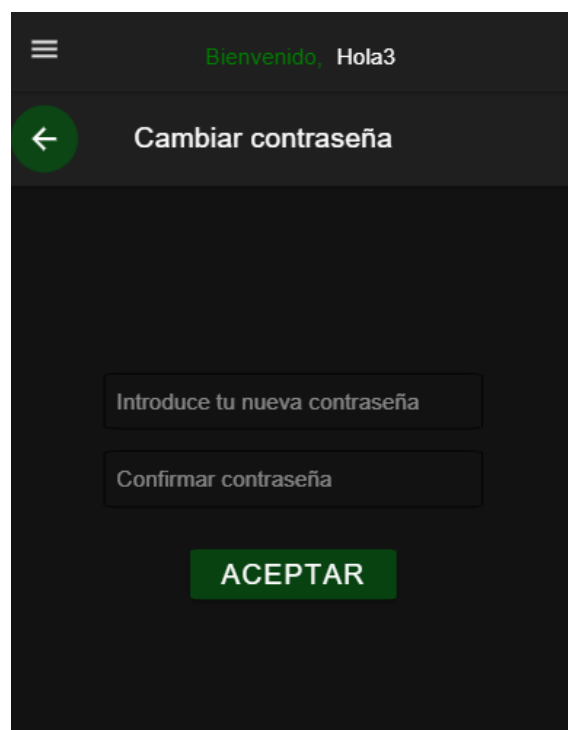


Figura 133: Cambiar contraseña

#### 14. Ver historial, favoritos e itinerarios creados

Desde la vista de perfil se dispone de 3 botones para acceder al historial del usuario, a su lista de itinerarios favoritos y a sus itinerarios creados: "ver historial", "ver favoritos" y "ver itinerar-

ios creados” respectivamente. Pulsando sobre ellos se accede a la ruta ”record”, ”favorites” o ”routesCreated” respectivamente y se muestra la lista correspondiente, como se ve en las figuras 134, 135 y 136 respectivamente. Pulsando sobre cualquiera de los itinerarios mostrados se accede a la vista detallada del mismo.

En el historial se muestran aquellos itinerarios que el usuario haya realizado, esto es, que haya pulsado sobre el botón ”Iniciar itinerario” del mismo.

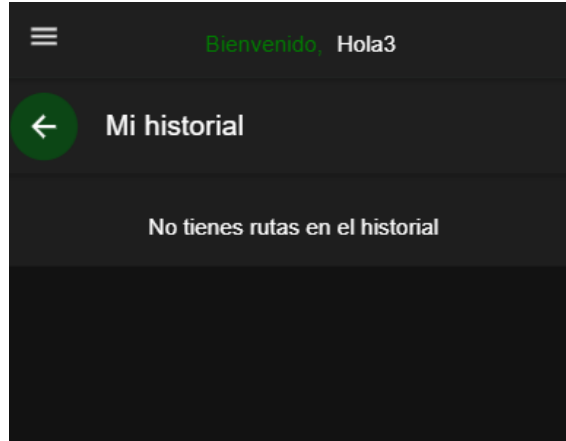


Figura 134: Historial

En la lista de favoritos se muestran los itinerarios que el usuario haya marcado como favoritos.

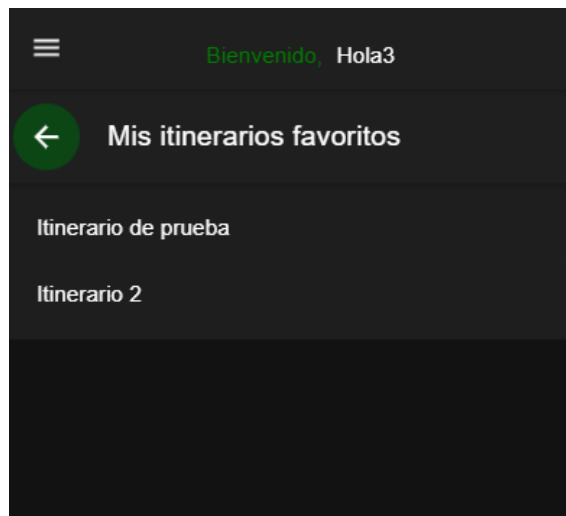


Figura 135: Favoritos

En la vista de itinerarios creados además de mostrar los itinerarios que ha creado el usuario autenticado se muestra una barra de búsqueda para filtrar los resultados de la lista.

