Android Developer Nanodegree: Final Project Proposal

GitHub Username: EricRybarczyk

Application Name: **Road Trippy**

Description

This app provides a clear and fun way to plan a driving trip, and helps you stay organized while you travel. Multiple days of itinerary can be saved for each trip, making it quick and simple to initiate navigation when it is time to drive. Smooth integration with Google Maps provides searching for destinations, creating and saving an overall route, adding planned stops along the way, and initiating Navigation for each leg of your journey. Additional notes can be saved and organized with each trip.

Intended User

Those traveling by car or other personal transportation via roadways can benefit from this app. The app is particularly helpful to those who like to plan a trip in advance and organize their travel details in a unified system.

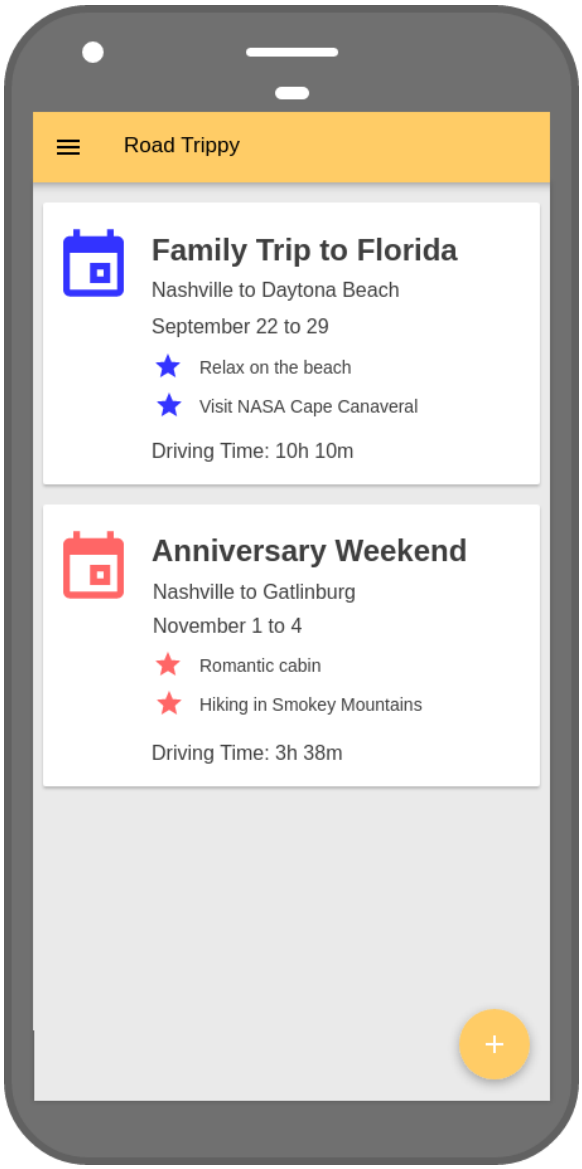
Features

- Search for locations and view results on a map within the app.
- View driving directions overview on a map within the app.
- Save details to a Trip Plan (origin, destination, waypoints, dates).
- Easily start Navigation in Google Maps from the Trip Plan within this app.
- Save additional text with the Trip Plan, entered in the app or shared to the app from other apps.
- View information about possible trip destinations (limited demo content at this time)
- Access details from past trips saved in the app.

