GitHub Username: gusdantas

Quaiscalingudum

Description

Sometimes you can't stay a minute more with the person you love because you live far away and if you lose the next bus which passes at 23h, only tomorrow morning.
Other times you just want to be advised up that it is too late for the breakfast and you need to go to work.
Quaiscalingudum is an app which will advise you about the bus you want to get inthe stop you chose to get at the time you need to get.

Intended User

People who needs to get a specific bus at a specific stop in a specific time and needs to be advised with some antecedence.

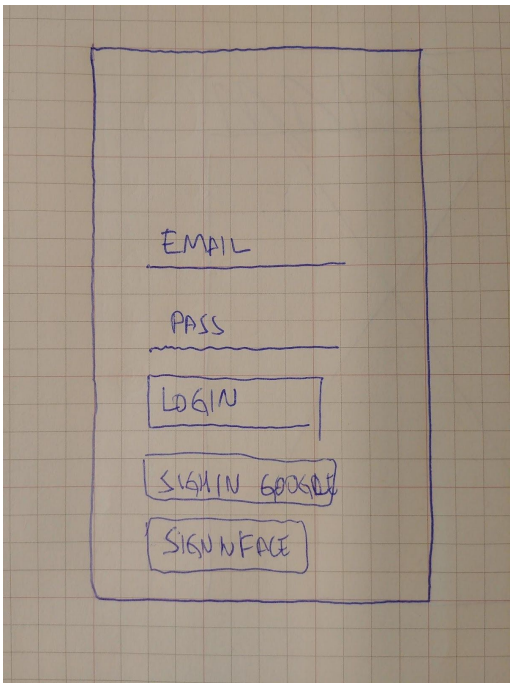
Features

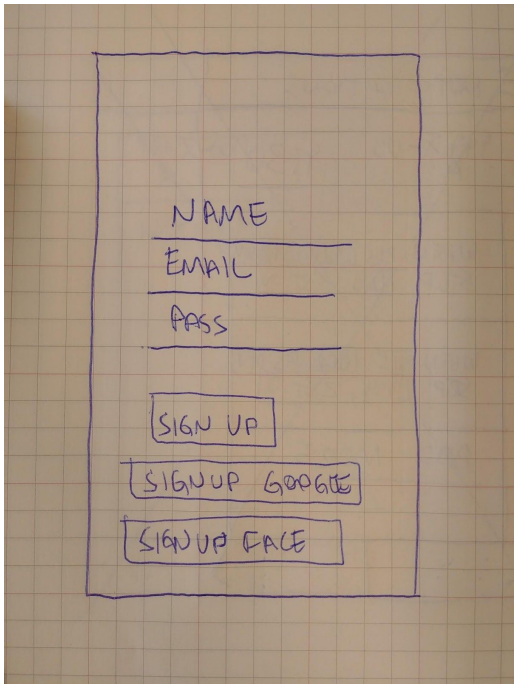
- Notifies the user when his bus departed from any of the stops of its itinerary;
- Notifies the user when his bus is about to arrive at of the stops of its itinerary;

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 Google Drawings, www.ninjamock.com, Paper by 53, Photoshop or Balsamiq.

