GitHub Username: jmunoz15

Pooling Clans

Description

Carpooling. Free rides. Sharing - a new, helpful carpooling app.

Pooling Clans is a helpful app to share and get rides with people who have some things in common with you. Reduce your green footprint sharing your car or having a ride with a friend, co-worker or classmate and improve the traffic in your city.

Intended User

Anyone who wants to reduce his/her green footprint and cooperate to solve the traffic problems in all the cities around the world.

There are two kind of users for this app:

- The first intended user is any person with a smartphone and a car who wants to offer free rides for the members of the clans she/he is part of.
- The second one is any person with a smartphone who wants to get rides from the other members of his/her clans.

Features

Login
App will allow users to login with Google and Facebook authentication.

Rides
Users can see a list of available or planned rides shared by members of their clans. They also will have an option to create a new ride and open the details of a ride.
A ride contains the following fields:

- Route displayed in a map
- Origin, Destination, Date and Time
- Driver and members using the ride
- A button to request a ride
- An option to score the ride

If all the members in the ride are agreed, the route could be changed in order to add stops or reduce the estimated duration.

Clans
Users can create clans or communities to share rides. Anyone in the clan can invite other users to join the clan, for security reasons, the app will display which user is inviting a new person to join the clan.

Admins of the person who invited a user to the clan are able to remove a user. Also, the user can leave the clan when he/she wants.

A Clan has the following data:

- Image, Name and Description

- Option to invite users
- Option to see the members of a clan

Profile

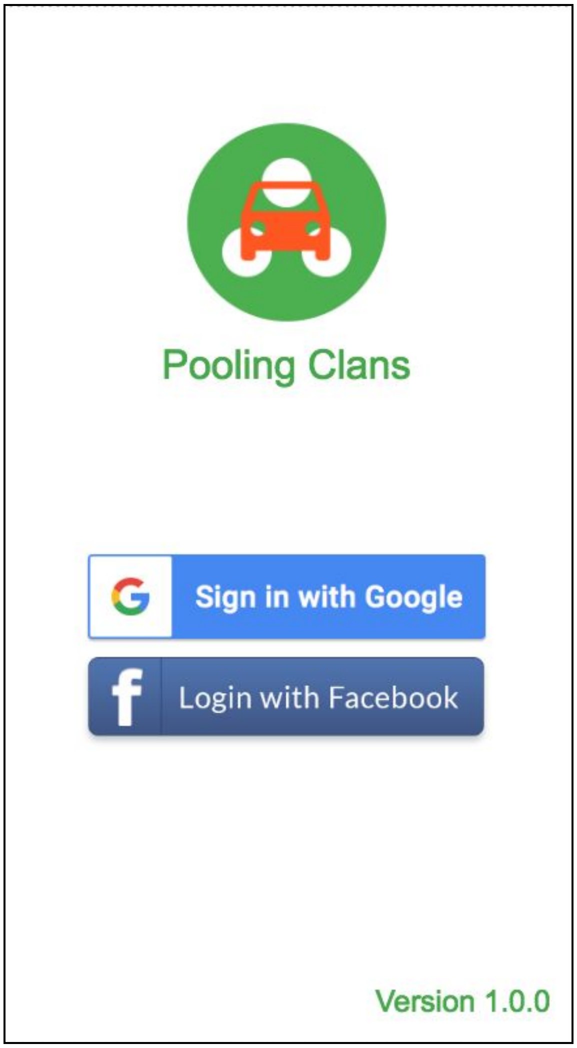
Users will have an option to modify and make some data visible for all the users of their clans. This view will display the contact info (phone, email, skype, etc), the profile picture and the current score of the user (Average of all the rides of the user, including driver and passenger)

Widget

The app will have a widget to present a list of nearby rides using geofencing features. The user will be able to see nearby rides on the widget. If he selects a ride, the ride detail view will be open.