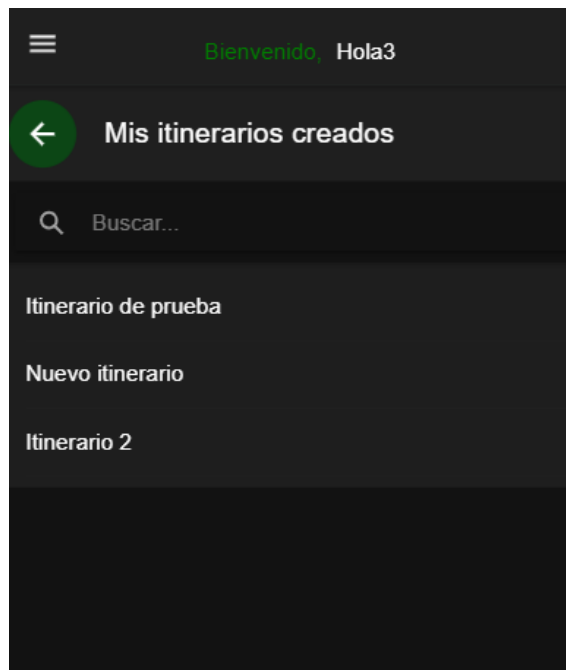


Figura 136: Itinerarios creados

## 15. Eliminar cuenta

Pulsando sobre el botón "Eliminar cuenta" situado en la vista de perfil se le solicitará confirmación al usuario para eliminar su cuenta (ver figura 137). En caso de hacerlo el usuario dejará de estar autenticado en el sistema, se le redirige al menú principal y se eliminan todos sus datos de la aplicación. Estos datos incluyen sus credenciales, nombre de usuario, opiniones e itinerarios creados, que serán eliminados a su vez de la lista de favoritos y/o historial de los otros usuarios en caso de tenerlos añadidos.

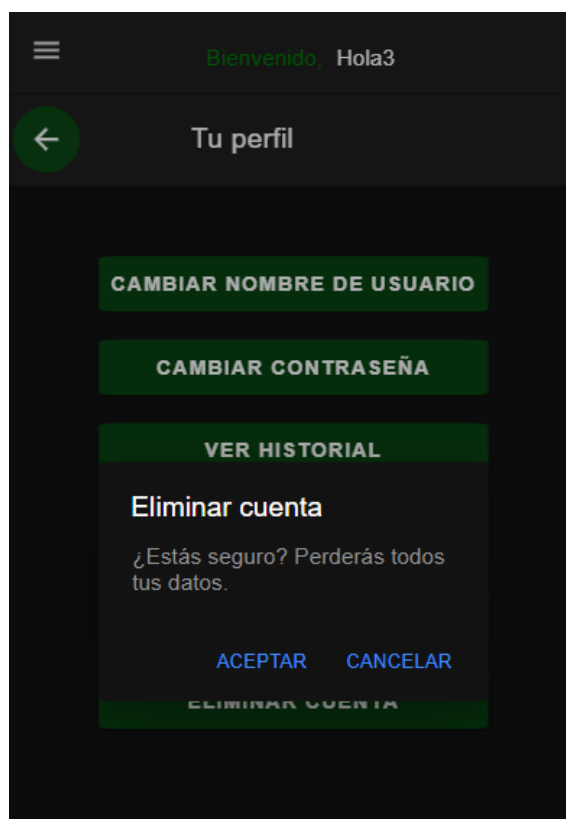


Figura 137: Eliminar cuenta

## 16. Menú principal de administrador

Iniciando sesión en la aplicación con una cuenta de tipo administrador se muestra el menú principal de este tipo de usuarios, diferente al de los usuarios normales tal y como se ve en la figura 138.



