Tokyo 2020 App

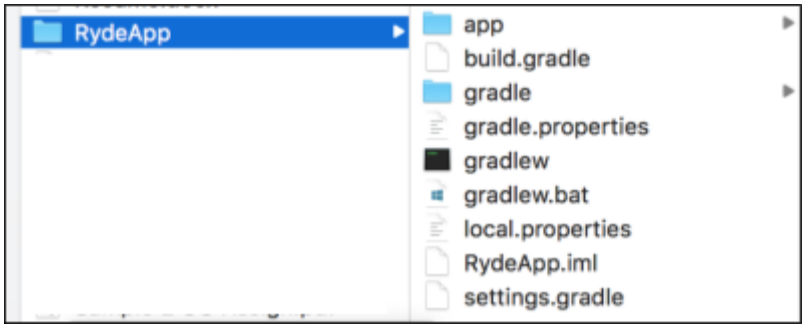
(35% of overall grade)

Due Date

Code submission: March 2, 2019 - 11:59PM to Blackboard Dropbox

Submission Instructions

1. This is a group assignment
2. Maximum 2 students per group.
3. Include student name and id at the top of all **README.md** file
4. Create a zip file containing your ENTIRE PROJECT. This is what the ENTIRE PROJECT means:



5. Upload zip file to Blackboard dropbox
6. One submission per group.

Logic Implementation Guidelines

- Where required, use object oriented programming principles (example: Movie class)
- Data between screens must be passed using intents.
- Where appropriate, data must be persisted with Room and/or SharedPreferences

User Interface Requirements

- Where appropriate, provide navigation drawers and toolbars.

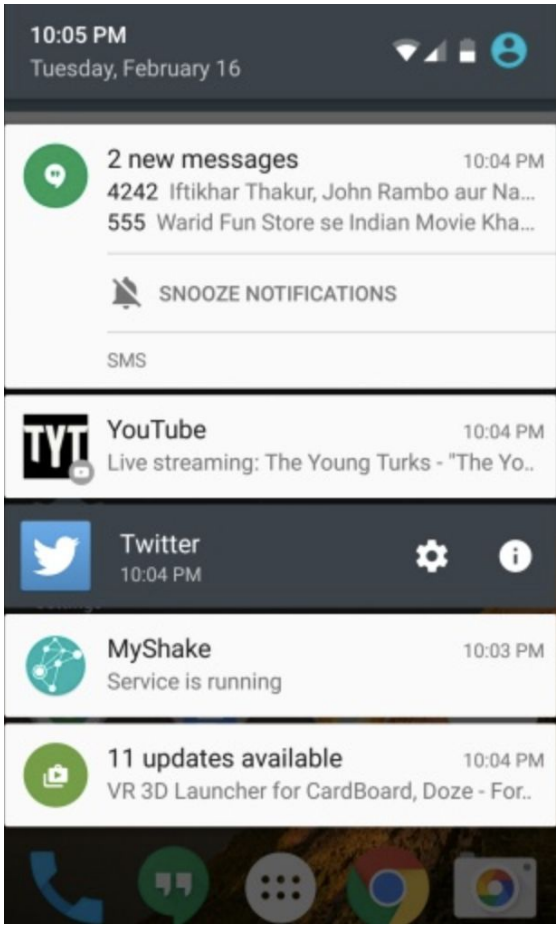
App Description

In July 2020, the Japanese city of Tokyo will be hosting the 2020 Summer Olympics. In preparation for the Olympics, the Tokyo Board of Tourism has hired you to build an Android app that helps tourists discover Tokyo's tourist attractions.

