

Birdwatching en las Marismas de Huelva





Objetivos

- Utilización de herramientas para desarrollo app
- Análisis y diseño de estructura de datos.
- Estudio de algoritmos para segmentar imágenes
- Planificación

Recursos

- App Inventor 2
- Firebase
- MatLab
- Personal Image Classifier
- Virtual Paradigm Online

Organización

- Análisis de aplicaciones similares
- Planificación, análisis, diseño, implementación y pruebas
- Conclusiones

REVISION DE APLICACIONES DE AVISTAMIENTO DE AVES



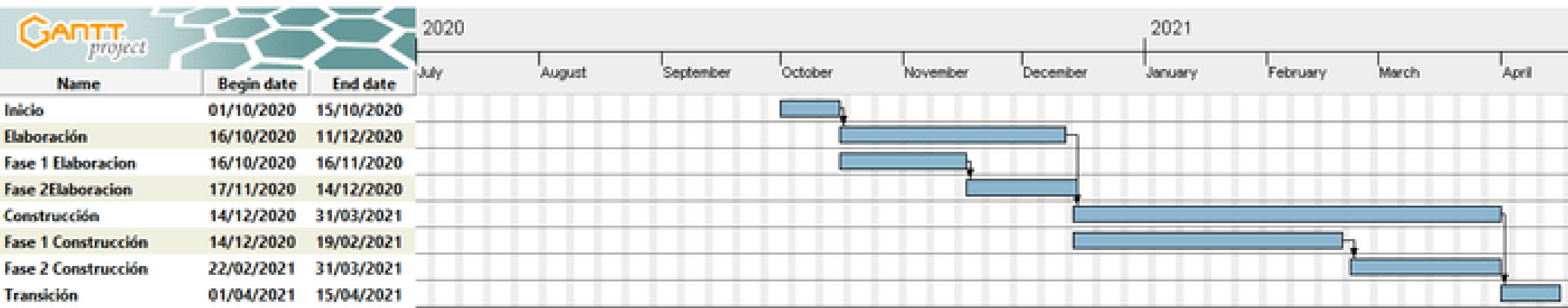
Aves de España

eBird

eBird

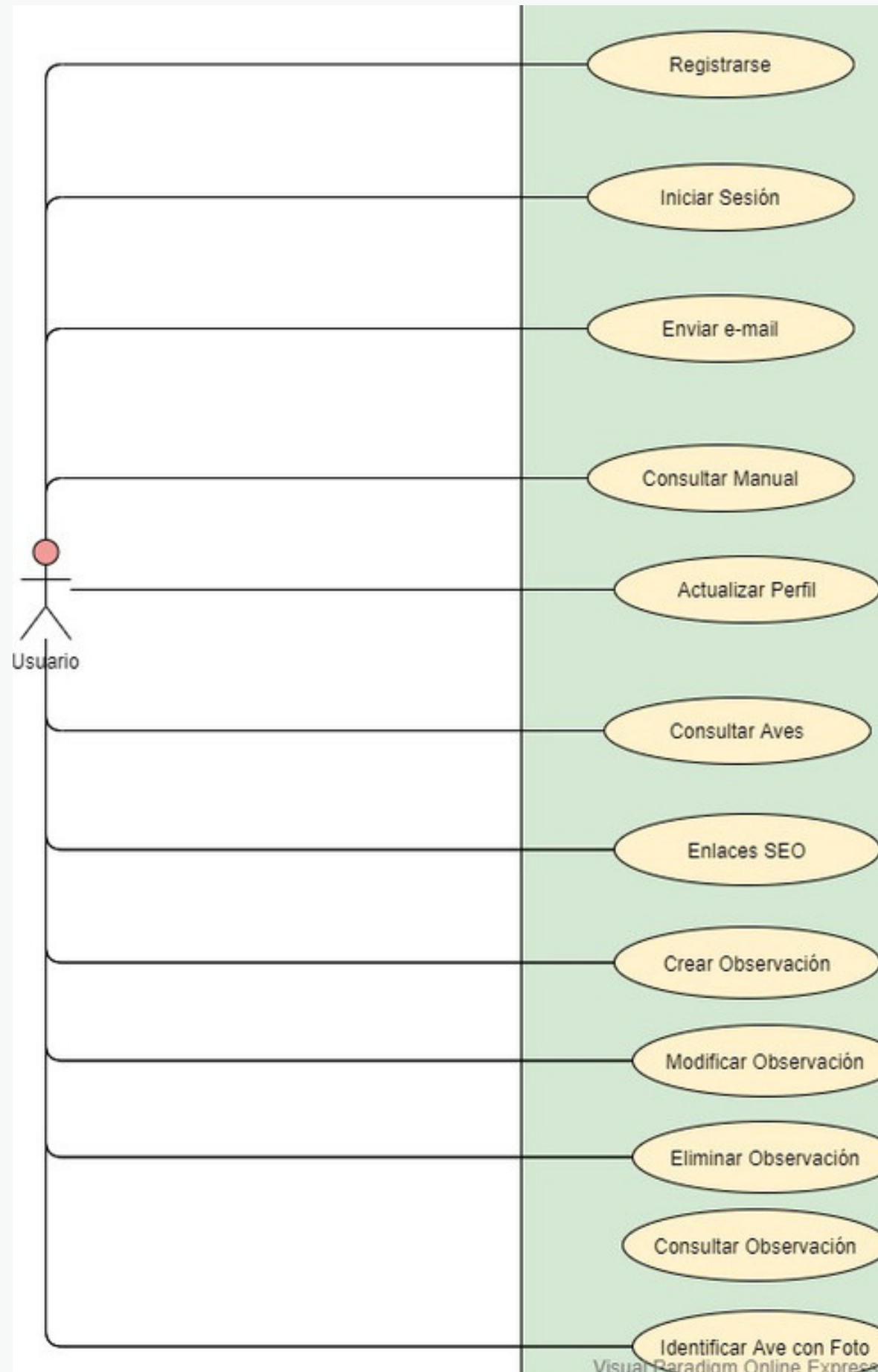


Merlin

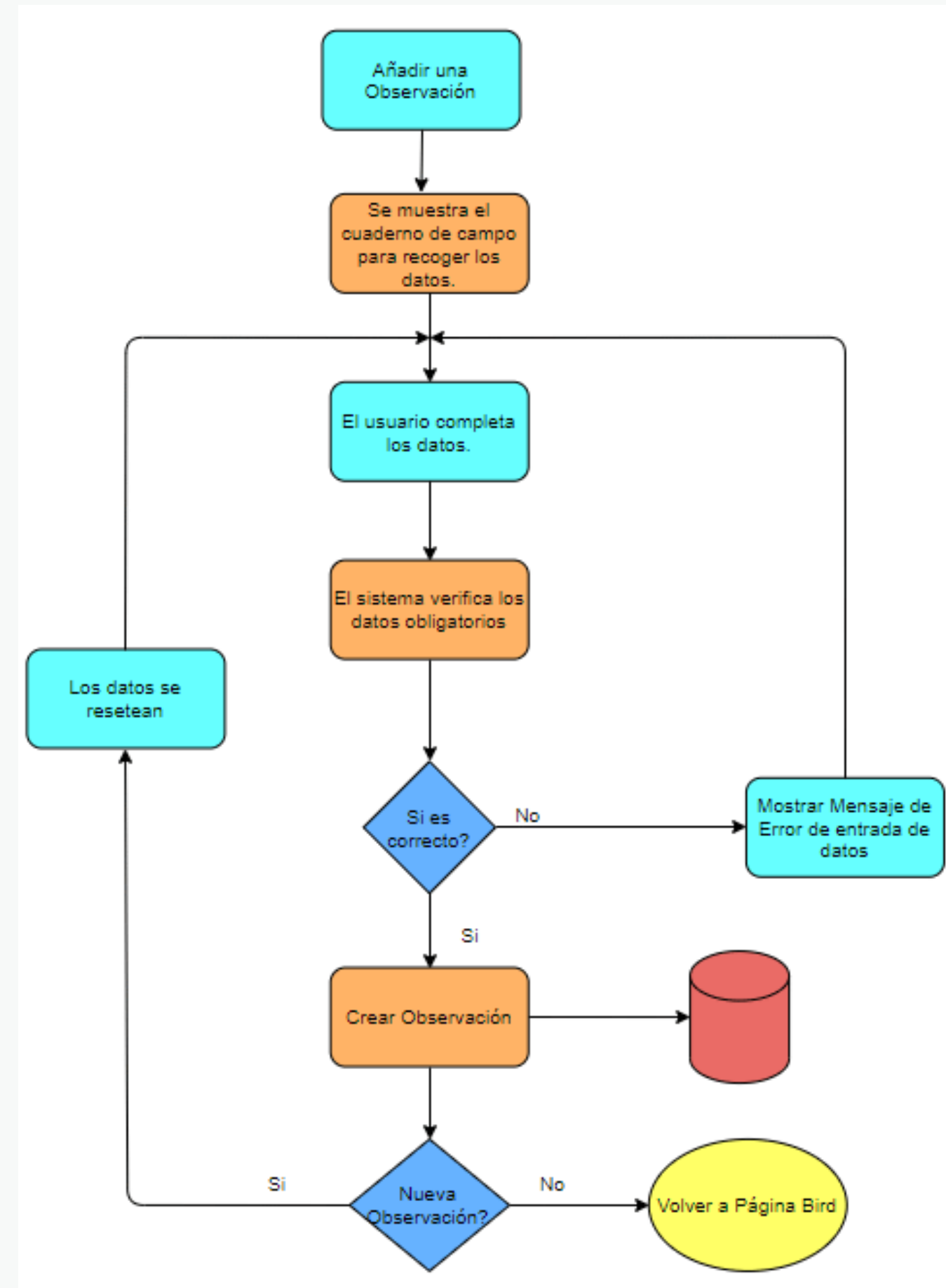


PLANIFICACIÓN Y ANÁLISIS DE REQUISITOS.