User Interface Mocks

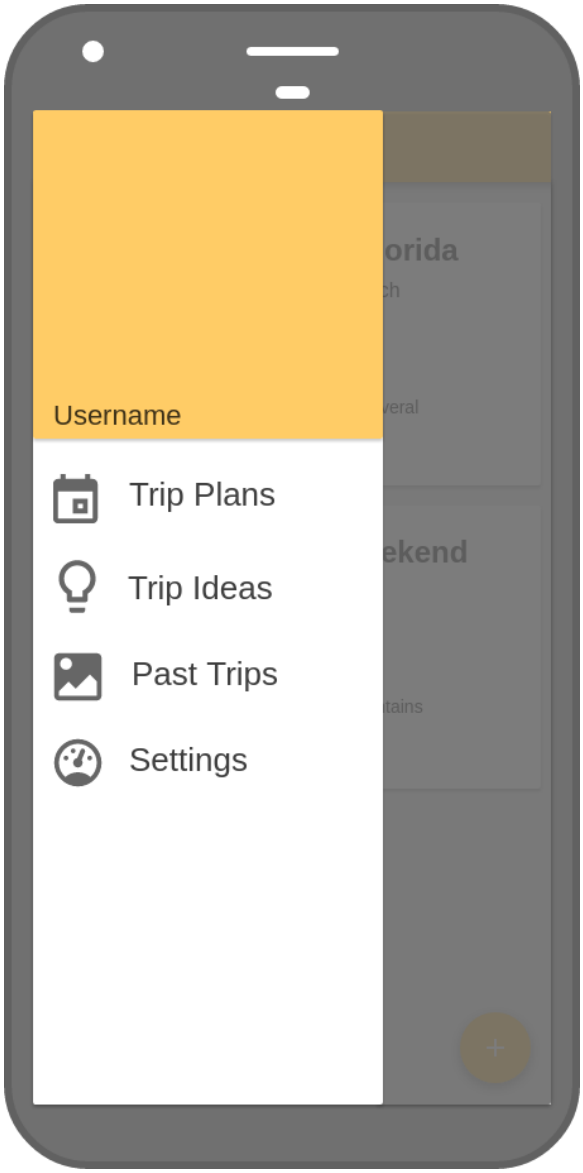
Main Screen

Showing two example trips created.



