GitHub Username: ojiofong

City Explorer

Description

City Explorer helps you find places of interest around a city. It allows one to explore immediate places around and also be able to get further information such as contact details, opening hours, reviews etc. City Explorer will also provide Map and Street views of places around.

Intended User

The target user is a tourist or anyone that is new to a city or region.

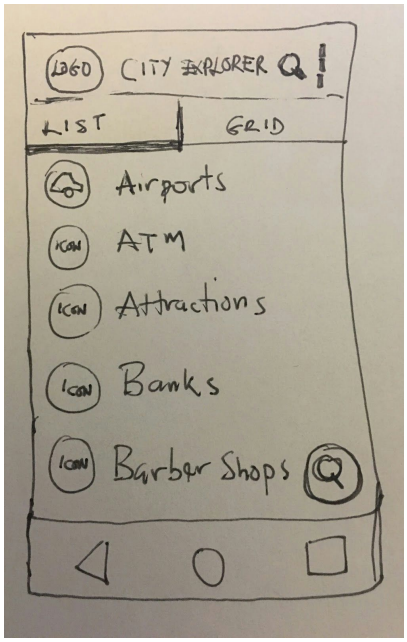
Features

- Provides list of popular places to search
- Allows search by custom key words
- Provides quick links to get directions
- Shows map
- Shows street view
- Saves custom search history for convenience
- Allows a user to change location to explore other regions

User Interface Mocks

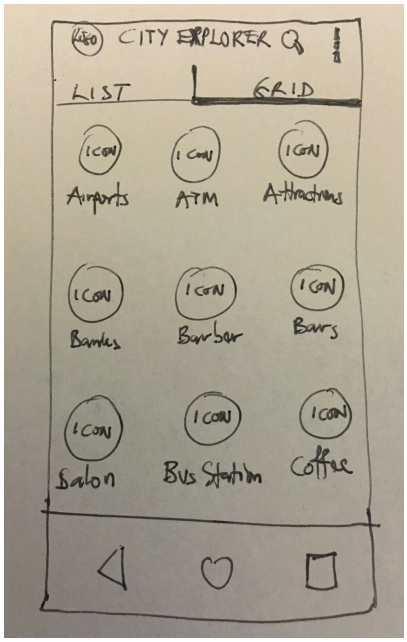
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1 (MainActivity - List)



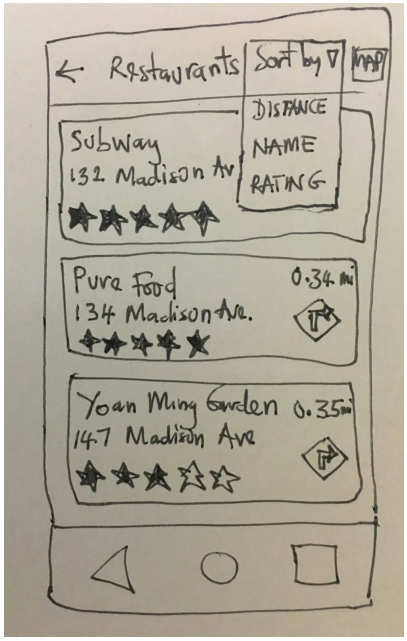
The List fragment is selected by default in the MainActivity. Selecting a list item will show a list of cards of the places around. The FAB allows a user search for places by inputting a custom keyword not shown on the list.

Screen 2 (MainActivity - Grid)



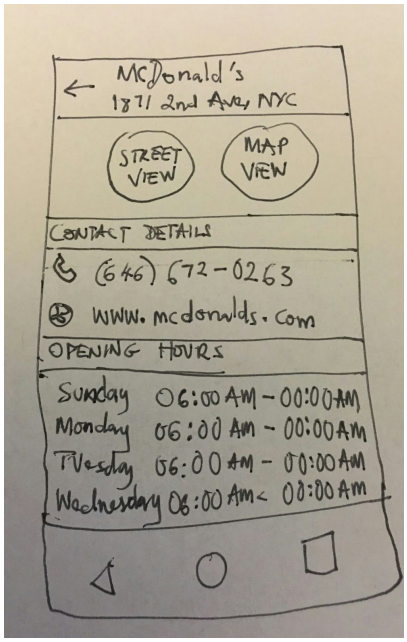
The Grid fragment is part of the MainActivity. It shows items in a Grid View. Selecting a list item will show a list of cards of the places around. The FAB allows a user search for places by inputting a custom keyword not shown on the list.

Screen 3 (Place Results Activity)



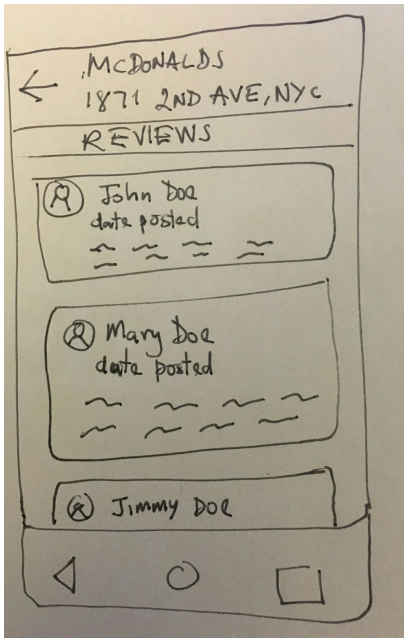
The Place Results Activity displays the list of place results. The list can be sorted based on distance, name or rating. Clicking on each card will open the Place Details view.