CASOS DE USO



DIAGRAMAS DE ACTIVIDAD



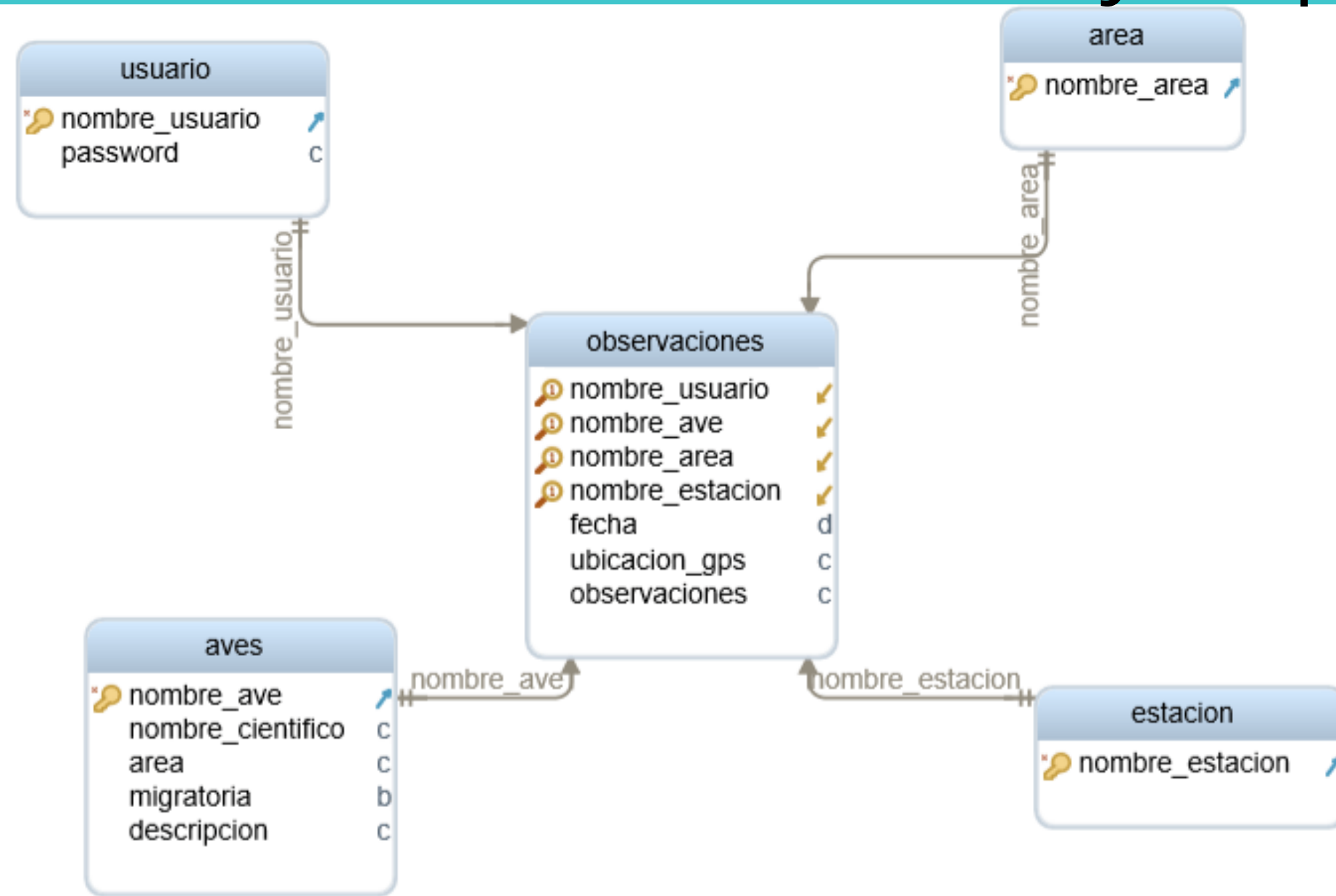
CREAR UNA OBSERVACIÓN

Especificación de un caso de uso



Nombre	Crear Observación
Autor	José Luis Campero Romero
Descripción	El sistema deberá mostrar la pantalla de añadir una observación cuando el usuario pulse la opción de cuaderno de campo y a continuación Añadir Observación.
Precondición	Estar identificado con su usuario y contraseña y seleccionado la opción de cuaderno de campo.
Secuencia	<ol style="list-style-type: none">1. El actor pulsa el botón de Añadir Observación2. El sistema solicita al usuario una serie de datos relacionados con el avistamiento, además de poder capturar una foto, audio y localización.3. El actor introduce los datos solicitados y almacena la observación.4. El sistema almacena los datos y avisa al usuario que la operación se ha llevado con éxito.
Excepciones	<ol style="list-style-type: none">1. Si se olvidan algunos datos, el sistema los validará y solicitará al usuario que rellene algunos datos obligatorios.2. Este caso de uso continúa con nuevas observaciones.
Postcondición	Se ha añadido una nueva observación.

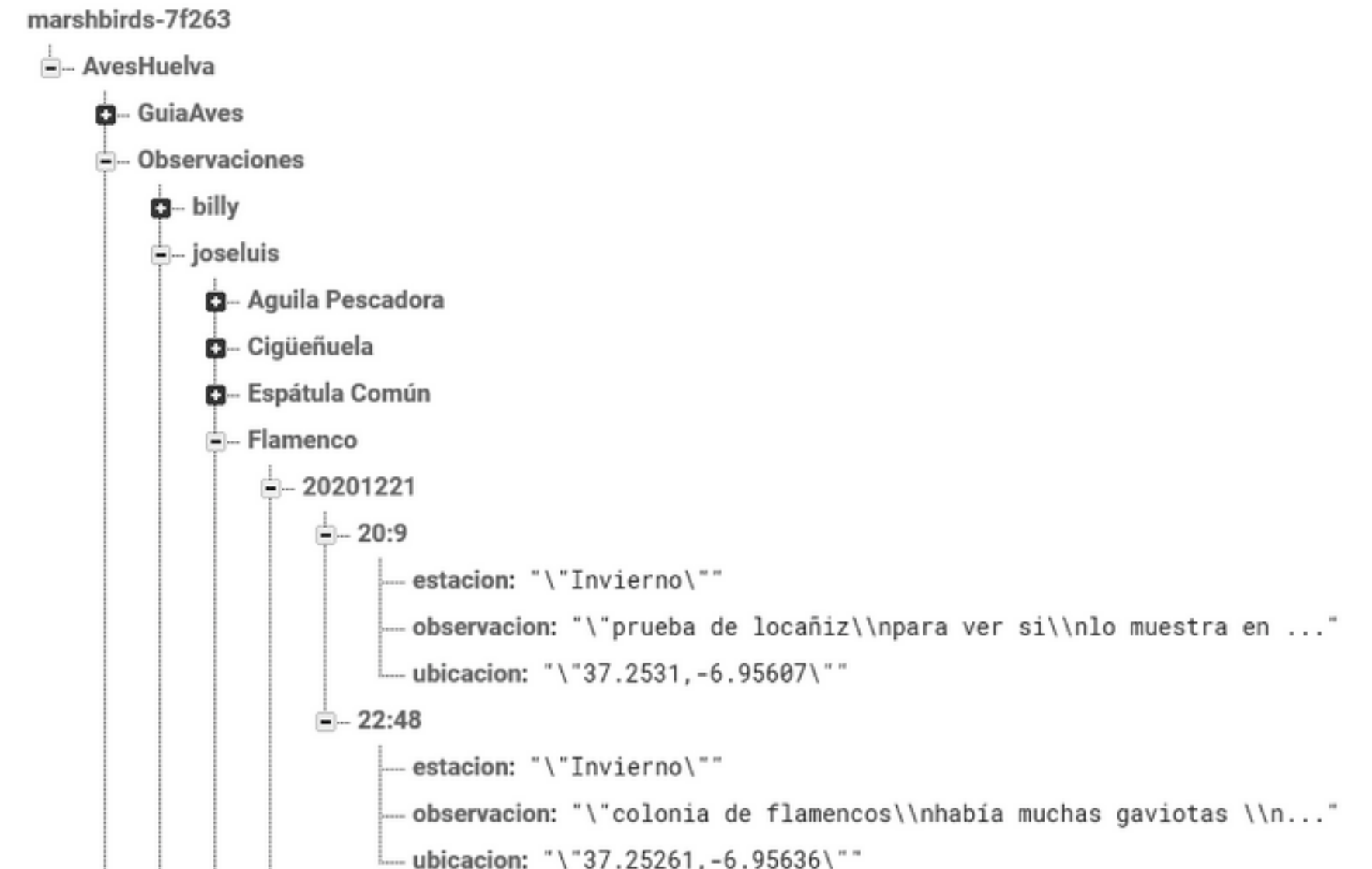
Diseño y Arquitectura



BASE DE DATOS RELACIONAL

Diseño de la Base de Datos:
Relacional versus JSON.