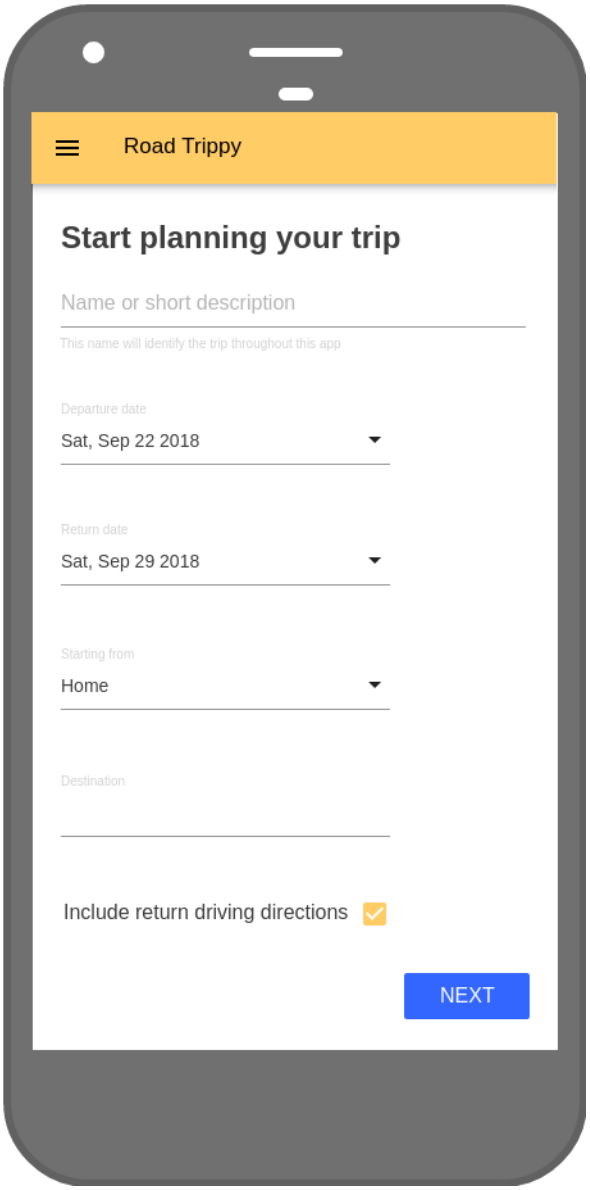
Navigation Drawer

Access to the main features of the app



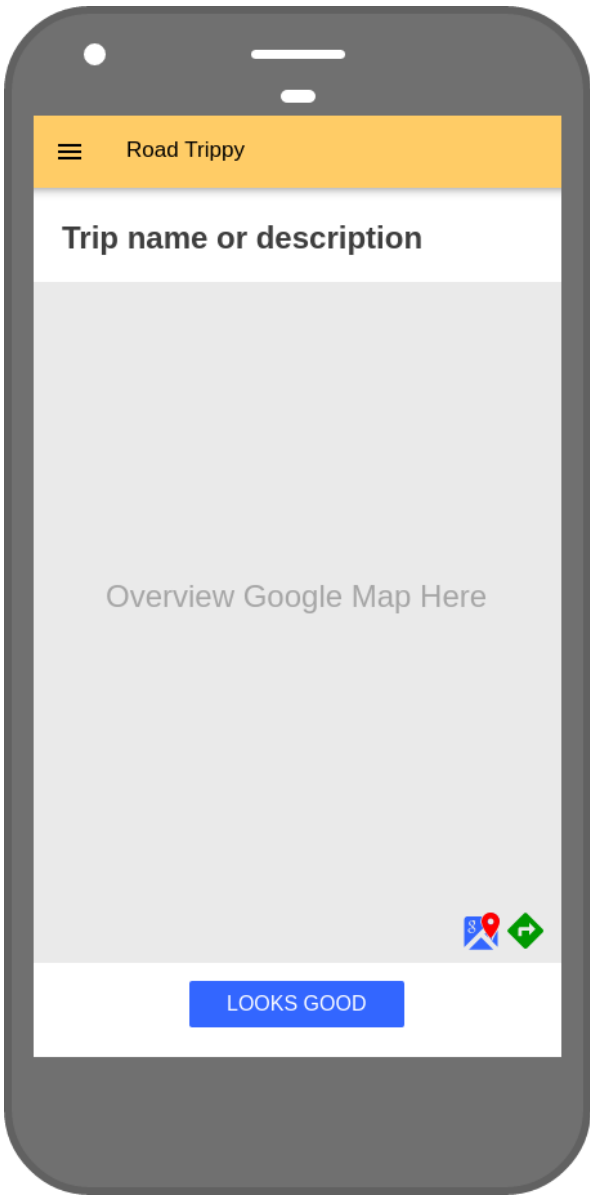
Create a Trip

Initial entry for a new trip. App will calculate days from date range provided. “Starting from” and “Destination” will allow searching for locations. User settings will also populate “Starting from” with known saved locations like “home” for the user.



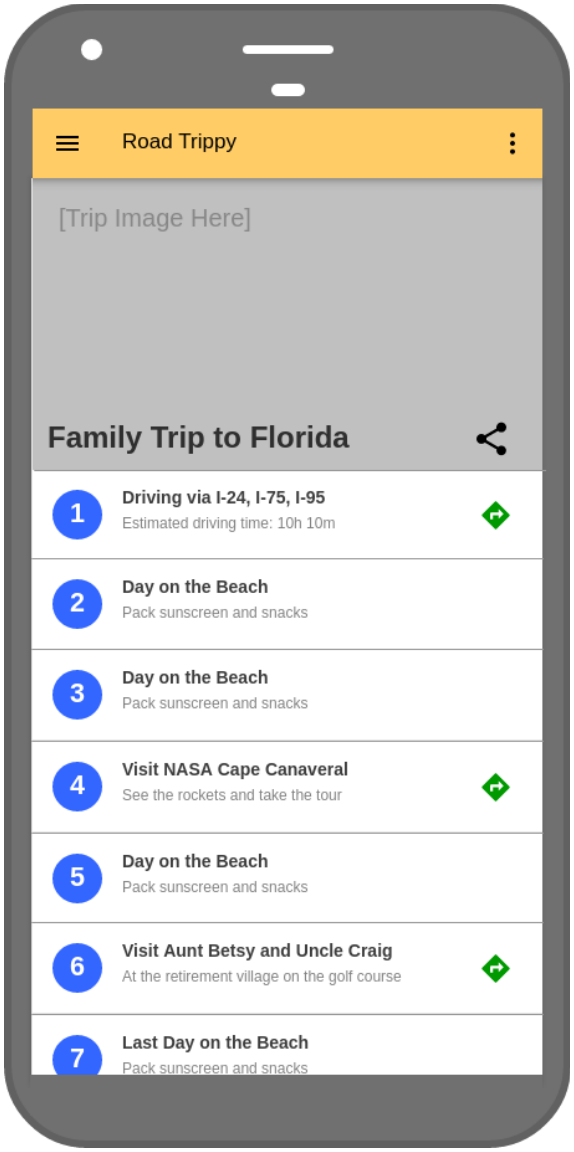
Trip Overview Map

Google Map showing route overview for the trip, based on origin and destination parameters provided on the prior screen. Icons to launch Google Maps via appropriate Intents are also provided.



