
GitHub Username: aliudurim

n'Trip

Description

n'Trip is an app destined to bring people with common interests in everyday things together to take trips. nTrip allows users to create trip plans to different places and different purposes ,invite people you may on may not know before. n'Trip is designed to make sure you don't take your next skiing trip alone again. Plan a trip post it invite people meet new friends have fun!

Intended User

n'Trip is for users who take recreational trips often, sometimes when planning this kind of trips is pretty hard to find people, so this app plans to bring these people together.

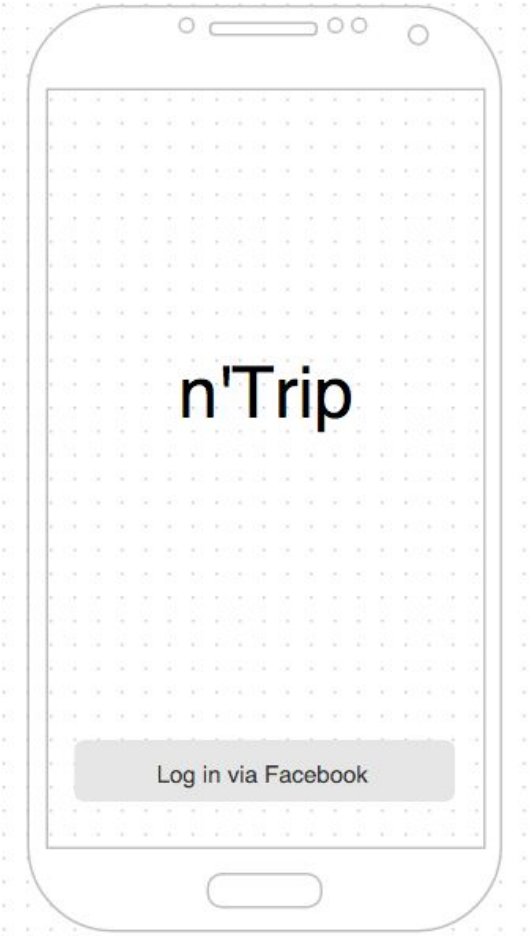
Features

- User can plan a trip by setting the type, date, meeting point and pick up of the trip.
- User can discover trips that are recently posted by other people.
- Users can join trips planned by other people.
- User can select the type of trips he's interested in.

User Interface Mocks

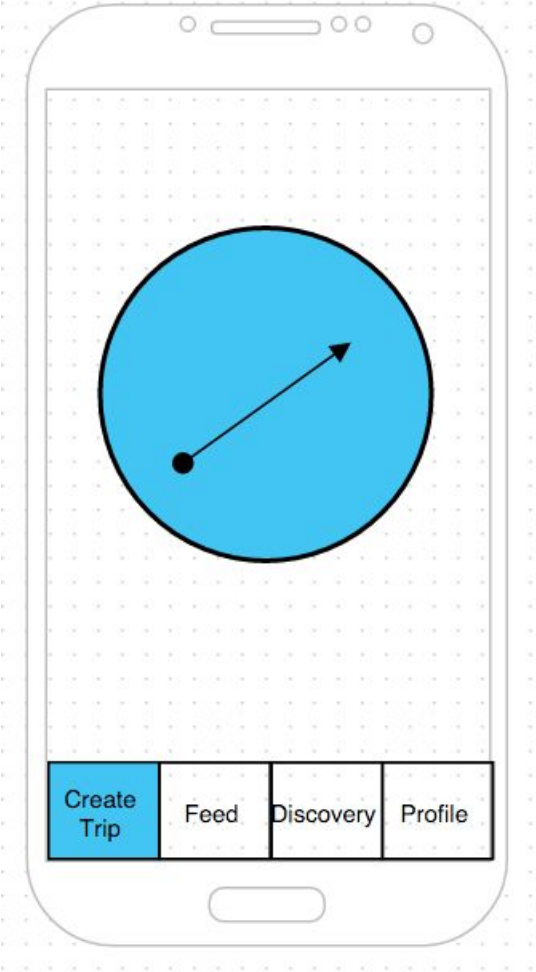
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1



