
GitHub Username: DiegoAmezquita

Fundacion Salva

Description

Permite ver informacion relacionada con la fundacion salva, los perros que estan cuidando, asi como informacion acerca de la fundacion, ademas permite ver el detalle de cada uno de los perros asi como varias fotos de cada uno.

Ademas permite llevar un registro de las mascotas asi como recordatorios para no olvidar nada.

Let you see information about the "Fundacion Salva" and the dogs that they are taking care, also you can register your own dog to create reminders and never forget anything about them

Intended User

para personas que aman a los animales y quieren ayudar a aquellos que no cuentan con un hogar o han pasado por situaciones dificiles, ademas de llevar un registro mas organizado de eventos relacionados con sus mascotas

For people who loves dogs and animals in generals and want to help them, also for those who need to create reminders about their dogs.

Features

- Saves information to use the app offline
- Login with Google account
- create profile for the user's dog (private profile)
- reminders for each dog
- widget to keep informed about the "fundacion salva"

User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1

Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]
Provide descriptive text for each screen

Screen 2

Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]

Provide descriptive text for each screen

Add as many screens as you need to portray your app's UI flow.

Key Considerations

How will your app handle data persistence?

Using SQLite and Content providers

Describe any corner cases in the UX.

When there is no internet connection the app shows the latest information and show the user that the information is not updated.

Describe any libraries you'll be using and share your reasoning for including them.

Picasso: handle the loading and caching of images.
OkHttp: handle the connections with the web services and the errors
CardView: display the nice cards from google
Ads: show banner ads and intersitial ads

Next Steps: Required Tasks

Task 1: Project Setup

- Import libraries
- Enable all the configuration on the IDE (Android Studio) like Instant Run and Proguard with the basic rules
- Create different flavors (Free and Paid)

Task 2: Implement UI for Each Activity and Fragment

- Build UI for LoginActivity
- Build UI for SignUp Activity
- Build UI for MainActivity (View Pager)
- Build UI for Feed Fragment

- Build UI for Information Fragment (Information about the “Fundacion Salva”)
- Build UI for Feed Detail fragment
- Build UI for Add pet Fragment
- Build UI for User’s pets Fragment
- Build UI for Pet detail fragment

Task 3: Create a basic flow

- Populate the app with dummy information
- Make the animations for the transitions (Shared elements)
- Create a complete flow for a new user (register, organization feed, add pet, add reminder)

Task 4: Create the Basic communication with the server

- Create the ServerManager class that handle all the request to the server
- Handle most of the possible cases (No Internet, timeout, bad response)

Task 5: Create the Content Provider manager

- Create the Sqlitehelper
- Create the tables
- Create the methods to populate, get, delete and update the information

Task 6:Integrate the real information with the user interface

- Make the real calls to the server
- store the information locally (Sqlite)
- notify the UI that the information has changed

Task 6: Optimizations and improvements

- Make a good profile to improve the performance
- Check that proguard is working properly for the release build