GitHub Username: Popnfresh234

I Love Youbike

Description

I Love Youbike is a simple and straightforward way for you to make use of Taipei's Youbike system. Quickly and easily locate Youbike stations and see how many bikes are available. I Love Youbike has a full English version as well as a full Chinese version. Data is retrieved from the Taipei City Open Data API and stored in a local database.

Intended User

I Love Youbike is intended for use specifically by travellers and residents in Taipei who are not fluent in written Chinese. Currently there is no application for accessing the Youbike system in English available.

Features
- Fetches Youbike station data from Taipei City Government Open Data API and syncs to a local database
- Provides a listing of all Youbike stations and how many bicycles are available at each station
- Uses user's location to determine nearest Youbike location
- Display a map showing the user's location and nearby Youbike stations

User Interface Mocks

Screen 1 – Main Screen
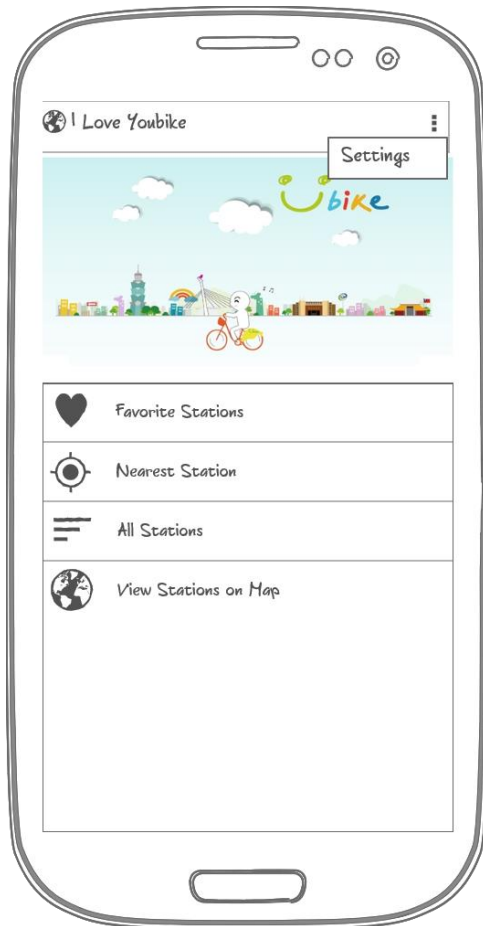
Provides list with items for favorite stations, nearest station by location, all stations, and view stations on map.

Settings menu is available in the overflow menu to change languages. This option is necessary as the data from the API can be returned in one of two languages, English or Chinese.

Screen 2 – List of Stations / Favorite Stations
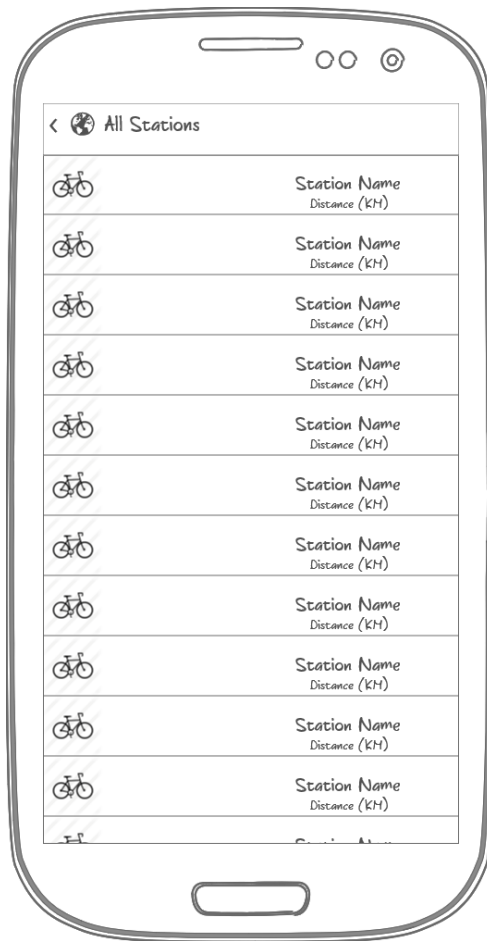
This screen provides a list of all stations in order of distance from user's current location
Clicking on a list item will take the user to a station details screen
Icon will show a different color for status of station

- Green for bikes available
- Red for no bikes available
- Yellow for no spaces available to return bikes