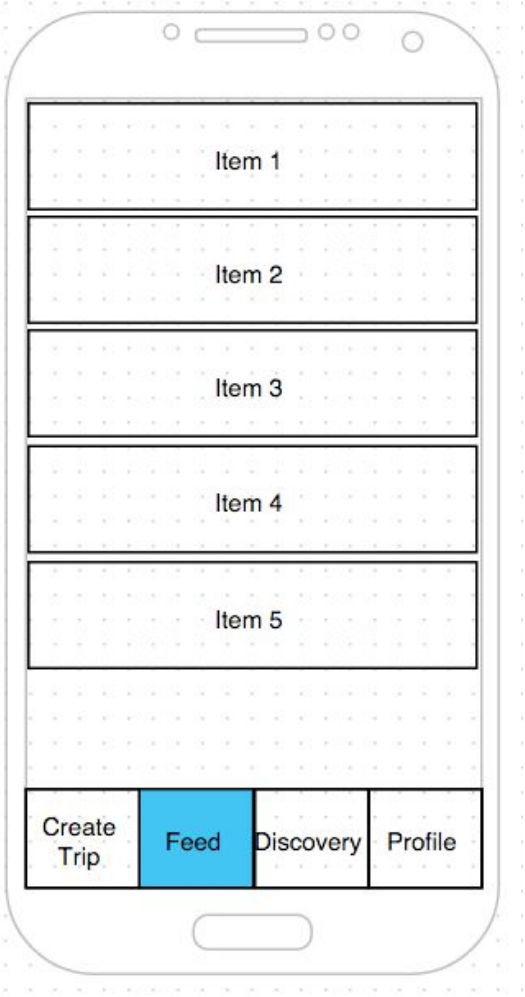
User can login via Facebook no registration needed.

Screen 2



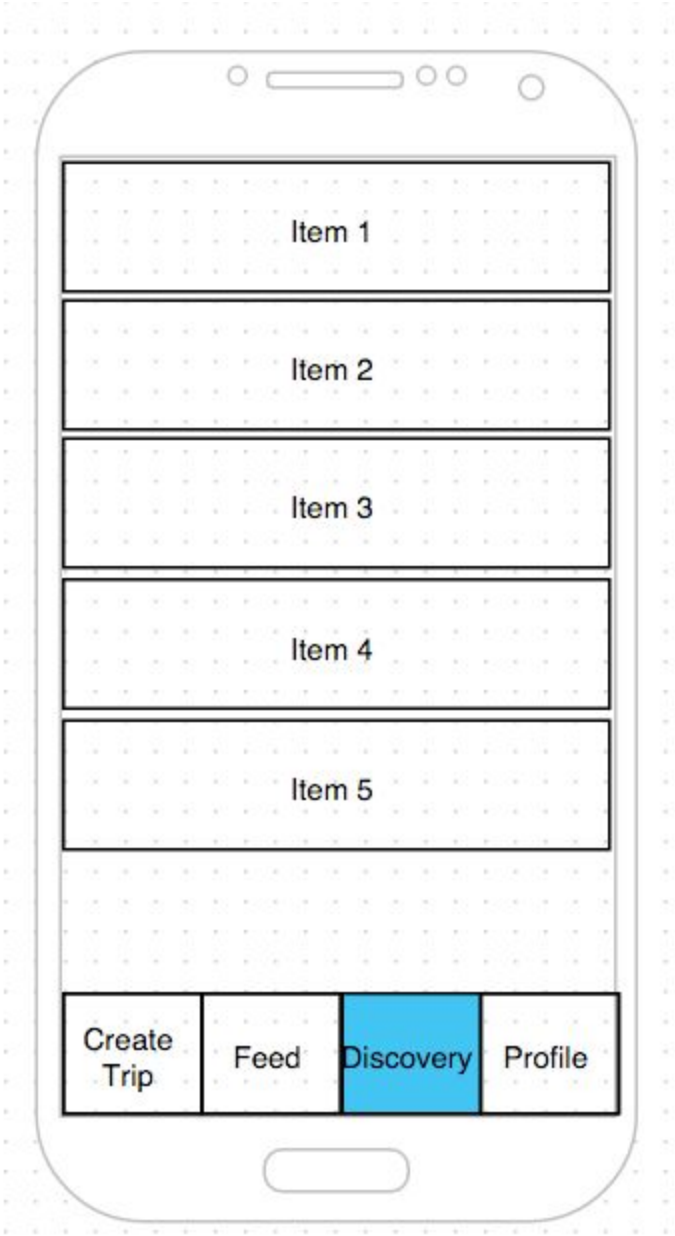
User can tap to blue button to plan a Trip. This also contains navigation tabs.