Figura 138: Menú principal de administrador

## 17. Lista de usuarios

Pulsando sobre el botón "Eliminar usuarios" mostrado en el menú principal se accede al listado de los usuarios registrados en el sistema, ubicado en la ruta "userList", como se ve en la figura 139. Esta vista muestra el nombre de usuario y el correo electrónico de todos los usuarios registrados en el sistema salvo el del propio usuario autenticado. En la parte superior se dispone de una barra de búsqueda con la que filtrar los resultados mostrados, ya sea por nombre de usuario o por correo electrónico. Pulsando sobre el icono verde con una papelera situado a la derecha se eliminará el usuario correspondiente del sistema, eliminando todos sus datos como se indica en la sección 15 de este anexo.

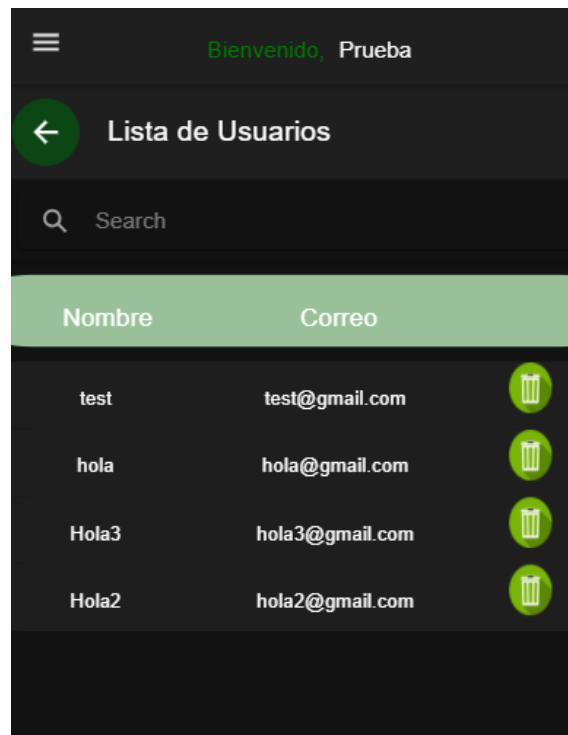


Figura 139: Lista de usuarios

## 18. Estadísticas

Desde el menú principal de administrador y pulsando sobre el botón "Ver estadísticas" se carga la ruta "stats" y se muestran las estadísticas sobre los itinerarios del sistema, como se visualiza en la figura 140. Estas estadísticas incluyen el usuario con mayor número de itinerarios creados y la cantidad de veces que una parcela se ha incluido en un itinerario.