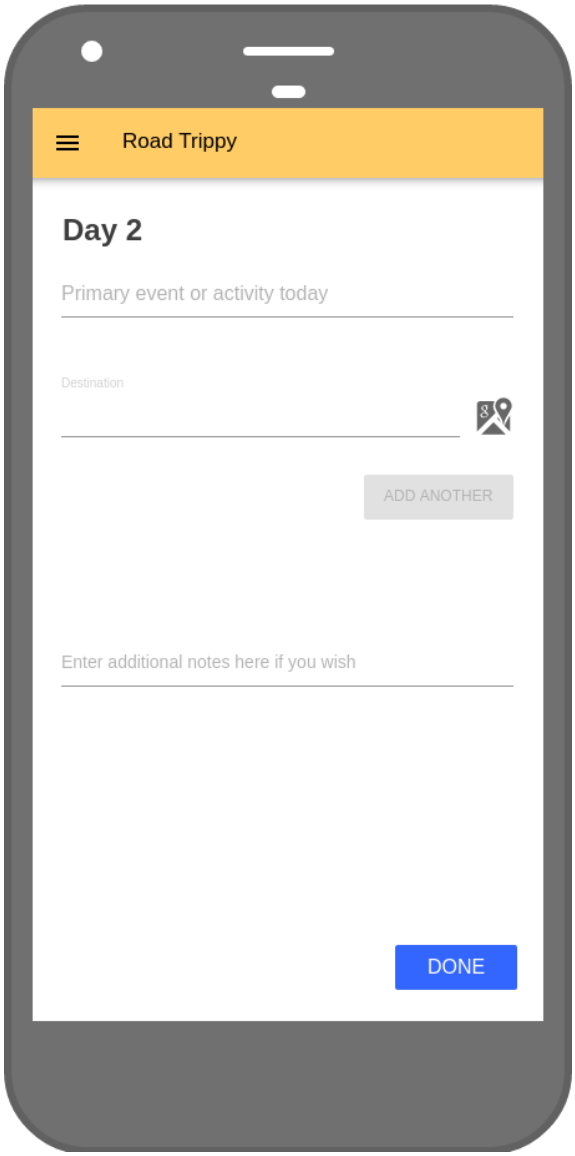
Trip Days List

List of calculated days for the trip, generated from date range provided. Text shown below represents items that have been edited (next screen) by the user. Initial screen would show default text prompting the user to enter details for each day. User may leave any days as default text with no loss of primary functions. Tapping the green Navigation icon will launch the Trip Overview Map.



