GitHub Username: MaximilianFrick

CoffeeTime

Description

Your mates and you have a coffee machine and you want to organise the coffee-making-process? *CoffeeTime* is an app which supports your group, to keep an eye over the process from buying the beans to having fresh nice coffee in your cup.

Intended User

Coffee -drinkers, -enthusiasts, - perfectionists, People who want to stay organized

Features

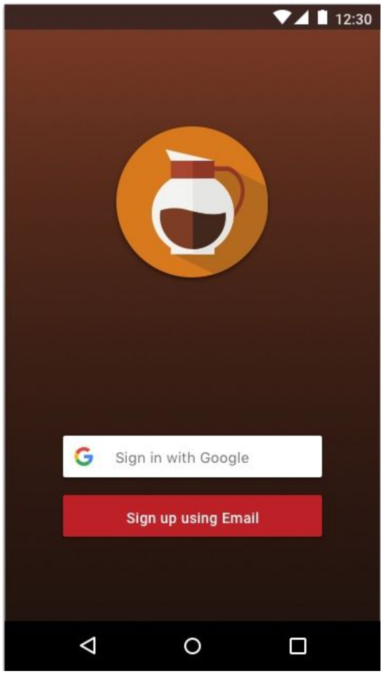
- Calculate amount of coffee needed for next jug
- Track coffee status
- Announce group when coffee is ready

Optional (not part of MVP)

- Rate coffee beans
- Check who is next to buy beans

User Interface Mocks

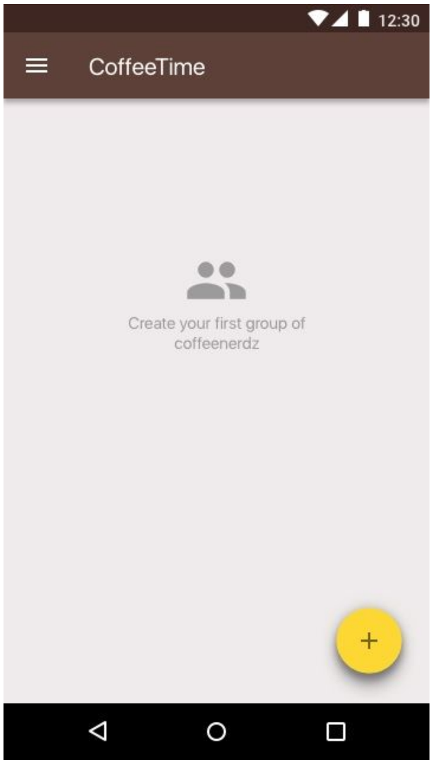
Screen 1



Login:

- Login via Google or Email

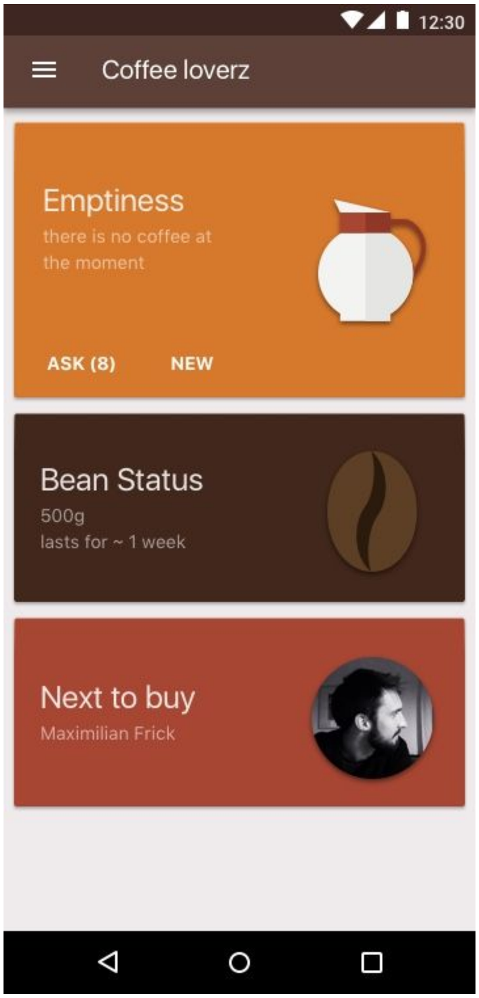
Screen 2



“No CoffeeGroup created”-View

- Possibility to create a new Group with clicking on “+” Floating Action Button.
- After Clicking “+” Dialog appears where you can set a name for the group
- Then after accepting inputs the group will be created.

Screen 3



Emptiness

there is no coffee at the moment

Bean Status

500g

lasts for ~ 1 week

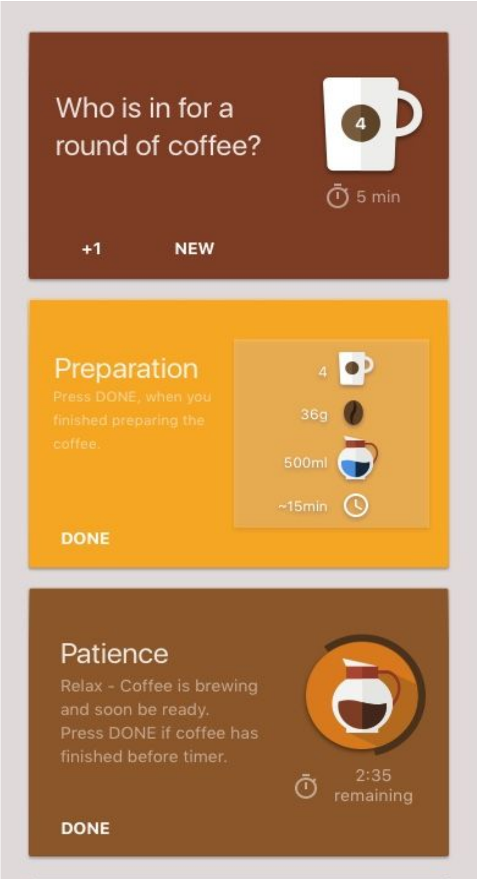
Next to buy

Maximilian Frick

GroupView:

