Capstone Project: Warsart

GitHub Username: meekmika

Table of Contents

Description

Explore street art in Warsaw! The app locates graffiti and murals all over the city. View pictures, read about the pieces, and find out how to get there. Save your favorites and create a list of places to visit. Discover the beuatiful city of Warsaw and its rich urban art scene at the same time!

Intended User

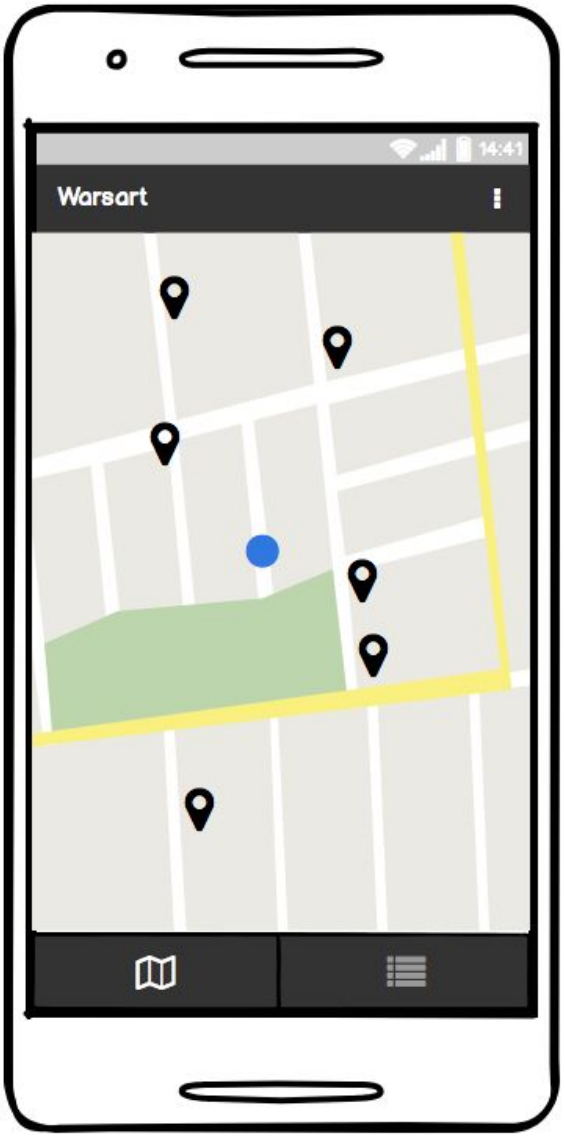
- Tourists
- Travelers
- Art enthusiasts

Features

- Locates street art on a map
- Lists street art in a list view
- Allows user to mark a piece of street art as favorite
- Allows user to filter map and list view to only show favorites
- Provides a detail view for every piece of street art with images, descriptions and address

User Interface Mocks

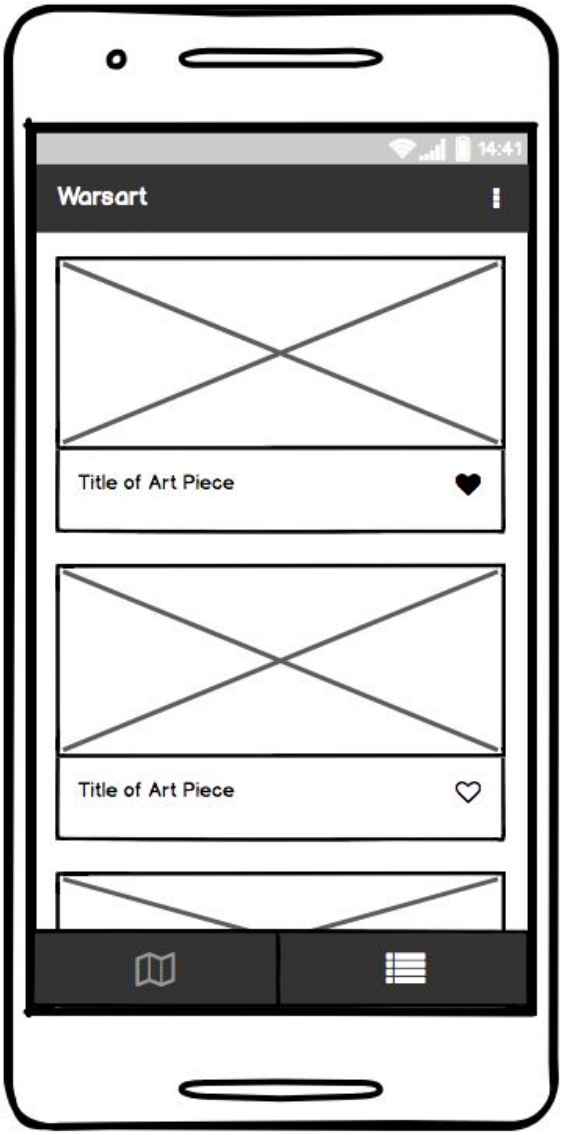
Map View



MainActivity with MapFragment

Each street art piece is marked on a map. The position of the device is also marked. Clicking on a marker opens up a Detail view with more information about the selected piece. Opening up the overflow menu lets the user toggle between only showing favorites or showing all of the pieces.