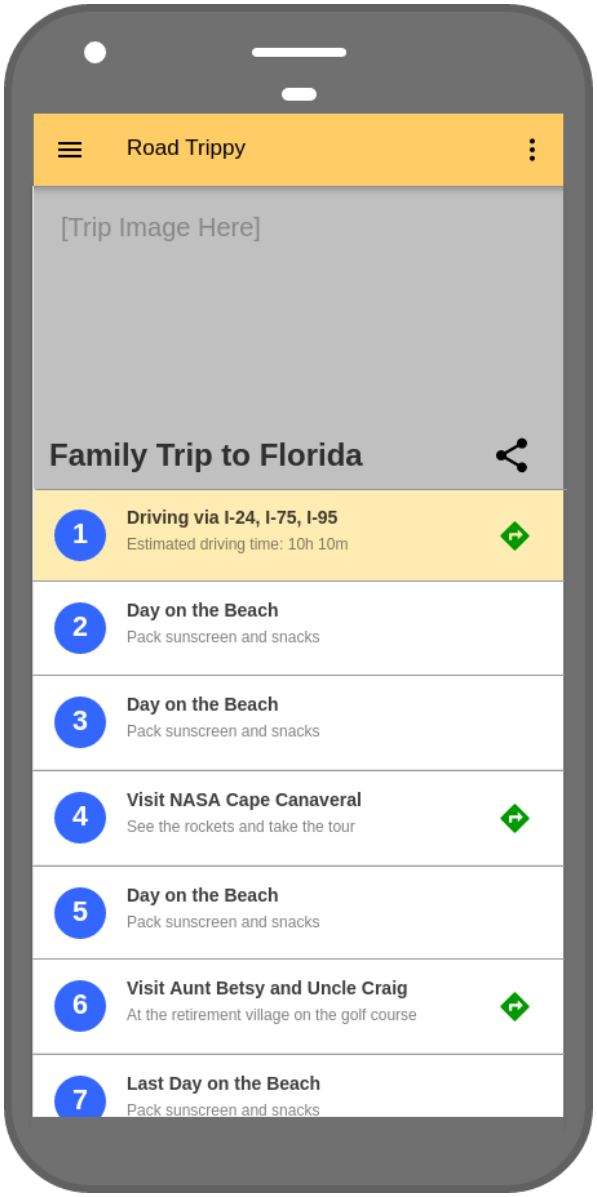
Trip Day View / Edit

Screen to view or modify details for the selected (from prior screen) day. User may optionally add one or more Destinations as part of this day. The Google Maps icon will allow them to search & select using Google Maps SDK integration in this app, resulting in the green Navigation icon & integration as seen on the prior screen (Days 1, 4, 6 in this example)



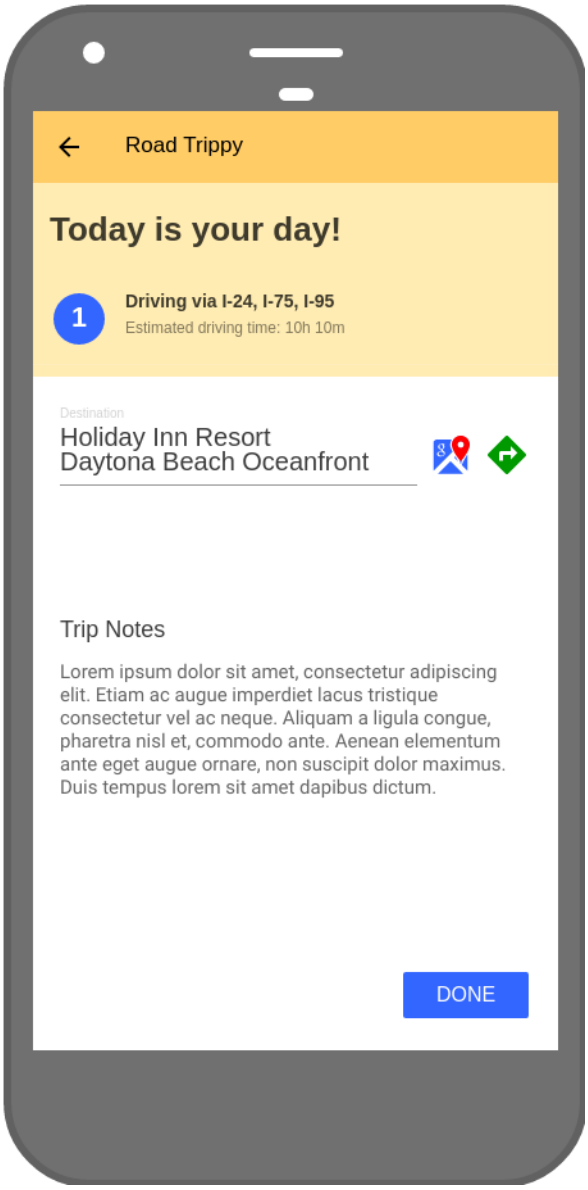
Trip Days List - Active Date Highlight

List of calculated days for the trip, showing the current date highlighted. Based on current date, active trip day is highlighted. Functionality is identical to this screen on any prior date, as shown previously.