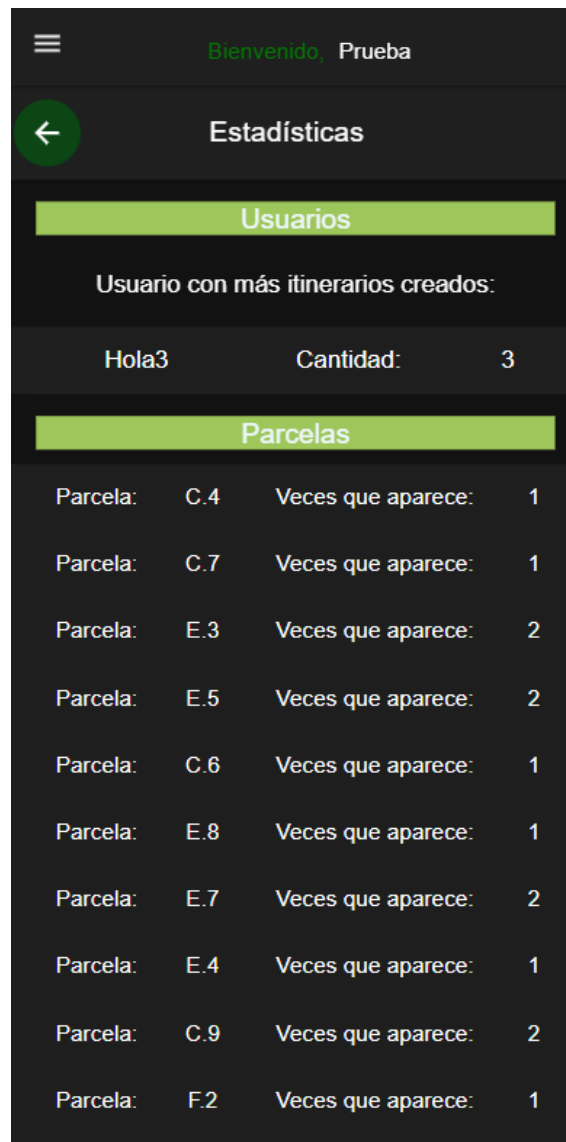


Figura 140: Estadísticas

## Anexo II: Guía de instalación de la aplicación

### 1. Instalación de la API

Para instalar la API es necesario tener NodeJS [18] en su versión 10.10.0 o superior ya que se trata de un servidor implementado con dicha tecnología, al igual que disponer del gestor de paquetes npm [24], instalado automáticamente con NodeJS. Del mismo modo es necesario disponer previamente del archivo JSON con las claves de acceso a la base de datos de FireBase [11] y FireStore [13]. Para ello se debe disponer una cuenta de Google y acceder a la consola de FireBase, accesible en este enlace: <https://console.firebase.google.com/>. En esta página se debe crear un proyecto para la aplicación. Una vez creado se debe acceder al apartado "Configuración del proyecto" y una vez ahí acceder a la pestaña "Cuentas de servicio". En este menú se pulsa sobre "Generar nueva clave privada". Esto genera un archivo JSON con las claves necesarias para conectar el servidor con FireBase y FireStore. Este archivo debe ser situado dentro de la carpeta `"/Backend/secret"` y ser renombrado a `"tfjgarden-firebase-adminsdk.json"` para que sea reconocido por el código del servidor.

Una vez realizados los pasos anteriores se debe abrir un terminal en la ruta `"/Backend/"` y ejecutar `"npm install"` para verificar que los paquetes necesarios estén instalados. Hecho todo esto el paso final consiste en ejecutar `"firebase emulators:start --only functions"` para lanzar a ejecución el servidor. Una particularidad importante de este proceso es que este servidor es ejecutado en local porque no se dispone de un servidor externo en el que ejecutarlo, por lo que solo se puede ejecutar de esta forma.

Un dato importante es que la aplicación no dispone de ningún modo de crear usuarios administradores, por lo que si se requiere de una cuenta de este modo es necesario registrar un nuevo usuario y desde la consola de FireBase acceder a la colección `"users"` y modificar el atributo `"isAdmin"` a un valor `true`.

### 2. Instalación de la aplicación móvil

Para ejecutar la aplicación es necesario disponer, como en el caso de la API, de NodeJS y npm instalados en el sistema, el primero en una versión igual o superior a la 10.10.0.

Una vez cumplidos estos requisitos se debe abrir un terminal en la ruta `"/Frontend/"` y ejecutar `"npm install"` para verificar que estén instalados todos los paquetes necesarios.

Para tener acceso a Mapbox y así acceder a su servicio de rutas es necesario tener una clave de acceso. Para obtenerla únicamente es necesario tener una cuenta en Mapbox, acceder a <https://account.mapbox.com/> y generar un token de acceso haciendo click sobre "Create a token". Una vez obtenido el token se debe acceder a la ruta `"/Frontend/src/app/secret"` y crear un archivo llamado `"mapBoxKey.ts"`, cuyo contenido será `"export const key = 'TOKEN'"` (sin las comillas dobles) donde `"TOKEN"` es el token generado en la página.

Una vez realizado todo esto se debe ejecutar `"ionic serve"`, que pasados unos segundos lanzará el navegador web predeterminado del sistema con la aplicación.