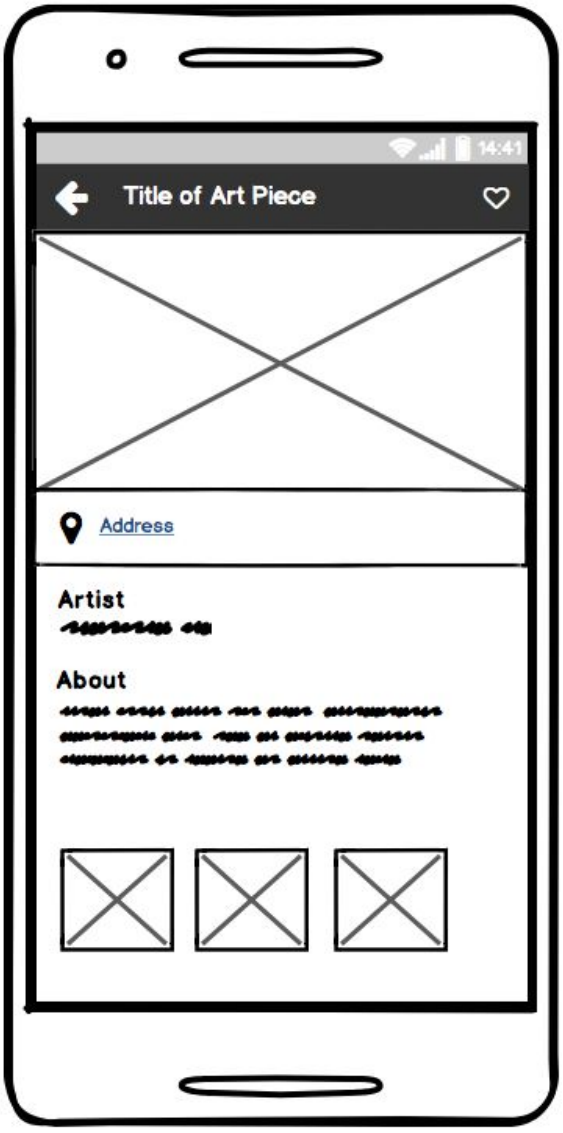
List View



MainActivity with ListFragment

A list with a card view for each street art piece. The user can mark a piece as favorite from this screen. Clicking a card opens up a Detail view. Opening up the overflow menu lets the user toggle between only showing favorites or showing all of the pieces.