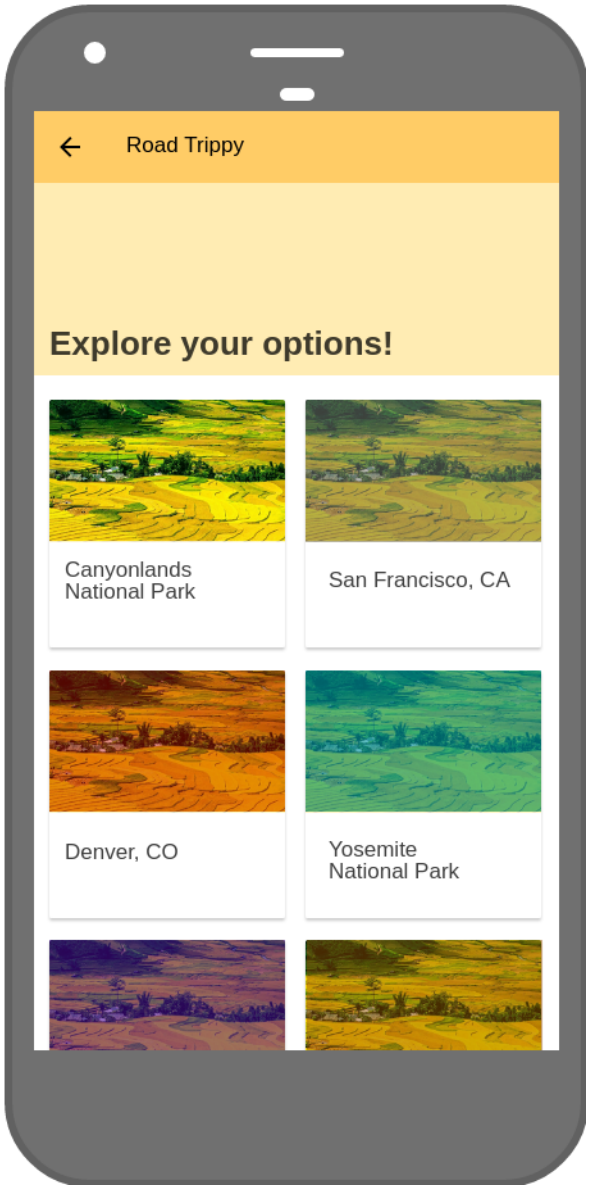
Trip Day - Active for Current Date

Day details for active trip day. Note, overflow menu in top navigation bar will provide option to modify, but normal scenario will focus on providing easy access to planned details. Icons for Google Maps and Navigation will launch Google Maps via appropriate Intents.



Trip Ideas Screen

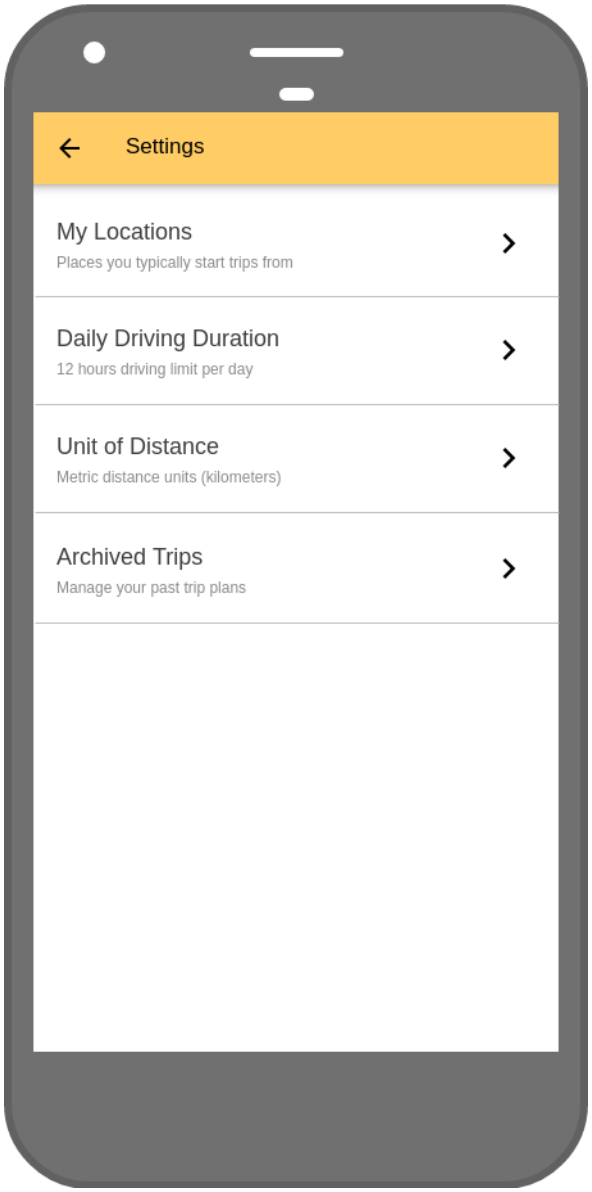
Provides suggested trip destinations. This feature will contain limited content for demonstration purposes. A future enhancement could further develop this feature, possibly in partnership with another app or website that provides this type of content.