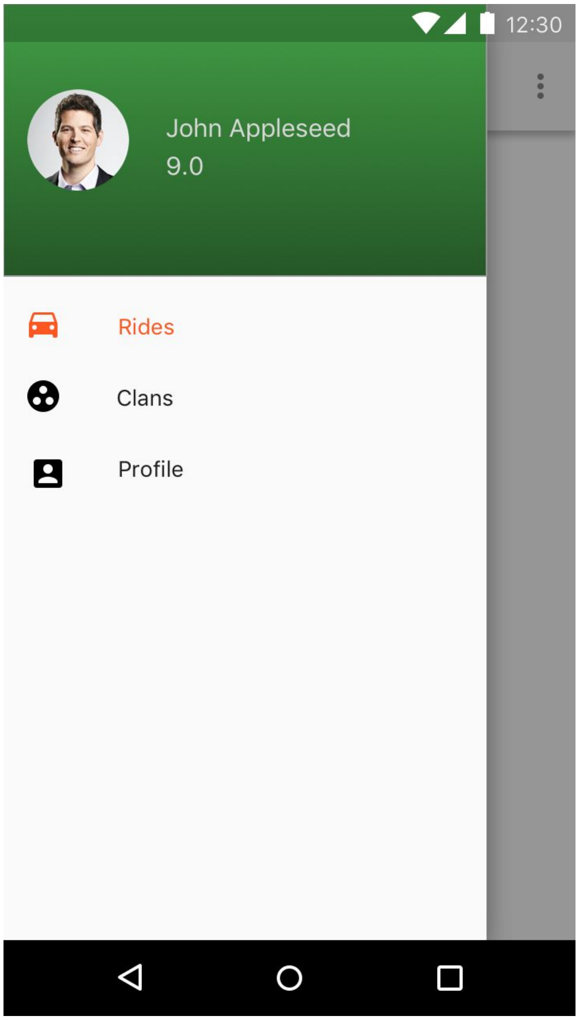
User Interface Mocks

Login Screen



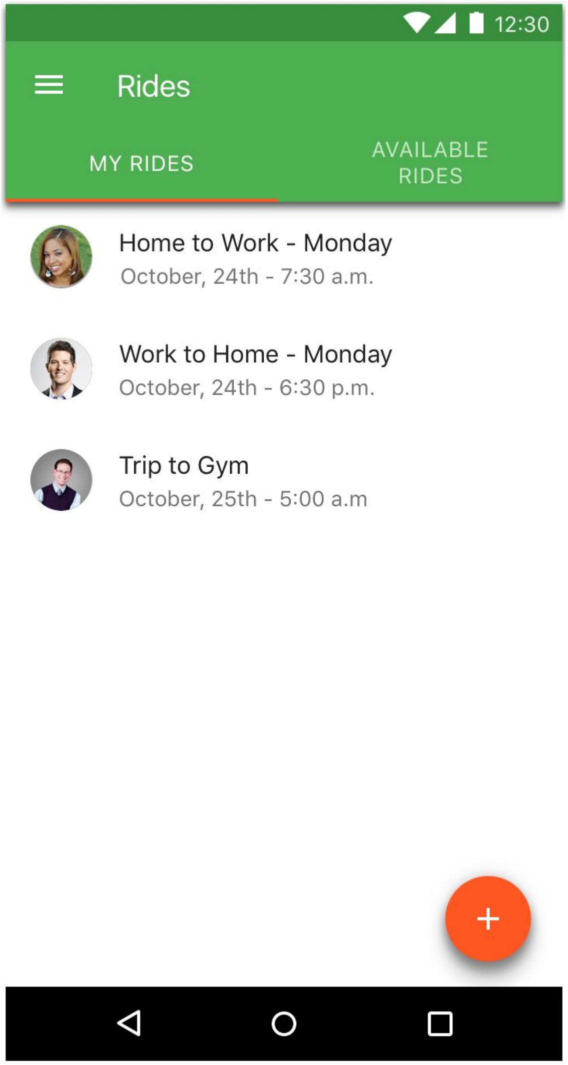
The login screen has the same design for both, smartphones and tablets. It provides the option to start a new session in the app.

