How to Use this Template

1. Make a copy [File → Make a copy...]
2. Rename this file: “Capstone_Stage1”
3. Replace the text in green

Submission Instructions

1. After you’ve completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it “Capstone Project”
3. Add this document to your repo. Make sure it’s named “Capstone_Stage1.pdf”

GitHub Username: Semmi Verian

Pick Me

Description

Human live as a society and as a groups. We have a lot of groups in our live such as, College group, Friend group, Games Group, Forums Group etc. At some point we like to hangout or decide to do something as a groups. We sometimes use some chatting app like Whatsapp, Line, etc to vote for where we should go, what should we eat, how many people can come to the meets, but you know at some point it will become a lot of chat came through in our chat groups and our chat groups purpose change from chatting to voting. Pick Me App is an application for groups who'd like to vote some places to go, what should we eat, and also we could count how many people will join the event and how many people can't. It will also include some discussion about the vote itself. Each people could have more than one groups, one groups could have more than one people. I hope i could get benefit from this with using pro membership or something in the future. They could also share the the app to their friend so they could vote. We will also extract the insight about the result from the vote itself

Intended User

I'd like to target every people who have groups and like to have some voting application when they want to decide something as the groups.

Features

List the main features of your app. For example:

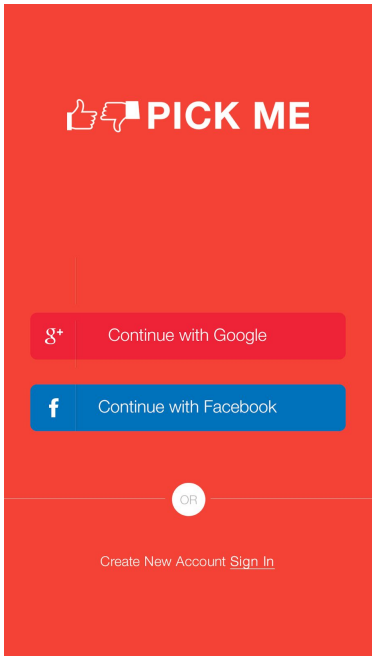
- User can login using facebook, gmail or email
- User can create a lot of their own groups
- User can invite people to their groups
- User can create a new Vote for the groups

- User can create pool whether people could join some event or not in their groups
- User can chat with all of the groups peoples
- User can share the vote
- User can get the insight after they did the vote or pool
- User can receive a notification when new vote is open and when the vote result is came out

User Interface Mocks

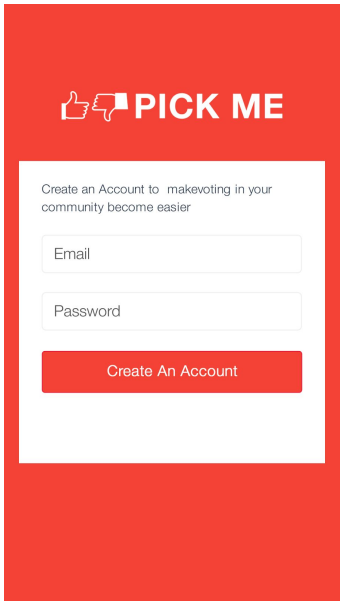
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1



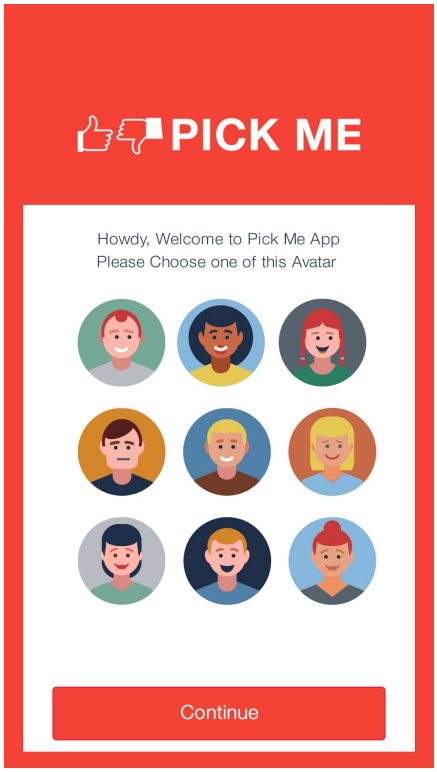
This is the login page of the app, user can logged in by using their facebook or google account, or user can logged in by inputting the email and password.