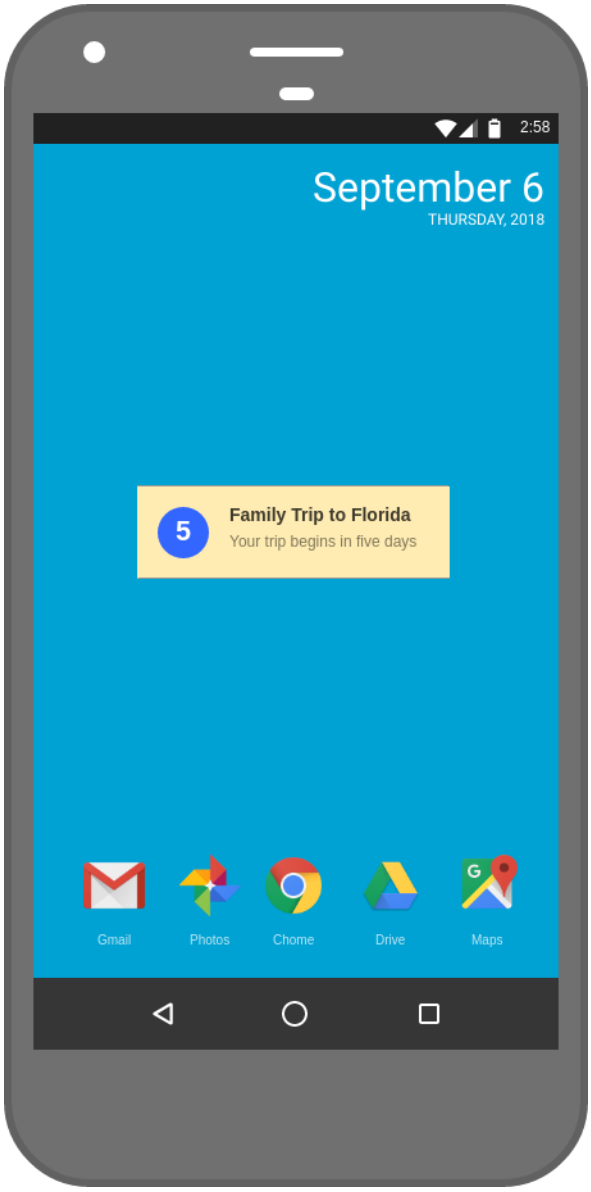
Trip Idea Detail Screen

Displays details about the destination selected on the prior screen, and allows the user to select this as the destination to begin a new trip plan on the Create a Trip screen.

Settings Screen



Widget



Key Considerations

How will your app handle data persistence?

User data in the app will be persisted to Firebase Realtime Database. This data storage option has been selected to allow the user to access the app on multiple devices with automatic data synchronization. This choice will also allow a future version of the app to share trip data in real time among approved family or friends who also use this app.

Describe any edge or corner cases in the UX.

- Trips that involve more than one consecutive day (number of hours TBD) of driving (start and/or return) need to prompt user to indicate how to handle it - such as select a waypoint, or drive straight through. This will determine how the app builds the list of days from the main input parameters.

- Each day within a trip can have one or more optional destinations. The user may return to that day's origin the same day, or may stay overnight at the intermediate destination, etc. Making this clear and easy for the user in the app is important.

- If the user modifies content on screen "Trip Day View/Edit" and navigates back, the app needs to confirm to save or discard changes.

- Some nuance details regarding user experience and app workflow need to be clarified as the Road Trippy app is built. In particular, consistent experience with the Google Maps icons to launch Maps and Navigation via appropriate Intents. The Road Trippy app must provide these icons to access the functionality provided by Google Maps (this app is not replacing Google Maps functionality, it is intended to integrate effectively) but only do so in a way that is clear and consistent to the user, and without detracting from the user experience within the Road Trippy app.