Screen 3



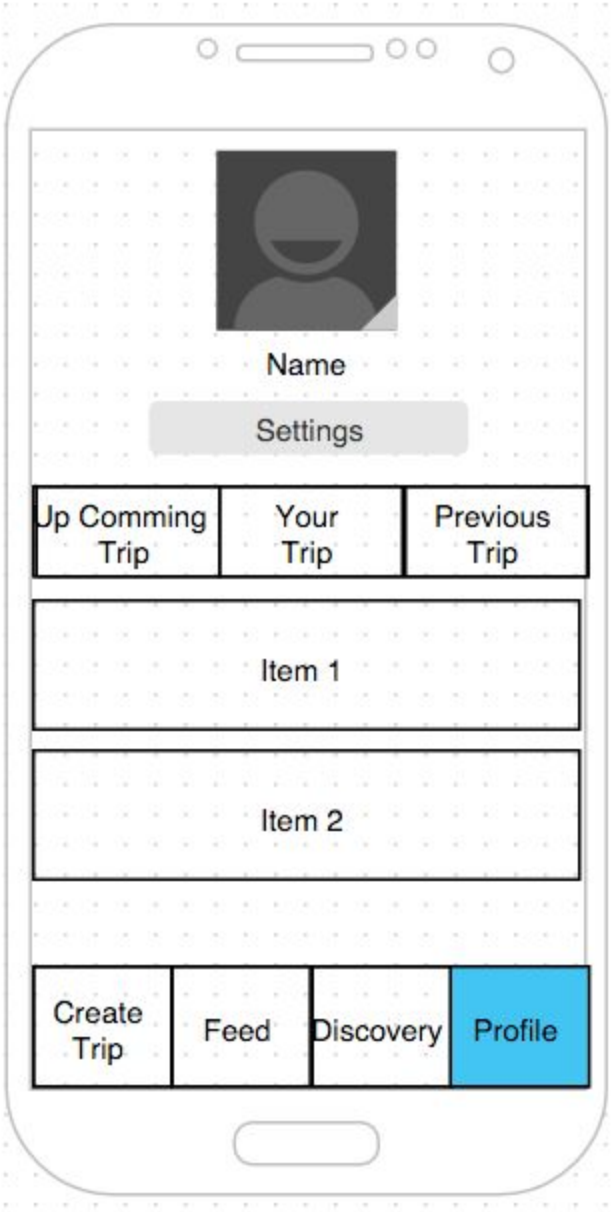
This screen shows the latest trips planned by other users that are open to join for everyone.

Screen 4



This screen shows a list of trips with the same type that the use have selected

Screen 5



Profile screen contains a list to show trips and 3 tabs to select the type of trips you want to see

Add as many screens as you need to portray your app's UI flow.

Key Considerations

How will your app handle data persistence?

The Content Provider for the app will be designed in php and MySQL. The web service serving the app will utilize a php framework such as Laravel.

This approach will allow more flexibility and customization for future improvements and easier implementation of new ideas.

Using free and open source technologies, will prove to be more cost effective in maintenance and scalability.

As the application's main purpose is storing off-device data, and connecting more users real time, the gathered amount of data will allow to further expand purposes and deliverables of the app.

The web service will serve the client apps via JSON encoded data, which is also human readable and can be adopted to any technology.

Describe any corner cases in the UX.

The users could create a very big amount of trips, which would flood the feed with a very big amount of global trips, which would prove to be very frustrating to a big number of users.

If we suppose that we have a daily number of 500 trips, the users could get lost in the feed and would be very hard for them to choose from different trips presented to them, without going to discovery tab, and filtering trips by their preferences.

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

The app will utilize these libraries, which will improve development time and app consistency

- Picasso (Image Loading)
- GSON (Parsing API Data)
- Percent (UI design based on percentage)
- Retrofit (Http Client)
- Google Play Services
- Facebook SDK

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Project Setup

Create a new android studio project. Setup version control (git using bitbucket or github), importing the necessary libraries into the project. Creating quick helper classes for different methods and functions used through the development.

Task 2: Implement UI for Each Activity and Fragment

- Build UI for Login Screen
- Build UI for MainActivity
- Build UI for Create Trip and Flow
- Build UI for Feed and Discovery
- Build UI for Custom Cells
- Build UI for Profile Fragment
- Build UI for Discovery Preferences
- Build UI for Settings
- Implement custom Animations for Views

Task 3: Data Models and Services

Creating the data models and service calls for fetching data from the API.

- Creating Custom Classes
- Creating Service Calls for retrieving data
- Creating Caching Mechanisms for Offline usage of data

Task 4: Implementing Functionality

After having the UI Setup and the flow between different activities and fragments, we start implementing functionality into the UI.

- Finalize Log-In Flow
- Fetch Feed data and display into the Feed Fragment
- Fetch Nearby data and display into the Discovery Fragment
- Implement "New Trip" functionality.
- Implement Profile Screen and populate it with user's data

- Implementing Settings Screen and the Relevant API calls

Task 5: Finalizing and Testing

After completing the above task, implementation of every screen and functionality should be verified, and tested with real data and real users, to reveal design/development flaws.

- Verifying Implementation of Screens and Functionality
- Testing based on Use-Cases and revealing flaws
- Fixing or re designing bugs and potential flaws.
- Smoke testing the final product

Add as many tasks as you need to complete your app.