Drawer Menu



The drawer menu displays a summary of the profile, also the different options the user have to use the application (Rides, Clans and Profile settings)

Rides



12:30

Rides

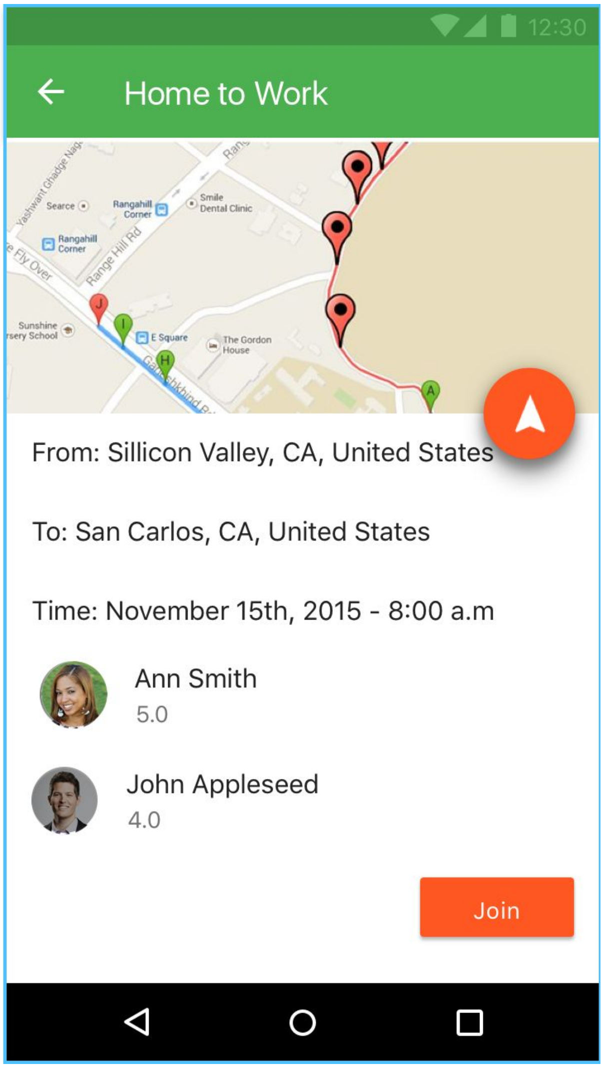
MY RIDES

AVAILABLE RIDES

Home to Work - Monday
October, 24th - 7:30 a.m.

Work to Home - Monday
October, 24th - 6:30 p.m.