BASE DE DATOS JSON



Realtime DataBase y Storage

1

REGLAS DE REALTIME DATABASE

```
1 {
2   "rules": {
3     ".read": true, // 2020-10-17
4     ".write": true, // 2020-10-17
5     "AvesHuelva" : {
6       ".indexOn": "GuiaAves"
7     }
8   }
9 }
10
```

2

PRIMER NIVEL DE DATOS

<https://marshbirds-7f263.firebaseio.com/>

marshbirds-7f263

└─ AvesHuelva

└─ GuiaAves

└─ Observaciones

└─ usuario

3

REGLAS STORAGE

```
1 service firebase.storage {
2   match /b/marshbirds-7f263.appspot.com/o {
3     match /{allPaths=**} {
4       allow read, write: if true;
5     }
6   }
7 }
8
```

4

PRIMER NIVEL STORAGE

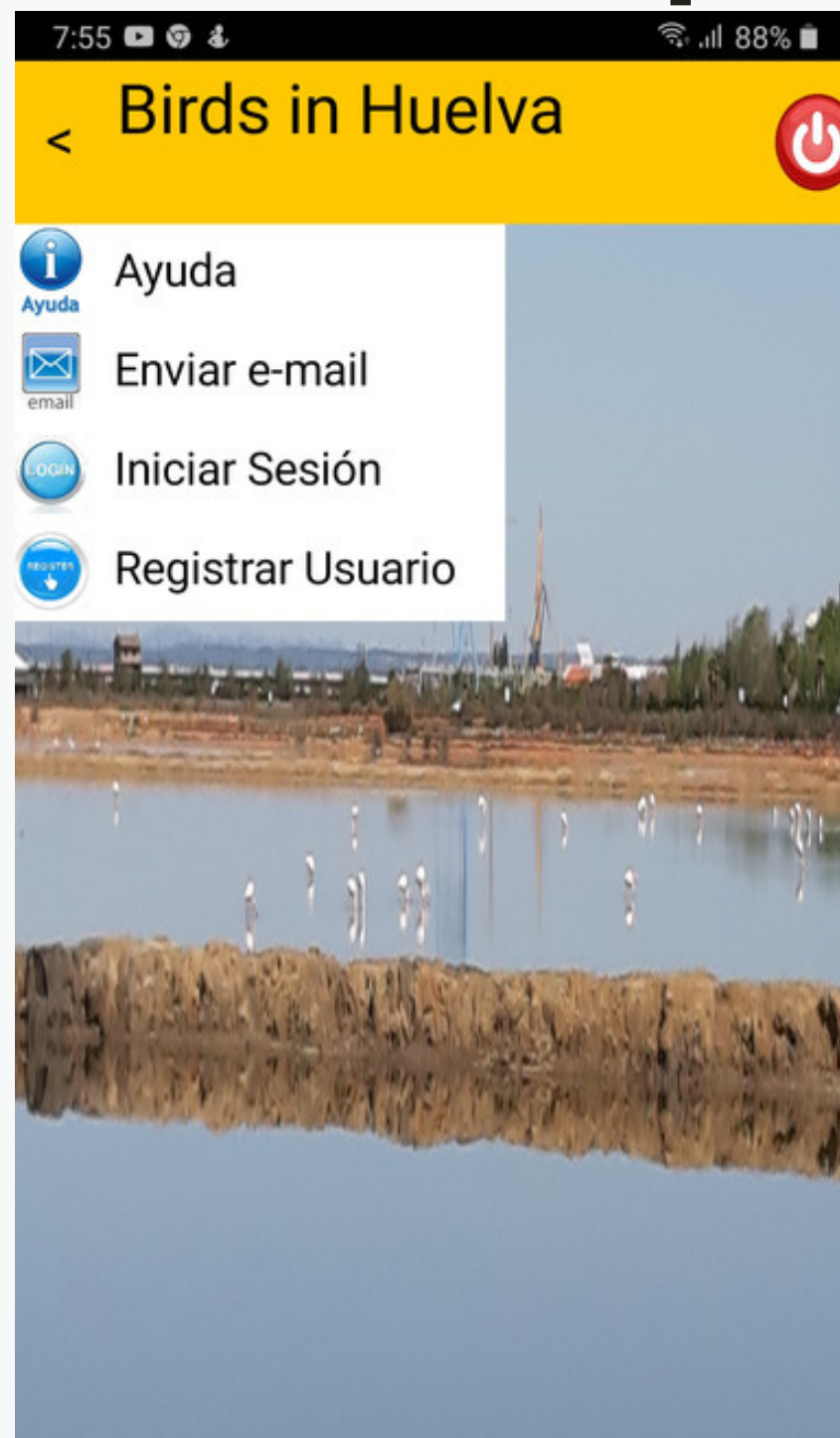
<gs://marshbirds-7f263.appspot.com>

☐ Name

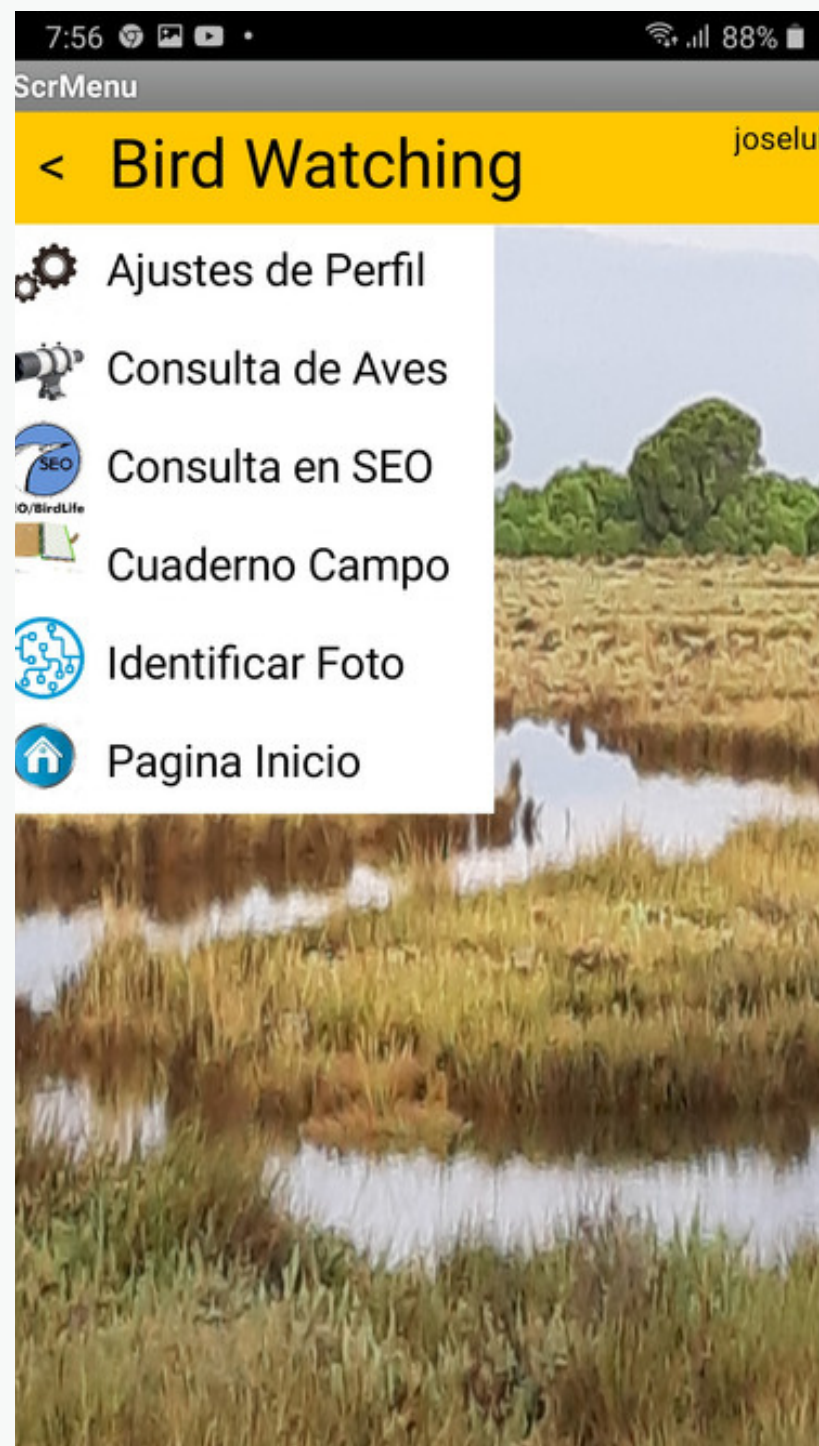
☐ AvesHuelva/

☐ Observaciones/

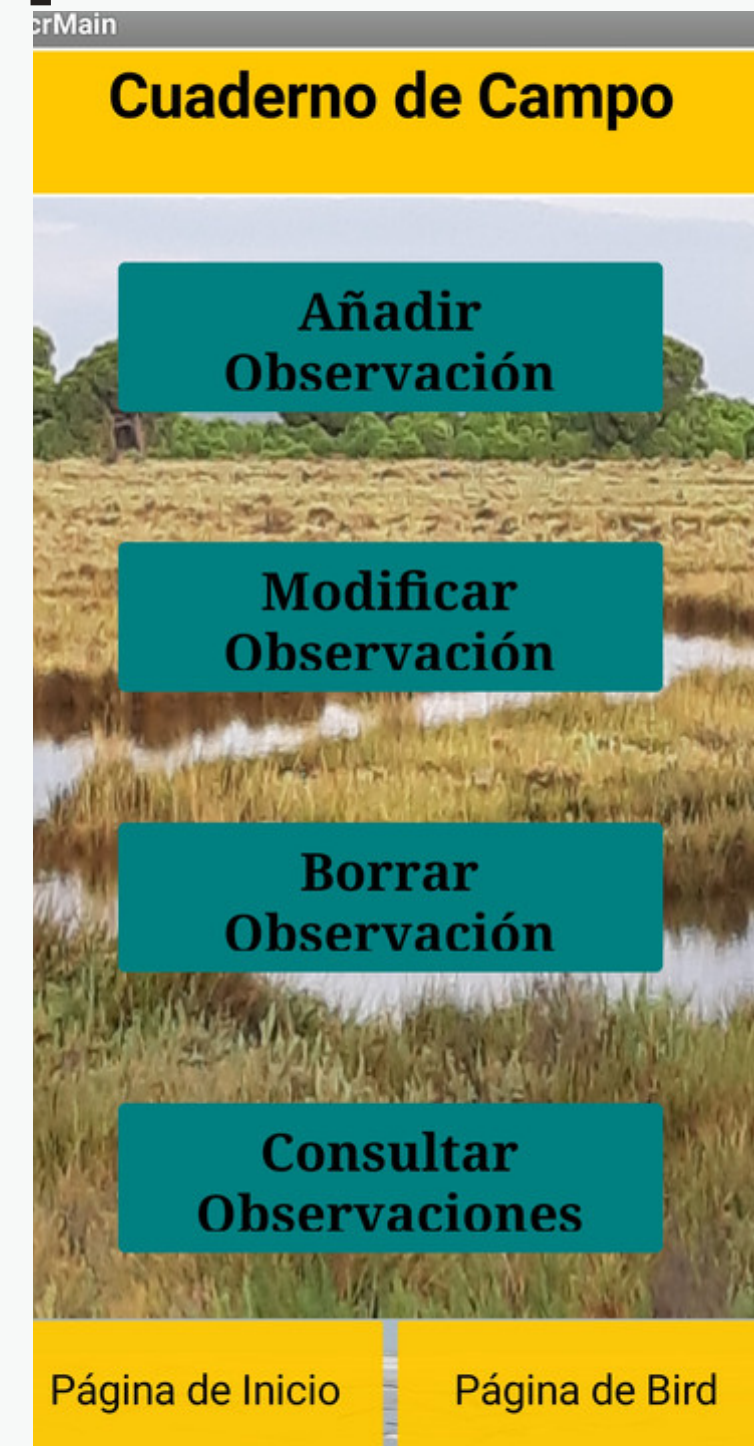
Prototipos e interacción de pantallas



Página Inicial



Menú BirdWatching



Menú Cuaderno de Campo



Crear una Observación



Consulta SEO



Consulta de Aves

IDENTIFICACIÓN DE AVES

Para la identificación de aves se han contemplado el uso de dos herramientas:

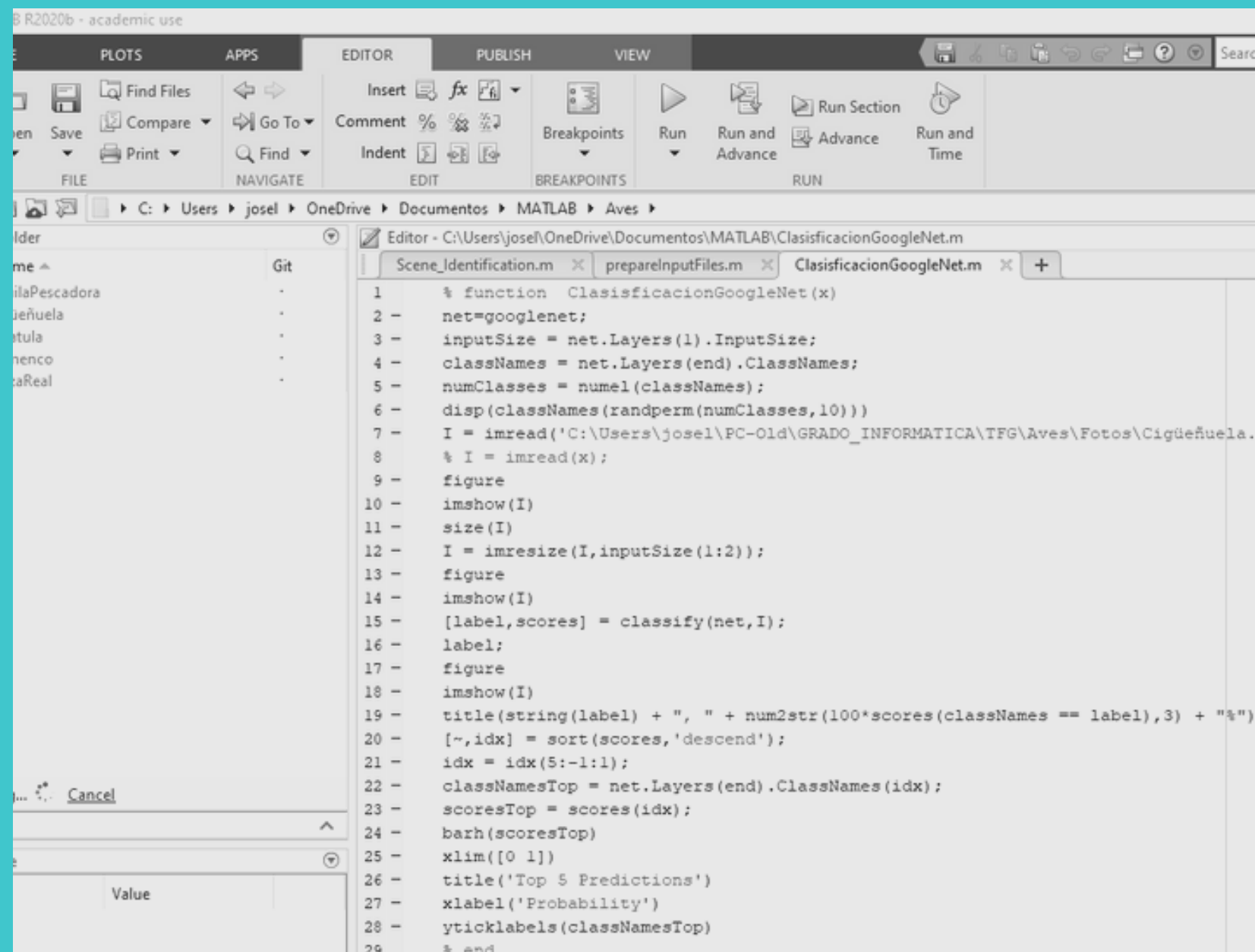
GoogLeNet

Clasificador de imágenes que utiliza una red neuronal convolucional profunda que se ejecutará desde MatLab donde se modificará el tamaño de la imagen para poder realizar la clasificación e identificación.

Personal Image Classifier

Herramienta de Machine Learning, la cual hace uso de Tensorflow.js

GOOGLNET



Cargar la red GoogLeNet previamente entrenada.

Este paso requiere el paquete de soporte del modelo Deep Learning Toolbox para la red GoogLeNet en MatLab.

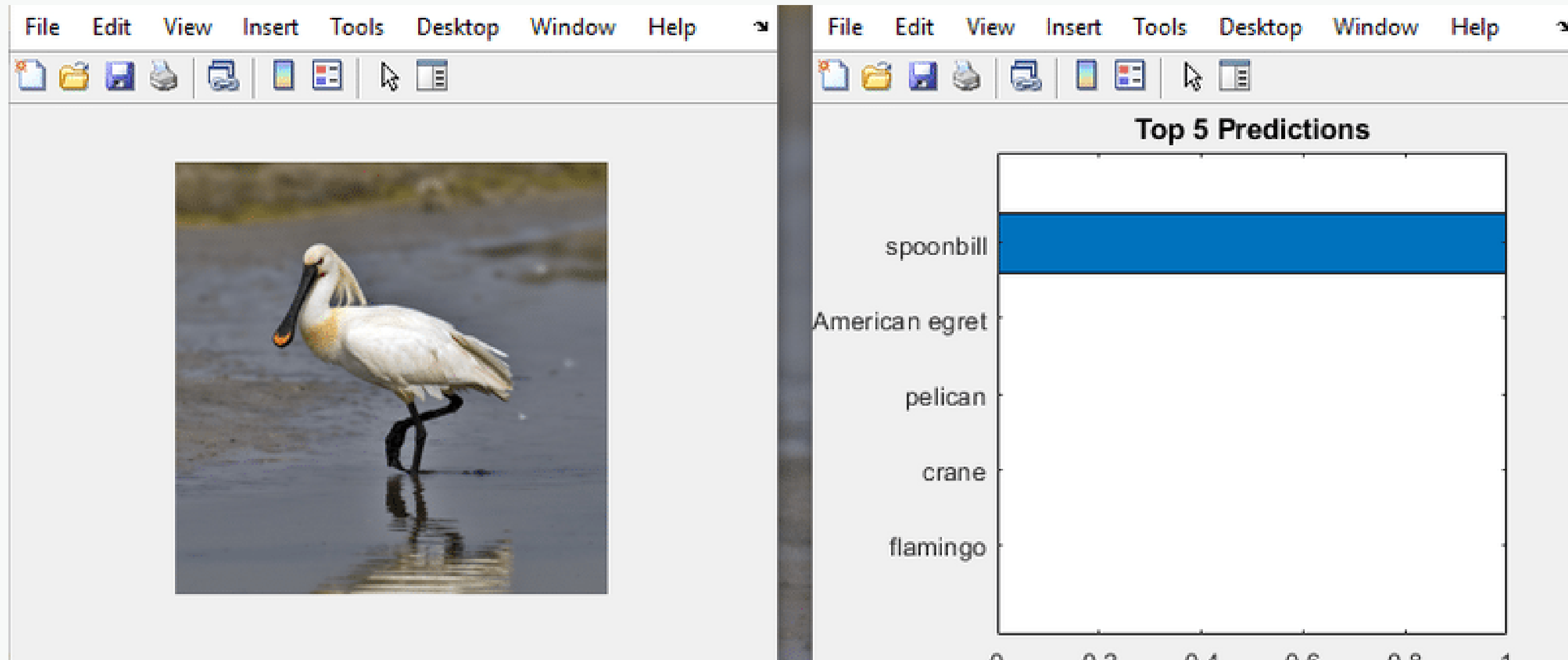
Modificación del tamaño de la imagen.