Screen 2



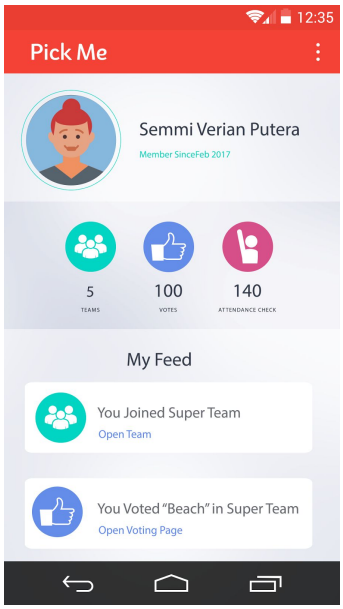
This is the register page if user decide to choose logged in using email and password

Screen 3



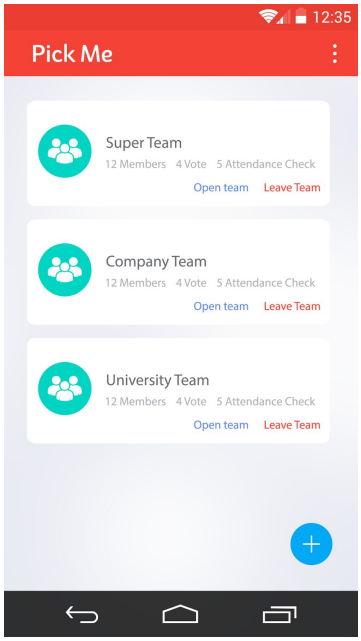
After user log in, they can choose to use the avatar that the app already provided for them, maybe i will add an option to let user upload their own photo

Screen 4



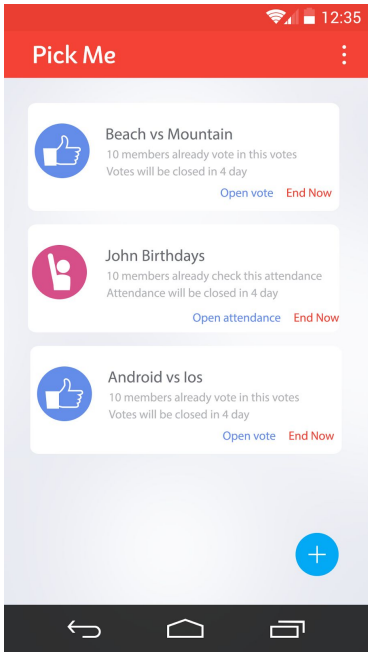
This is the Main View when user already logged in to the app. It will show their avatar, name and info about how long they already become a member in this app.After that we will show the statistics about how many teams that they already join, how many votes that user already did, and how many attendance user already did.In the last section we will show the feed about the latest history about what user did in this app.

Screen 5



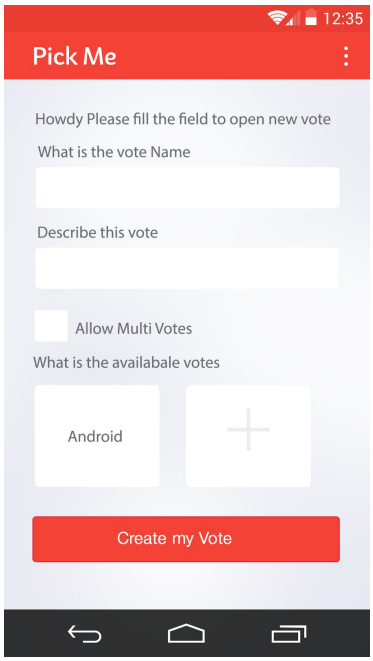
This is the View for user to see their team list that they already join. User can create new team by press the Fab to add the button, after creating the team user can invite the people to join the team.

Screen 6



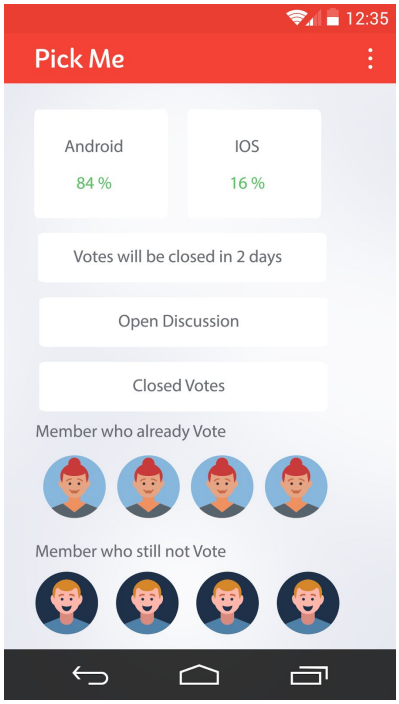
This is the view for the list of attendance and vote in one team, it will show some info and title about the votes/ attendance. User can also create new vote by pressing the Fab button.