Trip to Gym
October, 25th - 5:00 a.m



12:30

Home to Work

From: Sillicon Valley, CA, United States

To: San Carlos, CA, United States

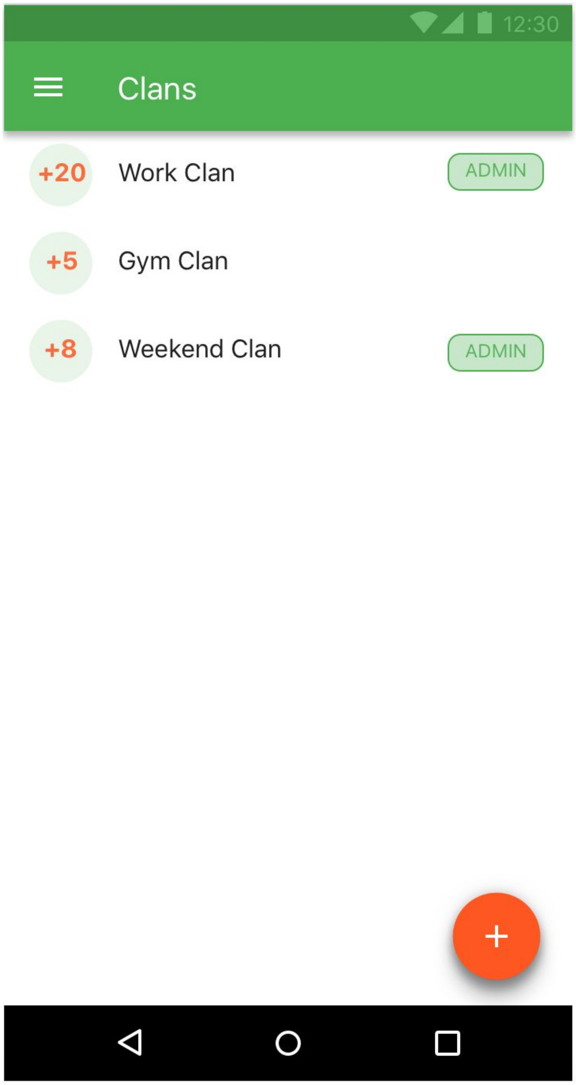
Time: November 15th, 2015 - 8:00 a.m

Ann Smith
5.0

John Appleseed
4.0

The rides list display a tabhost with the list of rides created by the user and the available rides created by other users. The ride detail presents a detail with all the info and members of a ride. In a tablet, the app will present the same views but using a master detail navigation