Una cosa de vital importancia es que la aplicación está pensada únicamente para su ejecución en local debido a que necesita que la API se esté ejecutando y esta no está disponible en un servidor externo. Por ello, no se dispone de un archivo de instalación para los dispositivos móviles.

Para poder visualizar la aplicación de manera correcta en el navegador es necesario acceder a las herramientas de desarrollador, accesibles mediante la combinación de teclas `"ctrl+shift+I"`. Una vez abierta la consola se debe activar la vista para dispositivos móviles con la combinación de teclas `"ctrl+shift+M"`.

## Anexo III: Campos del formulario de evaluación

El formulario empleado para elaborar la encuesta de evaluación con los usuarios viene descrito a continuación:

1. Datos generales del encuestado (ver figura 141).

The image shows a mobile application interface for a satisfaction questionnaire. The title is 'Cuestionario de satisfacción' and the subtitle is 'Cuestionario de satisfacción sobre aplicación para planificar visitas al Real Jardín Botánico de Madrid.' A red asterisk indicates that the following fields are mandatory. The form is divided into four sections: 1. 'Dirección de correo electrónico \*' with a text input field labeled 'Tu respuesta'. 2. 'Edad \*' with a text input field labeled 'Tu respuesta'. 3. 'Sexo \*' with four radio button options: 'Mujer', 'Hombre', 'Prefiero no decirlo', and 'Otro:'. 4. '¿Has visitado alguna vez el Real Jardín Botánico de Madrid? \*' with four radio button options: 'Sí, lo visito de manera habitual', 'Sí, lo he visitado en alguna ocasión', 'No, pero me gustaría visitarlo', and 'No, no me llama la atención'. At the bottom, there is a 'Siguiete' button, a progress bar, and the text 'Página 1 de 11'.

Figura 141: Sección 1 del formulario: datos generales del encuestado

2. Vista del mapa (ver figura 142).

## Cuestionario de satisfacción

\*Obligatorio

### 1. Vista del mapa

En el menú principal pulsa sobre el botón "Ver mapa". Pulsa sobre las distintas parcelas y observa la información que ofrecen. Vuelve al menú principal pulsando sobre la fecha ubicada en la parte superior izquierda.

Indica tu nivel de satisfacción con la vista del mapa \*

1 2 3 4 5 6 7 8 9 10

Muy bajo            Muy alto

Indica tu nivel de satisfacción con la información de las parcelas \*

1 2 3 4 5 6 7 8 9 10

Muy bajo            Muy alto

[Atrás](#) [Siguiente](#)  Página 2 de 11

Figura 142: Sección 2 del formulario: satisfacción con la vista del mapa

3. Consulta de itinerarios (ver figura 143).

## Cuestionario de satisfacción

**\*Obligatorio**

### 2. Consulta de itinerarios

En el menú principal pulsa sobre el botón "Ver itinerarios". Observa la lista de itinerarios que aparecen. Pulsa sobre la barra del buscador y escribe "Itinerario de prueba" para filtrar los resultados. Accede al itinerario que aparece y observa la información que muestra. Busca/consulta otros itinerarios si lo deseas. Vuelve al menú principal.

Indica tu nivel de satisfacción con la lista de itinerarios \*

1 2 3 4 5 6 7 8 9 10

Muy bajo            Muy alto

Indica tu nivel de satisfacción con el buscador de itinerarios \*

1 2 3 4 5 6 7 8 9 10

Muy bajo            Muy alto

Indica tu nivel de satisfacción con los datos del itinerario \*

1 2 3 4 5 6 7 8 9 10

[Atrás](#) [Siguiente](#) Página 3 de 11

Figura 143: Sección 3 del formulario: satisfacción con la consulta y vista de itinerarios

4. Consulta de información sobre plantas (ver figura 144).

## Cuestionario de satisfacción

\*Obligatorio

### 3. Consulta de información sobre plantas

En el menú principal pulsa sobre el botón "Ver información sobre plantas". Observa la lista de plantas que aparecen. Pulsa sobre la barra del buscador y escribe "Seigen" para filtrar los resultados. Pulsa sobre el elemento que aparece y observa la información que muestra. Busca y/o consulta otras plantas si lo deseas. Vuelve al menú principal.