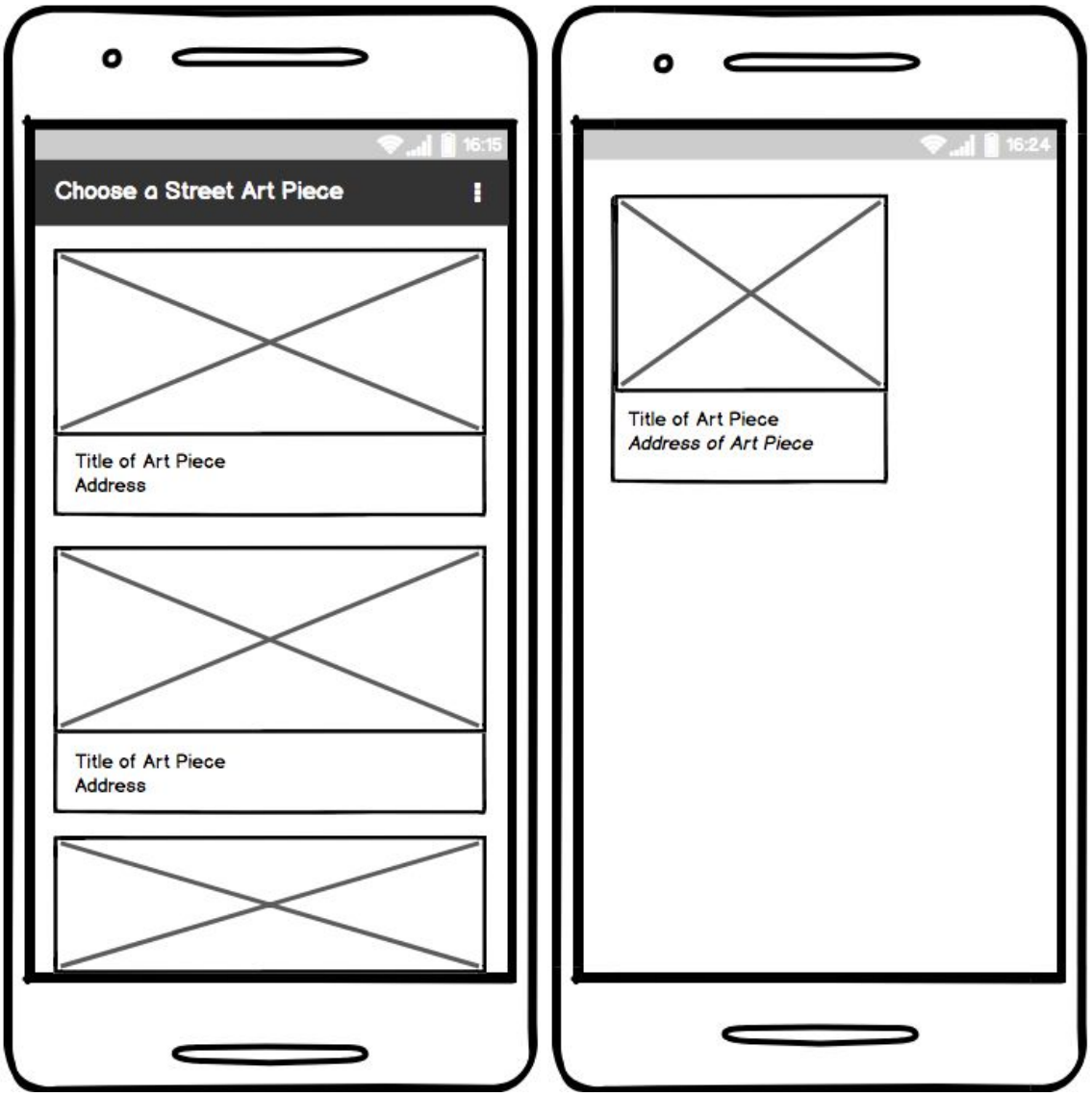
Detail View



DetailActivity

Details about a selected street art piece. The user can favorite the piece by clicking the icon in the app bar. Clicking the address opens up the location in a maps application, e.g. Google Maps. Clickling the up arrow closes the DetailActivity and shows the previous activity with correct fragment.

Widget Configuration Screen and Home Screen Widget



When placing a new app widget on the home screen the Configuration Activity opens and the user has to choose which piece to display in the widget. Clicking the home screen widget opens up the Detail view.

Key Considerations

How will your app handle data persistence?

Firebase Realtime Database will be used for storing information about all of the street art pieces and Cloud Storage for Firebase will be used for storing all of the images.
Room will be used to store a local, offline database of the user's favorite street art pieces.

(I have chosen to produce my own images and data to add to the Cloud Storage and Firebase Realtime Database manually. Party because I want to showcase my findings during my stay there and am a bit selective about what pieces I want to display in the app, but mainly because I haven't found any API out there to be satisfactory.)

Describe any edge or corner cases in the UX.

In the Map view, a map of the device's current location should be shown. Ideally the user should be in Warsaw to see nearby streetart on the map. If the location of the device is far away, e.g in Sweden, the user shouldn't be shown a map of the current location, i.e Sweden, but a map of Warsaw instead.

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

Picasso - for the caching and loading of images.
Timber - for better logging.

Describe how you will implement Google Play Services or other external services.

Maps - Display a map with markers for all of the street art pieces, as well as the current location.
Firebase Realtime Database - Store JSON data about the street art pieces.
Cloud Storage for Firebase - Store images of the street art pieces.

Next Steps: Required Tasks

Task 1: Project Setup

- Create new Android project
- Add dependencies
    - Picasso
    - Timber
    - Maps
    - Firebase
- Create API keys

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
    - Build UI for MapFragment
    - Build UI for ListFragment
- Build UI for DetailActivity

Task 3: Implement Maps and Firebase

- Setup Firebase Realtime Database
- Setup Cloud Storage for Firebase
- Populate database and storage with data

Task 4: Populate the Views with Data

- Put a marker on the map for each of the street art pieces in the MapFragment
- Populate the RecyclerView in the ListFragment
- Make the markers and listitems clickable and open DetailActivity with relevant data when clicked

Task 5: Mark as Favorite

- Allow the user to click a heart and mark the street art piece as favorite
- Allow the user to only show favorite street art pieces on the map or in the list with SharedPreferences

Task 6: Create Widget

- Build UI for Configuration Activity

- Build UI for App Widget
- Make widget display relevant data
- Make widget clickable and open DetailActivity with correct data when clicked

Task 7: Handle Error Cases

- Make sure all error cases are handled and that app does not crash
    - Display placeholder when image failed to load
    - Display appropriate error messages, e.g when the user has no internet connection

Task 8: Visual Pizzazz and Accessability

- Make sure the app is user friendly and looks neat
- Add app icon
- Add accessibility support, e.g content descriptions and RTL layout switching

Task 9: Building

- Create a release build and a keystore that is referred to by a relative path