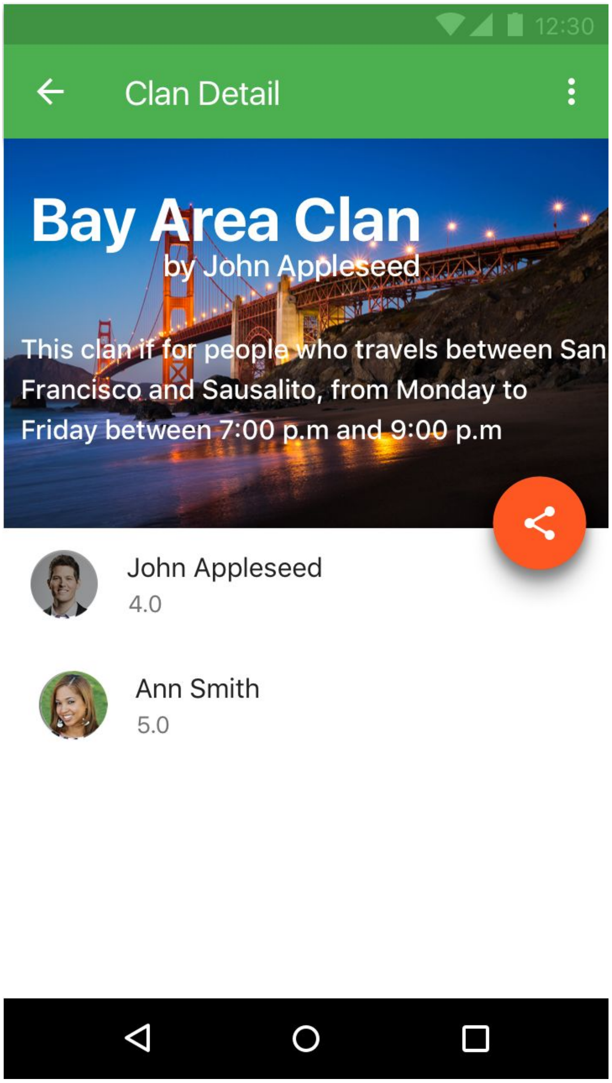
Clans



+20 Work Clan ADMIN

+5 Gym Clan

+8 Weekend Clan ADMIN

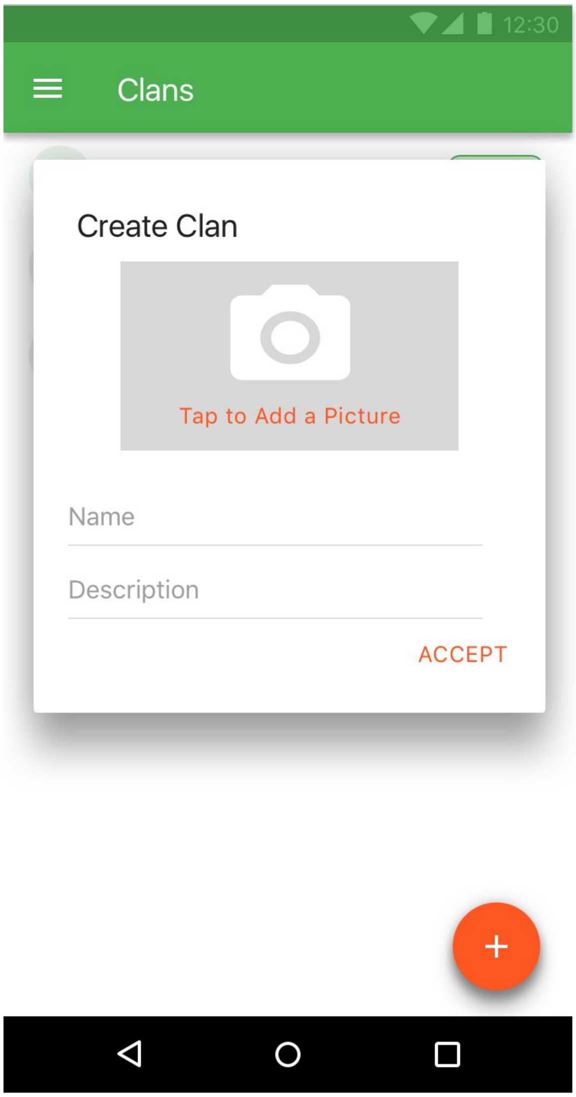


Bay Area Clan
by John Appleseed

This clan if for people who travels between San Francisco and Sausalito, from Monday to Friday between 7:00 p.m and 9:00 p.m

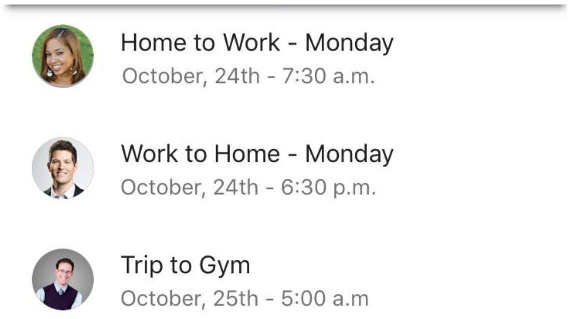
John Appleseed
4.0

Ann Smith
5.0



There are three views for clans, the list, the detail that shows all the information for a clan and the popup to create a new clan. In a table the list and the detail will be shown using a master detail navigation.

Widget

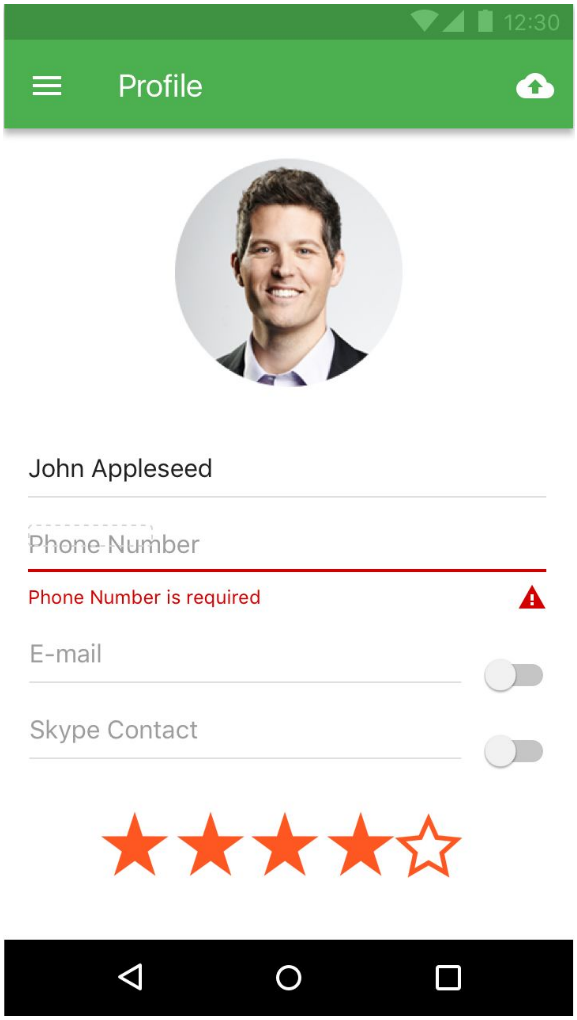


Home to Work - Monday
October, 24th - 7:30 a.m.