The app should have the following features:

- User account login and signup
    - Provide screens to allow user to signup for an account and/or login to an existing account
    - The users account should remember:
        - Username
        - Password
        - Which tourist attractions the user has on their "wishlist"
- Tourist Attractions:
    - A list of popular tourist attractions in Tokyo. This list must show the attraction name, address, photo, and a 1 line description of the attraction.
    - When user selects an attraction from the list, display a screen that shows more detailed information about the attraction. (this could include a video, longer description, or pricing)
    - Tourist attractions must be loaded from a database on the phone.
    - **Wishlist** - The app should allow user to add a tourist attraction to their "wishlist" of things to see.
    - **Ratings** - User should be able to leave a star rating for the tourist attraction. HINT: Use a "Rating Bar" view

- Olympic Schedule: A schedule of Olympic events
    - **Overall schedule:** Using a web view, display the overall schedule for the olympics: https://tokyo2020.org/en/games/schedule/olympic/
    - **Schedule by day**: User enters a date (using DatePicker). App displays the schedule of events for that day. The schedule is shown in a webview.
        - You will need to find a way to link the date entered with the events on that day.
        - The Olympics website shows a list of games by day. For example, here is a list of all events on July 25, 2020: https://tokyo2020.org/en/games/schedule/olympic/20200725.html
        - If the user enters a date is outside the Olympic Games time period (or that does not have any events), then your app should display an error - or display: “No events on that day”
- Olympic Game Reminders
    - User can signup for reminders of when a specific game or event is occurring. The app will display a notification before the event occurs.



- Contact Us
    - A page that allows user to contact the Tourism Board, either by email, phone or SMS.

- Admin Interface
    - Your app should provide an admin-only interface
    - This allows administrators to add new locations to the “list of tourist attractions”
    - After the admin adds a new location, all registered users should be able to see the update the next time the user logs in.

Database Modelling

For students who are less comfortable with databases, you can model the user and their wishlist like this:

USERS table:

username	password	type
thanos@gmail.com	1234	admin
wonderwoman@yahoo.com	abc00021	user
jonsnow@winteriscoming.com	gameofthrones2	user
superman@kypton.com	kk11iii	user

ATTRACTIONS table:

attraction_name	address
CN Tower	1 Main Street
Niagara Falls	55 Niagara Way
Eaton Center	255 Queen Street
Casa Loma	8 Kendall Avenue

WISHLIST table:

username	attraction
jonsnow@winteriscoming.com	CN Tower
jonsnow@winteriscoming.com	Niagara Falls
wonderwoman@yahoo.com	CN Tower
superman@krypton.com	Eaton Center
superman@krypton.com	CN Tower

When writing your DAO for the Wishlist, you can get a user’s wishlist like this:

```
@Dao
public interface WishlistDao {
    @Query("SELECT * FROM wishlist WHERE username = :name")
    public List<WishlistItem> getWishlistByUsername(String name);
}
```

DISCLAIMER: The query above may have errors, see full documentation here:
https://developer.android.com/training/data-storage/room/accessing-data#query-params

For students who feel more comfortable with databases, you can try modelling your database with JOINS:

USERS table: (primary key = id)

id	username	password	type
1	thanos@gmail.com	1234	admin
2	wonderwoman@yahoo.com	abc00021	user
3	jonsnow@winteriscoming.com	gameofthrones2	user
4	superman@kypton.com	kk11iii	user

ATTRACTIONS table:

attraction_name	address
CN Tower	1 Main Street
Niagara Falls	55 Niagara Way
Eaton Center	255 Queen Street
Casa Loma	8 Kendall Avenue

WISHLIST table: (using user_id as foreign key to the USERS table)

user_id	attraction
3	CN Tower
3	Niagara Falls
2	CN Tower
4	Eaton Center
4	CN Tower

Then you can query the table like this:

```
@Dao
public interface WishlistDAO {
    @Query("SELECT * FROM wishlist " +
           "INNER JOIN users ON users.id = wishlist.user_id " +
           "WHERE wishlist.user_id = :id")
   public List<WishlistItem> getWishlistByUser(int id);
}
```

DISCLAIMER: The query below may have errors, see full documentation here:
https://developer.android.com/training/data-storage/room/accessing-data#query-multiple-tables