This layout can be reused for the favorite stations screen

Screen 3 – Map Screen

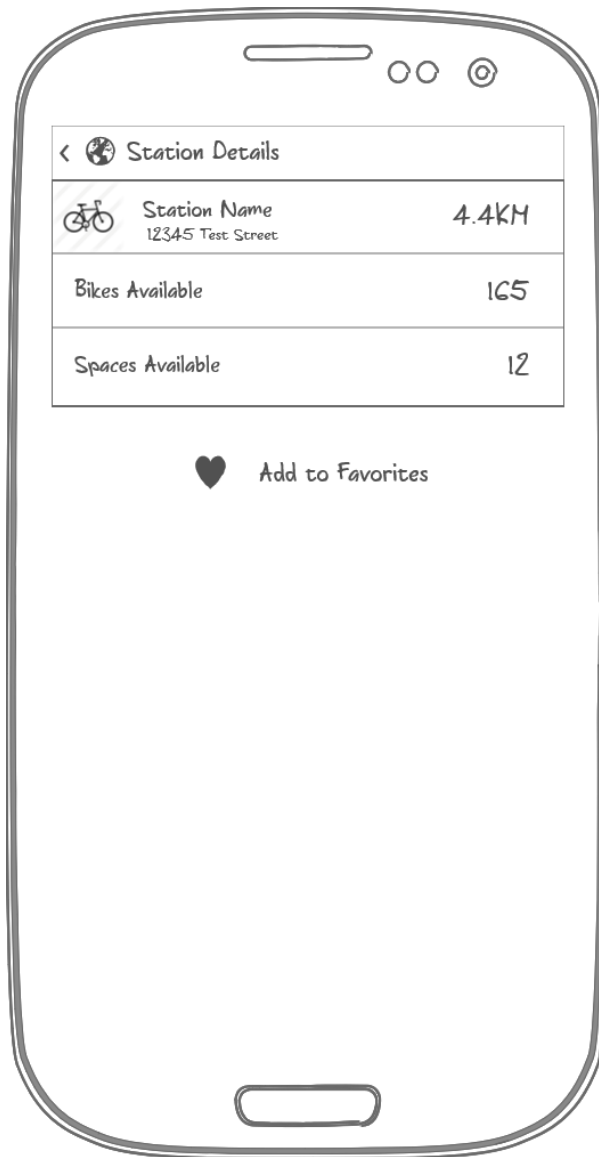This screen shows a map with the user's current location and Youbike stations
Icon will show a different color for status of station
- Green for bikes available
- Red for no bikes available
- Yellow for no spaces available to return bikes

Screen 4 – Station Details Screen

This screen contains details of the selected station
- Stations name, address, and distance from user
- Number of bikes available for rent
- Number of spaces available to return bikes
- Button to press to add station to list of favorites

Key Considerations

How will your app handle data persistence?

App data will be fetched from API and stored in an SQLite database. Sync will be handled by a sync adapter and content provider. An on demand refresh option will also be available as up to the moment data is needed.

Describe any corner cases in the UX.
Rotation while viewing a station's detail will cause the scroll position of the list of stations to be lost. Scroll position should be stored and restored on recreation of the list.

Attempting to use the app without google play services installed will limit features.

If the user has disabled location services or location cannot be determined distance to stations can't be determined. An empty view will be necessary in this case.

If user attempts to use the app for the first time without a network connection no data will be displayed, an empty view is necessary.

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

1. Appcompat library to ensure compatibility with as many versions of android as possible
2. Support design library will be used for backwards compatible material design features
3. Butterknife for reducing view boilerplate code
4. Google play services for maps and location
5. RecyclerView for handling the list of stations

Tasks

Task 1: Project Setup

Create a new Android project

Create four activities
- MainActivity
- StationListActivity
- StationDetailActivity
- StationMapActivity

Include needed libraries
- compile 'com.android.support:appcompat-v7:22.2.0'
- compile 'com.android.support:design:22.2.0'
- compile 'com.android.support:recyclerview-v7:22.2.0'
- compile 'com.google.android.gms:play-services:7.5.0'
- compile 'com.jakewharton:butterknife:6.1.0'

Task 2: Implement UI for Each Activity and Fragment

Build UI for Main Activity

- Build UI for All Station List Activity/Favorite Stations List activity
- Build UI for Station Detail activity
- Build UI for View Stations on Map Activity

Task 3: Implement Content Provider and Sync Adapter

Implement a database to store data retrieved from web API
- Create a database helper as well as a database contract to ensure data is entered into database correctly
- Create URIs for querying the databse. There will be two types of queries, a query for all stations and a query for specific stations based on a station ID number.

Implement a content provider and sync adapter to keep the database up to date
- Content provider need only be a stub since this app doesn't share data with other apps
- Implement sync adapter to fetch data from web API and store in the database via content provider

Task 4: Create All Stations activity

This activity will use a loader to query the database created by the sync adapter to return a list of all available stations. This activity can be reused as a favorite stations activity since the loader just needs to use a different query. The query parameter can be passed in as an extra with the intent.

- Implement RecyclerView to display data from the Loader

- Implement a RecyclerAdapter to fill the RecyclerView with data
- RecyclerView should implement an OnClickListener that calls through to the activity, passing back the clicked upon station's ID
- Activity should then launch a Station Details activity with the selected station's ID

Task 5: Create Station Details Activity

This activity will display the details of a station selected from the All Stations List or Favorite Stations List

- Implement a loader to query the database based on station ID passed as an extra with the intent that launched the activity
- Display key data from query in TextViews

Task 6: Create Settings Activity

The data from the web API contains values for each station's details in Chinese and in English. As the data is returned from the web API, it is not a string resource and thus not translatable and unaffected by the device's language settings. This settings activity will allow the user to choose which language results from the query should display in. This is a useful feature as users may wish to display the station data in one language or the other, i.e. to show to a taxi driver or to a friend to ask for directions.

- Implement a settings activity to set the desired language
- When displaying data from a query, check the preference and fetch the correct string from the cursor returned by the query.

Task 7: Create View Stations on Map Activity

The web API has data for each station's coordinates. This activity will use that data to plot stations on a map along with the user's current location

- Implement Google Maps API to add a map to the activity
- Implement Google Play Services in order to make use of device's location