La imagen que se desea clasificar debe tener el mismo tamaño que el tamaño de entrada en la red.

Capa de entrada de imagen.

El tamaño de entrada de la red es la propiedad InputSize de la imagen. La propiedad ClassNames de esta capa contiene los nombres de las clases aprendidas por la red


RECONOCIMIENTO DE AVE CON GOOGLNET





PERSONAL IMAGE CLASSIFIER


Personal Image Classifier


1 Add Training Data 2 Select Model 3 Add Testing Data 4 View Results

×  Flamenco (20 examples) Flamenco20.jpg

×  Espátula Común (20 examples) espatula20.jpg

×  Aguila Pescadora (20 examples) Aguila_pescadora_19.jpeg

×  Garza Real (20 examples) garzareal20.jpg

×  Cigüeñuela (20 examples) cigu23.jpg

Creación de etiquetas para crear el modelo

El primer paso es la creación de etiquetas a las cuales les añadimos imágenes para crear nuestro modelo.

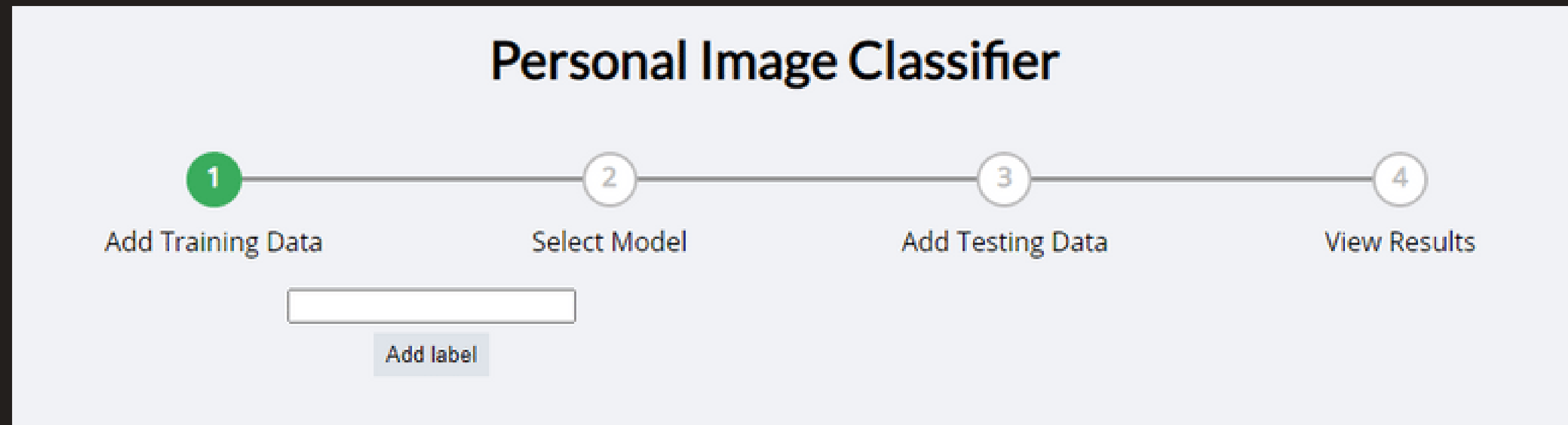
Training

Una vez creadas las etiquetas con sus respectivas imágenes se procede al entrenamiento del modelo.