Indica tu nivel de satisfacción con la lista de plantas \*

1	2	3	4	5	6	7	8	9	10
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Indica tu nivel de satisfacción con el buscador de plantas \*

1	2	3	4	5	6	7	8	9	10
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Indica tu nivel de satisfacción con la información ofrecida sobre las plantas \*

1	2	3	4	5	6	7	8	9	10
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Atrás
SiguientePágina 4 de 11

Figura 144: Sección 4 del formulario: satisfacción con la consulta y vista de plantas

5. Menú lateral, inicio de sesión y registro (ver figura 145).

**4. Menú lateral, inicio de sesión y registro**

En el menú principal pulsa sobre el botón situado en la parte superior izquierda de la aplicación y observa el desplegable que se muestra. Pulsa sobre iniciar sesión y a continuación consulta la vista de registro pulsando sobre el botón "Crear cuenta". Vuelve atrás pulsando sobre la flecha situada en la parte superior izquierda de la aplicación y en el menú de inicio de sesión autentícate con las siguientes credenciales:

- Correo: [usuario@gmail.com](mailto:usuario@gmail.com)
- Contraseña: 123456

Se te devolverá al menú principal donde podrás ver que aparece una nueva opción "Crear itinerario" además de un mensaje de bienvenida en la parte superior de la aplicación.

Indica tu nivel de satisfacción con el menú lateral \*

1 2 3 4 5 6 7 8 9 10

Indica tu nivel de satisfacción con el menú de inicio de sesión \*

1 2 3 4 5 6 7 8 9 10

Indica tu nivel de satisfacción con el menú de registro \*

1 2 3 4 5 6 7 8 9 10

[Atrás](#) [Siguiente](#) Página 5 de 11

Figura 145: Sección 5 del formulario: satisfacción con la consulta y vista de plantas

6. Crear itinerario (ver figura 146).

**Cuestionario de satisfacción**

\*Obligatorio

**5. Crear itinerario**

Con la sesión iniciada en la aplicación y en el menú principal pulsa sobre el botón "Crear itinerario". Pulsa sobre distintas parcelas y añade las que quieras al itinerario. Elimina las que quieras pulsando en el botón con el símbolo "menos" a la derecha del nombre de la parcela en la lista situada en la parte inferior. Vuelve al menú principal sin haber pulsado sobre el botón "Crear".

Indica tu nivel de satisfacción con el menú de creación de itinerarios \*

1 2 3 4 5 6 7 8 9 10

[Atrás](#) [Siguiente](#) Página 6 de 11

Figura 146: Sección 6 del formulario: satisfacción con la vista de crear itinerario

7. Nuevas opciones en los itinerarios (ver figura 147).

## Cuestionario de satisfacción

\*Obligatorio

### 6. Nuevas opciones en los itinerarios

Desde el menú principal pulsa sobre el botón "Ver itinerarios". En la barra del buscador escribe "itinerario de prueba" y accede a él. Observa los nuevos botones que aparecen: símbolo de corazón para añadir/eliminar de favoritos y botón para añadir una opinión.  
 Pulsa sobre el corazón para añadir el itinerario a favoritos y vuelve a pulsar sobre él para eliminarlo de favoritos.  
 Pulsa sobre el botón "Añadir opinión" para añadir una opinión al itinerario y una vez añadida editala y/o elimínala pulsando sobre "Editar opinión". Vuelve al menú principal.

Indica tu nivel de satisfacción con el botón para añadir/eliminar de favoritos \*

1   2   3   4   5   6   7   8   9   10  
                          

Indica tu nivel de satisfacción con el botón para añadir/editar/eliminar opiniones de los itinerarios \*

1   2   3   4   5   6   7   8   9   10  
                          

Atrás
SiguientePágina 7 de 11

Figura 147: Sección 7 del formulario: satisfacción con la nuevas opciones de los itinerarios al estar autenticado

8. Opciones de perfil de usuario (ver figuras 148 y 149).

## Cuestionario de satisfacción

\*Obligatorio

### 7. Opciones de perfil de usuario

En el menú principal pulsa sobre el botón situado en la parte superior izquierda de la aplicación y pulsa sobre el botón "Tu perfil". Observa las distintas secciones ofrecidas y accede a ellas, sin pulsar sobre "Eliminar cuenta". Una vez observadas, vuelve a abrir el menú lateral y pulsa sobre "Cerrar sesión".