Screen 4 (Place Details Activity)



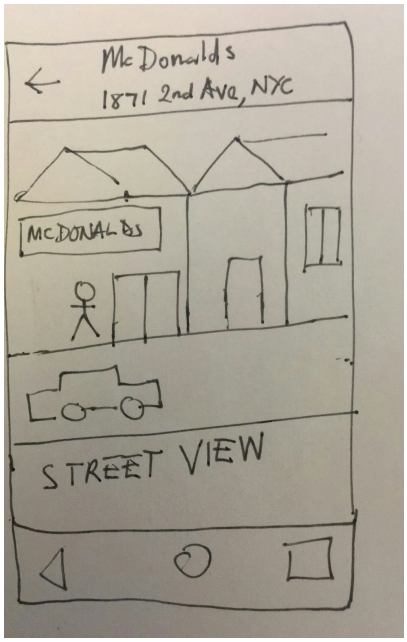
The Place Details Activity displays a detail view of the place selected. It will contain contact details, link to reviews, street view, map view etc.

Screen 5 (Reviews Activity)



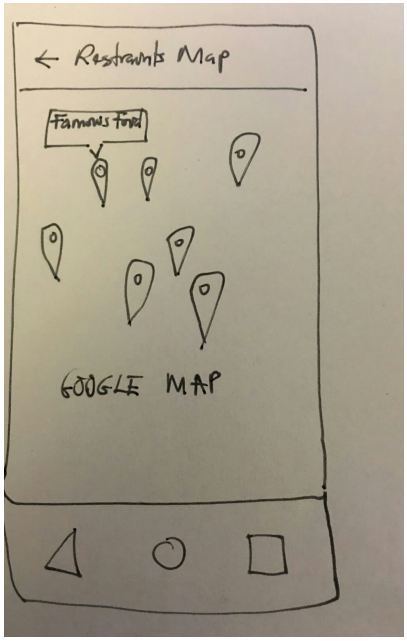
Reviews Activity displays the list of reviews for a place. Each card in the list contains the author's name, comments and date of posting.

Screen 6 (StreetView Activity)



The StreetView Activity will show a street view of the place selected. A user can interact with the screen and explore the street or area.

Screen 7 (Map Activity)



The Map Activity will show the list of place results on a Google map. Selecting a marker will display a snippet showing the name and address of the place represented by the marker.

Key Considerations

APIs

Google Places API - To get list of places around a user.
Google Maps V2 API - Required to implement Google maps on android client.
Google Street View API - Required to show street view of places.

Data persistence

Data (search history & settings) will be stored locally on the device via SharedPreferences.
Google Play Services is required for Google Maps etc.

Corner cases in the UX.

The MainActivity will have a TabLayout with 2 tabs linked to a ViewPager. The first tab1 or page1 will contain a list view of popular places while tab2 or page2 will contain a grid view. On first run the default tab selected is List. The app will remember a user's last tab selection and will set that as a the default.

Libraries to use

Google Play Services - To access Google Services and APIs.
Support Design Library - To access Material design widgets e.g. FAB etc.
CardView - Extends FrameLayout to show information in cards.
RecyclerView - Provides a more advanced and flexible version of ListView.
Gson - For data serialization and deserialization.

Next Steps: Required Tasks

Task 1: Project Setup

- Create Android Project on Android Studio
- Set up Git repo on GitHub
- Configure libraries needed
- Create Google Project and enable required APIs

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity (Includes TabLayout & ViewPager)
- Build UI for Fragment for popular places List View
- Build UI for Fragment for popular places Grid View

- Build UI for Place Search Results (shows list of cards)
- Build UI for Google Maps
- Build UI for Street View
- Build UI for Settings
- Build UI for Place Detail
- Build UI for Place Reviews (shows list of cards)
- Build UI for Changing location

Task 3: implement Google Project APIs and Google Play Services

Implement Google Play Services in order to access Google Services and APIs.

- Create Google project
- Activate Google Places API
- Activate Google Maps API
- Activate Google Street View API
- Include specific Google Play Services aar library in app module's build.gradle file
- Check for Google Play Services update or availability during usage
- Include Meta-data Specifying Google Play Services version in AndroidManifest.xml

Task 4: Create Layouts and get assets

The Layouts will provide a visual perception of how the UI will look like. The layouts should be able to adapt to different screen sizes and orientation.

- Create List layouts
- Create List item layouts
- Create multi-pane UI layouts in Tablets.
- Implement RecyclerView to hold list of places and results.
- Get needed android icons, images etc. needed

Task 5: Data Storage

Data will be stored stored locally via SharedPreferences:

- Create Java Bean class to hold search history data
- Implement serialization using Gson
- Save serialized json string in SharedPreferences
- Run test CRUD operations from Android client

Task 6: Others

- Create Adapters required
- Connect Adapters with RecyclerView
- Populate with RecyclerView with Data
- Connect UI components with real data from the cloud