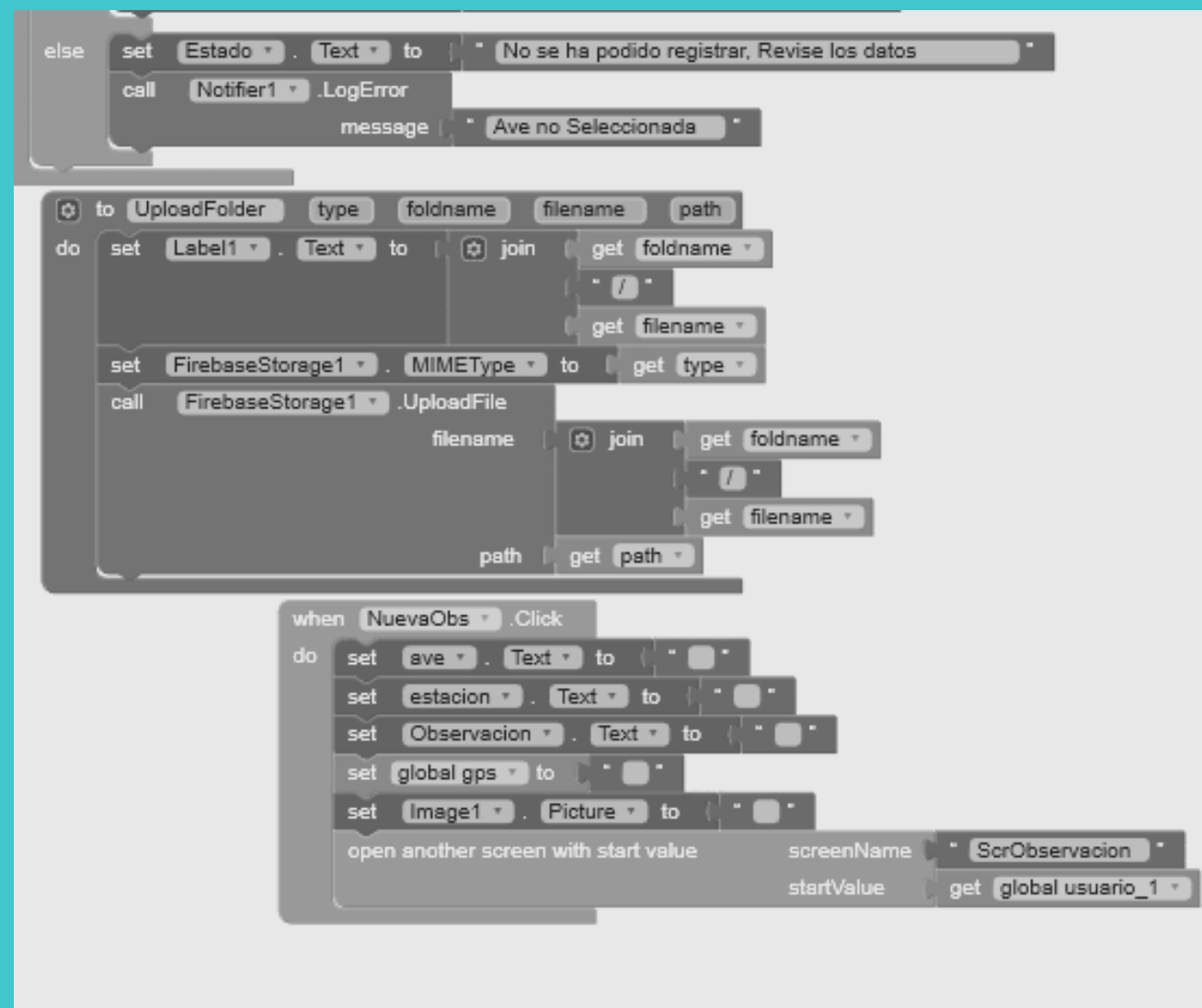
Exportación del modelo

Finalizado el entrenamiento, se crea y se exporta el modelo para poder utilizarlo con una extensión específica que se integra con nuestra app desarrollada con App Inventor.

PERSONAL IMAGE CLASSIFIER



IMPLEMENTACIÓN APP INVENTOR



Diseño de Pantallas

Las pantallas se diseñan en la parte gráfica designer donde tenemos una paleta de herramientas dividido en: user interface, layout, Media, sensores, social, storage, connectivity y extensiones.

Desarrollo con Bloques

Es el lugar donde se desarrolla la lógica de cada pantalla, de forma intuitiva podemos conectar bloques utilizando lenguaje de control..

Extensiones

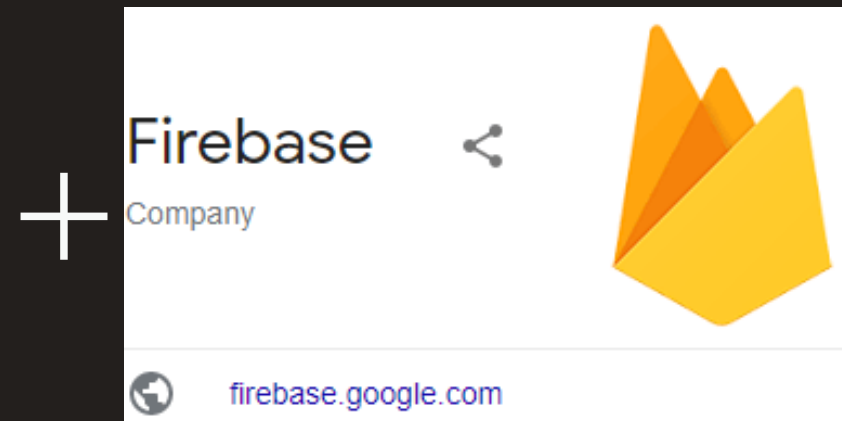
Software que se puede integrar para utilizar en el desarrollo de bloques como por ejemplo la de FireBase o la de Personal image Classifier.

TECNOLOGIA UTILIZADA

App Inventor



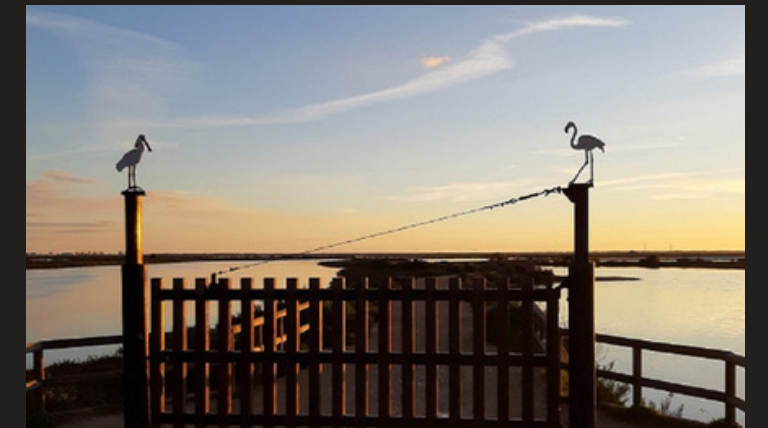
Firebase



MatLab y PCI



App BirdWatching en
las Marismas de
Huelva



PRUEBAS

Se han realizado pruebas de todos los requisitos mencionados en el proyecto junto a sus casos de uso:

- El usuario se puede registrar e iniciar en la aplicación.
- El usuario puede consultar las aves y buscarlas según determinados criterios.
- Consultas de descripción y características del ave.
- Grabar observaciones en el cuaderno de campo.

CONCLUSIONES

- Dificultad al interpretar una base de datos noSql.
- Utilización nuevas tecnologías.
- Nueva fase en la que se incluyan otros idiomas para formar parte de un proyecto con la EOI Huelva referente a ecoTurismo.

BIBLIOGRAFIA

- <https://firebase.google.com/>
- <https://console.firebase.google.com/project/marshbirds-7f263/database/marshbirds-7f263/data>
- <https://www.qrcode-monkey.com/es/>
- <https://online.visualparadigm.com/app/diagrams/#diagram:proj=0&type=UseCaseDiagram&gallery=/repository/b92c57a8-fe64-40ed-8d29-f990c60442ba.xml&name=ATM>
- <https://sourceforge.net/projects/ganttproject/>
- <https://es.mathworks.com/solutions/image-video-processing/object-recognition.html>
- <http://appinventor.mit.edu/explore/resources/ai/image-classification-look-extension>
- <https://www.tensorflow.org/overview>
- <https://classifier.appinventor.mit.edu/oldpic/>