- Information-Tile which displays status of current jug → When clicking “Ask” or “New” → leads to Screen 4
- Display current Stock of Beans (Optional: Display average consumption)
- Optional display who is next to buy coffee beans in the group (Feature not final and maybe not part of the first version of this app)

Screen 4



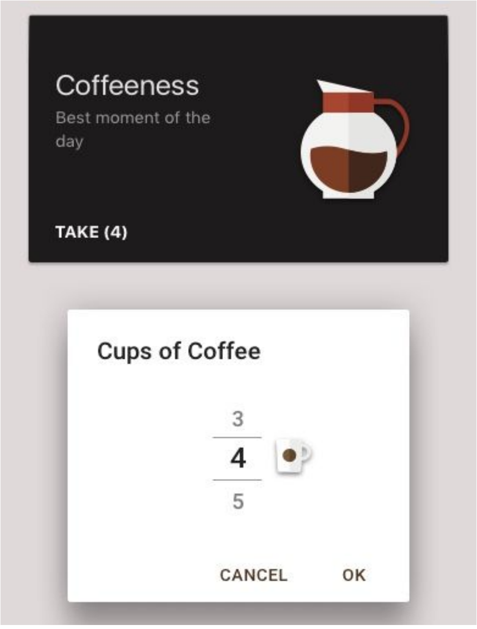
Who is in for a round of coffee?

Preparation

Press DONE, when you finished preparing the coffee.

Patience

Relax - Coffee is brewing and soon be ready.
Press DONE if coffee has finished before timer.



Coffeeness

Best moment of the day

Cups of Coffee

Other States of first Coffeecard:

- State 1 Ask for new Round
- State 2 Summary
- State 3 Ongoing Process
- State 4 Done
- State 5 Empty Jug (Screen 3)
- Dialog to select Number of Cups

Key Considerations

How will your app handle data persistence?

I will use Firebase for datahandling and Notifications.

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

- Retrofit for handling data exchange.
- Butterknife for binding Views
- Dagger for dependency Injection
- Some Stuff from the support library like CardViews, RecyclerViews.
- Also i want to try out Kotlin here as well, as it is now officially supported :)

Describe how you will implement Google Play Services.

Google Cloud Messaging (FCM) for Notifications

Next Steps: Required Tasks

Task 1: Project Setup

- Create project with a blank screen
- Configure gradle tasks for better code quality (findbugs, checkstyles)
- Create Github-Repo and make init commit/push
- Create a Firebase Project and connect it with application

Task 2: Login

- Create UI for Login-Screen
- Implement registration/login with Google+
- Optional: implement email-login

Task 3: Connect/Create Group

- Create layout for Screen 2
- Implement feature to create a group (give a name)
- Implement display the list of groups (Optional: Display information of how many people are in a group)
- Possibility to join a group after clicking a list element

Task 4: GroupView

This is the “MainView”. If the user once created or joined a group he will always get to this screen after starting the app.

Display several Cards:

1. Status
    - Display current status of the coffee jug (most important feature of this app)
    - The different states are described in Screen 4
        i. State 1: New Round - Any user in the group can ask for a new round. Then everybody in the group gets a notification, where you can decide if you want a cup or not. You have also the possibility to directly create a new jug without asking and enter the desired cups of coffee.
        ii. State 2: Preparation - Dependant on the amount of cups there will be informations displayed regarding the making of a new jug. (Water amount, how many beans, how long will it approximately take, …). When the user is done with preparing the coffee machine he can press “DONE”.
        iii. State 3: Patience - During this status the coffee process is in its ongoing state. This is just an indicator for all users that they have to wait a specific time until the coffee is ready. Here there user can always press “DONE” if it is faster than the setted timer.
        iv. State 4: Coffeeness - The coffee is ready and all users who are “subscribed” to this jug get a notification to take a cup of coffee. With hitting the “TAKE” Action button of the Card you can actually inform the others that there are only “2” left for example (See Screenshot) (The number in the bracket displays the cups of coffee which are left)
        v. State 5: Emptiness - Like in Screen 3 when there are no cups left in the jug then there is an empty jug displayed. This is the idle state.

2. Amount of Beans
    - Display the amount of beans which are left → indicator to buy new beans
3. Optional: Whos is next to buy
    - Displays the person who is next in the row to buy new beans

To make this happen a communication needs to be setup to the Real-Time-Database of the Firebase project.

Summary

During the whole implementation phase, there needs to be testing of the features. As there are a lot of features described, i already noticed some parts as optional to not go beyond the scope and make a MVP here. As this is an application, we (in my working company) would actually use, there is a big interest in a further development after creating this MVP, where the optional features will be implemented. Also a colleague of mine is interested in developing an iOS version, as we have a lot of iPhone users as well ;)