

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any corner cases in the UX.](#)

[Describe any libraries you'll be using and share your reasoning for including them.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

GitHub Username: jmoix

GetPix

Description

Have you ever been in the middle of messaging someone and wanted to send a picture as the perfect reply, but searching and downloading an image just seemed too tedious. GetPix solves this problem by providing an easy to use interface for searching images and attaching them to messages so that you can finally express yourself the way you want, without the tedious task of searching on the web. GetPix allows you to save images to your favorites for later use, as well!

Intended User

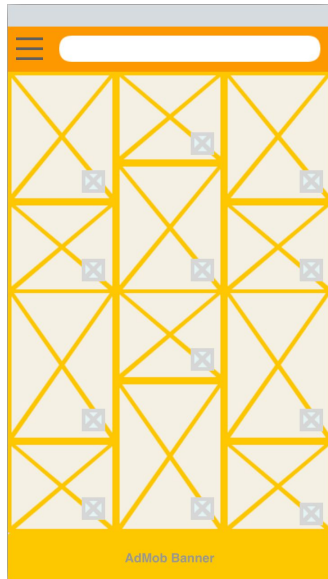
The intended user is anyone who likes to communicate through instant messages.

Features

- Easily search for images using the Google search API.
- Attach images to instant messages, directly from the app.

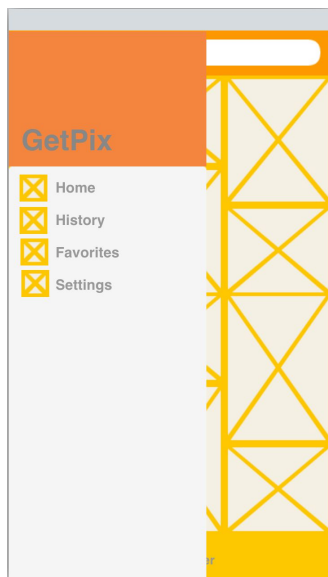
- Image caching to save on data charges.
- Image favoriting for later quick access.
- View history of images that have been sent in the past.

User Interface Mocks



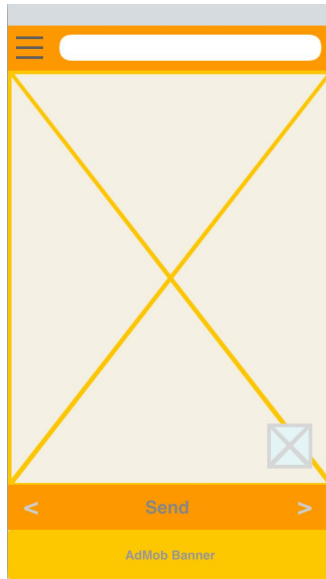
Screen 1 - Home

The home screen provides the search interface from which you can choose an image to send or save to your device for later use. Like all other screens, you are provided access to the navigation drawer in order to allow easy access to other screens. A Favorite indicator is provided with the image search results, allowing the user to favorite/unfavorite an image. An AdMob banner is placed at the bottom of the screen.



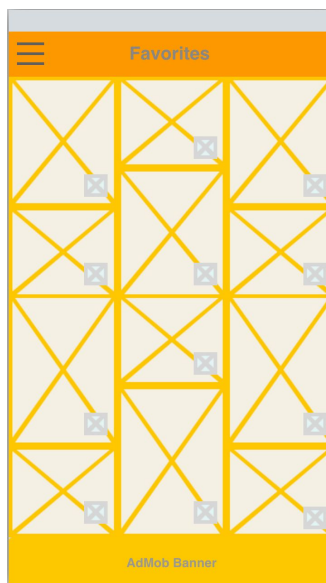
Screen 2 - Navigation Drawer

Clicking the hamburger icon on any screen will provide access to the navigation drawer.



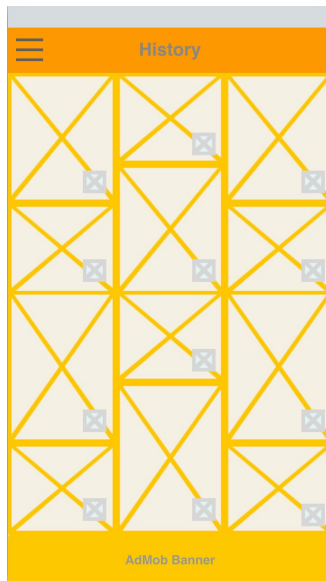
Screen 3 - Image Screen

Clicking on an image in search results, history, or favorites will navigate the user to the image screen. Here they can toggle the favorite option or send the image via a share intent. An AdMob banner is placed at the bottom of the screen.



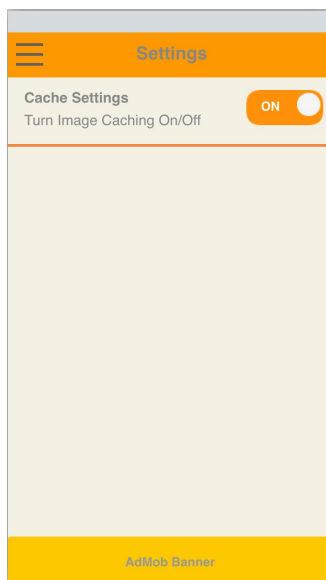
Screen 4 - Favorites

The Favorites screen lists all the images that the user has favorited. An AdMob banner is placed at the bottom of the screen.



Screen 5 - History

The history screen lists images that have been shared. An AdMob banner is placed at the bottom of the screen.



Screen 6 - Settings Menu

The Settings menu allows the user to toggle application options, such as image caching. An AdMob banner is placed at the bottom of the screen.

Key Considerations

How will your app handle data persistence?

The app will handle persistence of the images through the use of a content provider that will be developed specifically for the app.

Describe any corner cases in the UX.

All screens will be handled by the fragment manager and the manipulation of its backstack. The user can return to any screen by using the phone's back button or by using the navigation drawer.

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

- Android Design Support Library for For UI Design and Material Components.
- Android Support Library for Other needed support components.
- Butterknife for Variable binding.
- Retrofit for Performing Http requests.
- RecyclerView for Image GridView recycler.
- Google Play Services for AdMob.
- Gson for json parsing.
- Espresso for instrumentation tests.

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Project Setup

- Create project in Android Studio.
- Add needed libraries to Gradle.

Task 2: Implement initial Home Activity UI

- Implement Material color scheme in colors resources.
- Add Home Activity and Home Fragment.
- Implement the Toolbar into the UI.
- Add the Navigation Drawer.
- Add AdMob Banner at the bottom of the screen.

Task 3: Networking

- Setup request and response objects.
- Implement networking calls to make search using retrofit.
- Implement response parsing using Gson.

Task 4: Data Management

- Implement content provider to store references to favorites, current/last search results, and cached images. This includes creating tables, the content contract, and database helper.

Task 5: Fragments

- Implement the search result, favorites, history, image view, and settings fragments.
- Images should be loaded into the fragments using data stored in the content provider and loaded with a cursorloader.
- Wire in fragment loading to the navigation drawer.
- Implement fragment management using the fragment manager.

Task 6: Data Sharing

- Setup app to share image data with other apps, such as the messaging app.