Work to Home - Monday
October, 24th - 6:30 p.m.

Trip to Gym
October, 25th - 5:00 a.m

Profile



John Appleseed

Phone Number

Phone Number is required

E-mail

Skype Contact

The profile allows the user to update his personal data and share it with other users. It also displays the current rating.

Key Considerations

How will your app handle data persistence?

The app will use firebase as a back-end system in order to store the data in the cloud. Locally, the app will have a Content Provider and a SQLite database.

Describe any corner cases in the UX.

For example, how does the user return to a Now Playing screen in a media player if they hit the back button?

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

- Glide (Image loading and caching)
- Retrofit (Web Services consumption)

Describe how you will implement Google Play Services.

Location and Context
The app will use the current location of the user in order to automatically display the origin for a ride.

Maps
The app will use the Google Maps API for Android in order to display the route of a ride.

Google Analytics
The app will use Google Analytics to get some information of the users and their interaction with the app

Next Steps: Required Tasks

Task 1: Project Setup

Configure all the web projects and libraries required to build the project. Also, define the database, configuration and content provider to store the data locally

- Configure Firebase Project
- Create Android Base Project

- Add dependencies
- Setup app flavors and configurations
- Create Base Fragments and Activities (Including Drawer Menu)
- Create the Database
- Create the Content Provider

Task 2: Login

Implement all the methods and create the view to allow users to start a new session. The session will be stored in the device, it means, users have to login only the first time.

- Create view to display login
- Implement Google Authentication with firebase
- Implement Facebook Authentication with firebase
- Handle Authentication response
- Store user account

Task 3: Sync Data

Create a service to sync data with the server in the background

- Create Service to sync data
- Send data to the server
- Receive and store data
- Execute service automatically to sync data

Task 4: Ride List

Create the Ride List to display all the available rides for an user.

- Create loader Load the ride list asynchronously
- Create the layout to display the ride list
- Create the view and adapter to display each ride

Task 5: Ride Detail

Create a base view to edit/create/view a ride

- Create layout to display the ride detail
- Create a fragment to create a new ride
- Create a fragment to edit an existing ride
- Create a fragment to view a ride
- Integration of Google Autocomplete API to enter the origin and destination

- Load ride information
- Store ride information
- Create layout to display members
- Load driver and passengers
- Request a ride
- Score a ride

Task 5: Google Maps Integration

Integration of the google maps library in order to display the route of a trip

- Create layout to display maps
- Add ability to add stops
- Add ability to select routes
- Add ability to change routes
- Draw route in a map
- Send notification when a user wants to change the route’
- Display notification to update a route

Task 6: Display Clan List

Create the Clan List to display all the available rides for an user.

- Create loader Load the clan list asynchronously
- Create the layout to display the clan list
- Create the view and adapter to display each clan

Task 7: Clan Detail

Create a base view to edit/create/view a clan

- Create layout to display the clan detail
- Create a fragment to create a new clan
- Create a fragment to edit an existing clan
- Create a fragment to view a clan
- Load clan information
- Store clan information
- Create layout to display members
- Invite a user
- Remove Users
- Handle role (Admin - Member)

Task 8: Profile

Create a view for the profile administration

- Take Photo
- Load Image from gallery
- Load / Store image
- Create view to display profile
- Save / Load profile settings
- Display current score

Task 9: Widget

Create a view for the profile administration

- Set up Geofence monitoring
- Implement Geofence monitoring
- Load widget data
- Display widget