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

1. Butterknife: to simplify code for view binding
2. Retrofit & Gson: For calling remote API's over HTTP
3. Picasso: to integrate images in the app with efficient coding

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

1. Google Maps SDK: provides Map display & interactivity within the app
2. Google Maps API: provides API interface to search for Places and Directions data
3. Firebase Realtime Database: for storage of user data generated in the app
4. Firebase Authentication: authenticate users to control access to their data in Firebase and to personalize the app

Rubric-Specific Considerations

- App will be written solely in the Java language, along with XML & image resources.
- App will utilize only stable release versions of all libraries, Gradle, and Android Studio.
- App will meet Udacity Android Developer Nanodegree - Core App Quality Guidelines as defined at http://udacity.github.io/android-nanodegree-guidelines/core.html

Next Steps: Required Tasks

Task 1: Project Setup

- Create initial Android Project using Android Studio. minSdkLevel level will be 19 (Android 4.4 KitKat), compileSdkVersion and targetSdkVersion will both be 27 (Android 8.1 Oreo.)
- Initialize local Git repository, create master and develop branches, and link to GitHub remote.
- Add to .gitignore file using template from https://github.com/github/gitignore/blob/master/Android.gitignore
- Add library dependencies to the app build.gradle file.

Task 2: Implement UI for Each Activity and Fragment

- Build UI for Main Activity
- Build UI for Navigation Drawer
- Build UI for Create a Trip activity
- Build UI for Trip Overview Map activity
- Build UI for Trip Days List activity
- Build UI for Trip Day View / Edit activity
- Build UI for Trip Day Active Current Date activity
- Build UI for Trip Ideas List activity
- Build UI for Trip Idea Detail activity
- Build UI for Settings activity
- Apply appropriate navigation within activities via AppBar options
- Configure Styles, Color Palette , and Dimension resources
- Create any needed Drawable resources such as icons, including Launcher icon

Task 3: Develop data model

- Entities
- Attributes / Elements
- Cardinality
- Association to User
- Database Rules to secure access to user data

Task 4: Develop Java model classes

- Firebase data model classes
- Google Places API model classes
- Google Directions API model classes
- Trip Ideas data model classes

Task 5: Implement Permissions Verification

- App must check for appropriate Location permissions to support Google Maps API calls
- App must prompt user to grant Location permissions if not already granted

Task 6: Implement AsyncTasks

- Google Maps Places API calls
- Google Maps Directions API calls

Task 7: Implement FirebaseUI Authentication

- User sign-in
- User sign-out
- User profile data (if available & applicable to this app)

Task 8: Integrate Firebase Realtime Database

- Implement Database Rules to secure access to user data
- Save user generated Trip data
- Retrieve the Trip data for display
- Retrieve the Trip Ideas data for display
- Update the Trip data as the trip happens (date driven)
- Update the trip to Archived state after Trip completion
- Implement data removal associated with Archived Trips (User Setting)

Task 9: Implement UI Workflow for Google Maps Places search

- “Create a Trip” screen allows Places search for trip Origin and Destination
- Each “Trip Day” can have user add one or more Destinations via Places search
- Smooth user experience is required to display search results and allow the user to select from the results, or initiate a revised search
- User will be shown a Google Map within the app to visually confirm the location, and will tap a button to confirm. This results in the app storing the key details for the selected Place with the Trip data.

Task 10: Implement component to generate trip images

- App will utilize a feature of Google Maps SDK or API to save a map image related to the trip
- Image will be saved with the user generated trip data
- Image will be displayed in CollapsingToolbarLayout where appropriate, such as the “Trip Days List” activity (see mockups).
- Image may also be displayed on Main Screen as Card item background with a scrim filter. This design option will be evaluated when the app is built.

Task 11: Implement Element Transition Animations

- Effective transition animations will be implemented

Task 12: Implement Widget

- Home screen widget will display basic information from the next trip scheduled
- Number of days remaining until departure will be highlighted
- Tapping the widget will launch the app to the “Trip Days List” screen for the trip

Task 13: Create demo content for Trip Ideas feature

- App will contain 6 to 10 sample data items for the Trip Ideas feature
- Content will be obtained from public domain sources, such as US National Park Service
- Content will be persisted to Firebase, and secured so any user can view this data

Task 14: Implement Sharing Functionality

- Accept text content shared from other apps into this Road Trippy app
- Inbound shared text will allow the user to associate the text with a Trip or a Trip Day
- Implement functionality to share primary trip text data to external apps

Task 15: Configure App Signing Configuration

- Signing configuration required by the project rubric will be configured
- As required, the keystore and passwords will be included in the repository
- As required, keystore will be referred to by a relative path