Indica tu nivel de satisfacción con la vista "Tu perfil" \*

1   2   3   4   5   6   7   8   9   10

○ ○ ○ ○ ○ ○ ○ ○ ○ ○

Indica tu nivel de satisfacción con el menú "Cambiar nombre de usuario" \*

1   2   3   4   5   6   7   8   9   10

○ ○ ○ ○ ○ ○ ○ ○ ○ ○

Indica tu nivel de satisfacción con el menú "Cambiar contraseña" \*

1   2   3   4   5   6   7   8   9   10

○ ○ ○ ○ ○ ○ ○ ○ ○ ○

Figura 148: Sección 8.1 del formulario: satisfacción con las opciones de perfil de usuario I

Indica tu nivel de satisfacción con el menú "Ver historial" \*

1   2   3   4   5   6   7   8   9   10

○ ○ ○ ○ ○ ○ ○ ○ ○ ○

Indica tu nivel de satisfacción con el menú "Ver favoritos" \*

1   2   3   4   5   6   7   8   9   10

○ ○ ○ ○ ○ ○ ○ ○ ○ ○

Indica tu nivel de satisfacción con el menú "Ver itinerarios creados" \*

1   2   3   4   5   6   7   8   9   10

○ ○ ○ ○ ○ ○ ○ ○ ○ ○

Atrás
Siguiente

Página 8 de 11

Figura 149: Sección 8.2 del formulario: satisfacción con las opciones de perfil de usuario II

9. Administrador (ver figura 150).

### 8. Administrador

En el menú principal pulsa sobre el botón situado en la parte superior izquierda de la aplicación y observa el desplegable que se muestra. Pulsa sobre iniciar sesión y autentícate con las siguientes credenciales:

- Correo: [admin@gmail.com](mailto:admin@gmail.com)
- Contraseña: 123456

Observa el nuevo menú principal. Pulsa sobre "Lista de usuarios" y en este menú no pulses sobre la papelera situada a la derecha de cada uno. Vuelve al menú principal y pulsa sobre "Ver estadísticas" y observa las estadísticas que se ofrecen.

Despliega el menú lateral y pulsa sobre "Cerrar sesión".

Indica tu nivel de satisfacción con el menú principal de administrador \*

1 2 3 4 5 6 7 8 9 10

Indica tu nivel de satisfacción con la lista de usuarios \*

1 2 3 4 5 6 7 8 9 10

Indica tu nivel de satisfacción con las estadísticas ofrecidas \*

1 2 3 4 5 6 7 8 9 10

[Atrás](#) [Siguiente](#) Página 9 de 11

Figura 150: Sección 9 del formulario: satisfacción con la vista y opciones del administrador

10. Satisfacción general con la aplicación (ver figura 151).

**Cuestionario de satisfacción**

**\*Obligatorio**

**9. Satisfacción general con la aplicación**

Indica tu nivel de satisfacción (en general) con el diseño de la aplicación \*

1 2 3 4 5 6 7 8 9 10

Indica tu nivel de satisfacción (en general) con la fluidez de la aplicación \*

1 2 3 4 5 6 7 8 9 10

¿Tienes alguna sugerencia y/o comentario?

Tu respuesta

**Atrás** **Siguiente**  **Página 10 de 11**

Figura 151: Sección 10 del formulario: satisfacción general con la aplicación

11. Fin (ver figura 152).

**Cuestionario de satisfacción**

**10. Fin**

Muchas gracias por participar en esta encuesta.

**Atrás** **Enviar**  **Página 11 de 11**

Figura 152: Sección 11 del formulario: agradecimientos