Screen 7



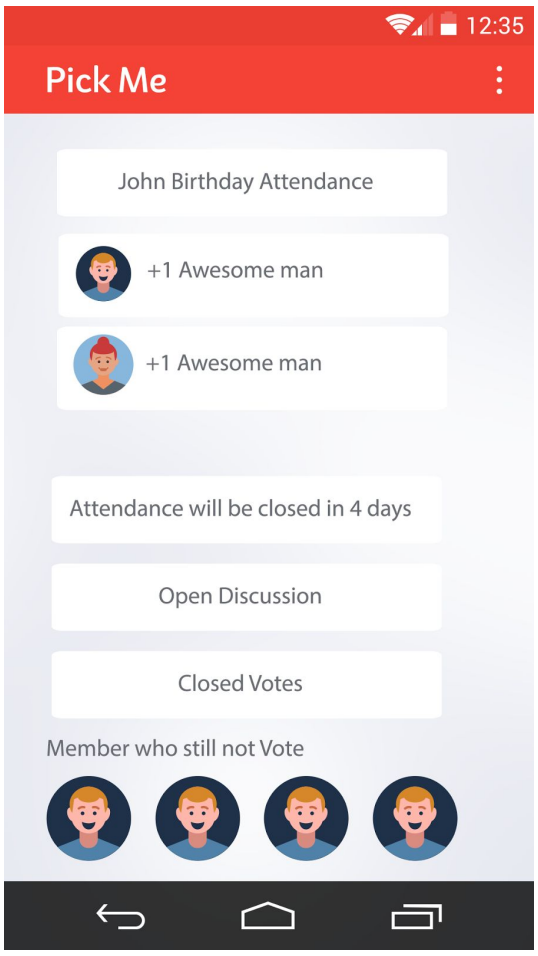
This is the view for creating new votes or attendance. There is one more field that i forgot to put is when the date for the vote to be closed. The available votes will be a recyclerview that allow user to put as many options as they want to the vote section.

Screen 8



This is the view when user opening some votes. It will show the percentage of the current vote result, info about the votes itself. A button to open discussion that will led user to open the vote discussion and an option to let user closed this vote. It also have some information about the user that already vote and the people who still not vote. If the user still not vote, user can send notification to ask them to use their vote.

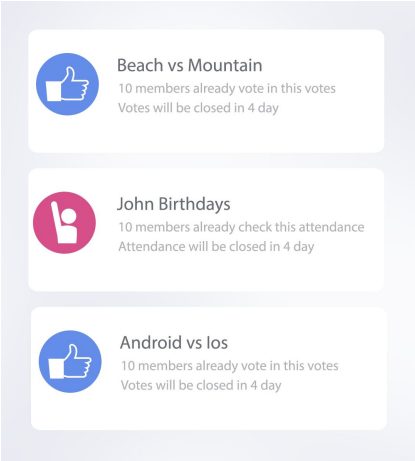
Screen 9



This is the view when user opening some attendance list. It will show people who already check in this attendances. There is also info about the vote, a button to open discussion that will led user to open the attendance discussion and an option to let user closed this attendance. It also

have some information about the people who still not check in and can send a notification to let them decide whether they could check in or not.

Screen 10



This is the widget that user can drag to their home launcher. The purpose of this widget is to let user see their latest votes, so it become easier to user engage with our app

I Have another feature that will be shown in the app but still not creating the mockup like. Inviting user to the team, creating the new team, showing the result of the votes or attendances.

Key Considerations

How will your app handle data persistence?

I am going to use Firebase realtime messages as a way to persist my data

Describe any corner cases in the UX.

When user click back from the vote app will going to return to the groups list activity
The app will support the RTL layout
Using the same font across the app so user don't have a confusion about the app
Following the same material design guidelines

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

- Glide for image caching
- Firebase Auth for authentication
- Firebase Storage for saving images to the firebase
- Firebase Real time Database to persist the data
- Dagger as the dependency injection
- RXjava for controlling data stream (still not sure whether will using this or not)
- Jack Jill for java 8 support

Describe how you will implement Google Play Services.

I am going to use Analytics, Crash reporter for the insight about our app.sing firebase as my main backend services

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

Project Setup and Up and running

- Creating new Project
- Configure Libraries and Dependencies
- Creating Firebase Project
- Inserting Firebase to our App
- Think about the Firebase Data Strutcture

Task 2: Implement UI for Each Activity and Fragment

List the subtasks. For example:

- Build UI for All List
- Set Id to be called in the java class
- Binding Ui and Java Class

Task 3: Implement Dagger and MVP

For creating clean architecture app. We are going to implement dagger as the dependencies, and MVP as the design pattern with help of Rxjava as the datastream framework.

Task 4: Building the App

Building the app including

- Logic to run the app
- The activity stack to be considered in the app
- Synchronous and Asynchronous Task
- Loader and Data persitance
- Notification system

Task 5: Creating the Widget

After the App done we will creating the widget for the user, so they can easily interact with our app without opening it.

Task 6: Testing the App

We will test our app, and if we have time, we will create some unit test to make sure everything run correctyl

Task 7: Uploading to play store

We will upload the app to the play store so people can try and give a feedback about the app

---

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it "Capstone Project"
3. Add this document to your repo. Make sure it's named "Capstone_Stage1.pdf"