Screen 1 - Login and sign up

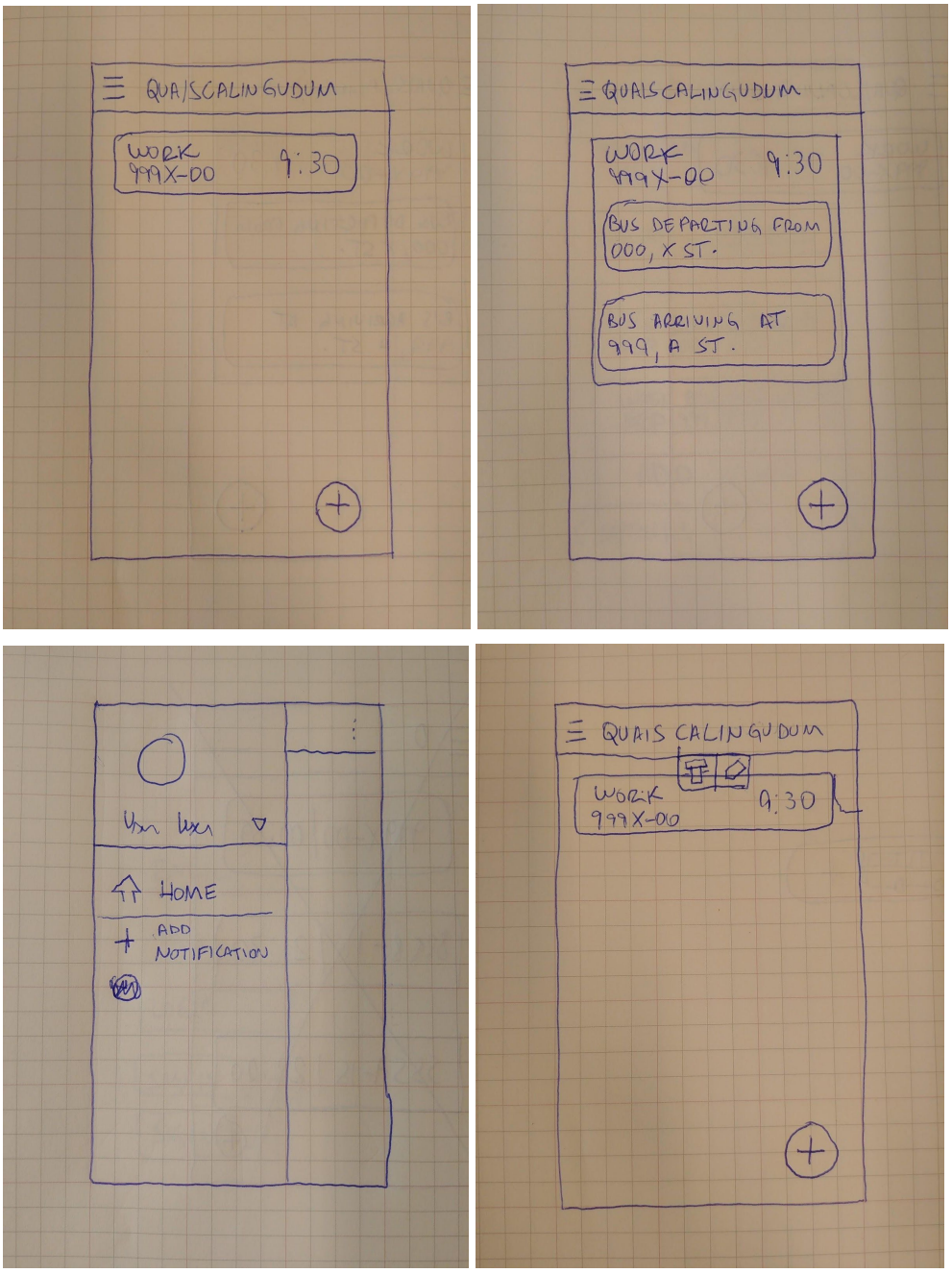




Here the user will be able to:

- Login using Google/Facebook to have access to the app.
- Sign up using email providing a few other infos.

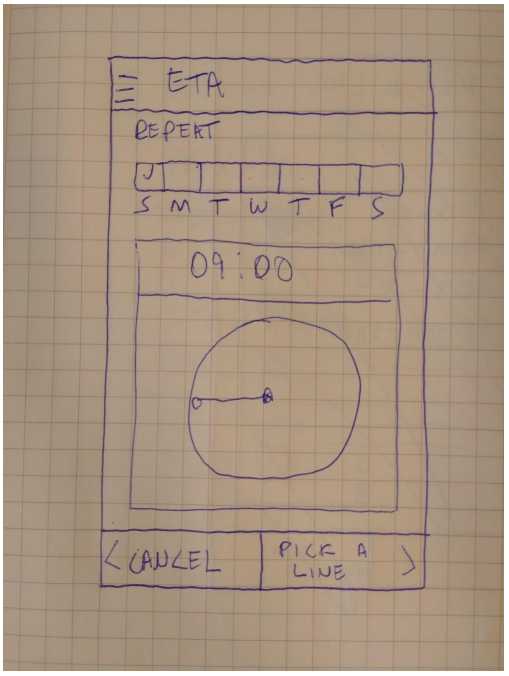
Screen 2 - Main Screen

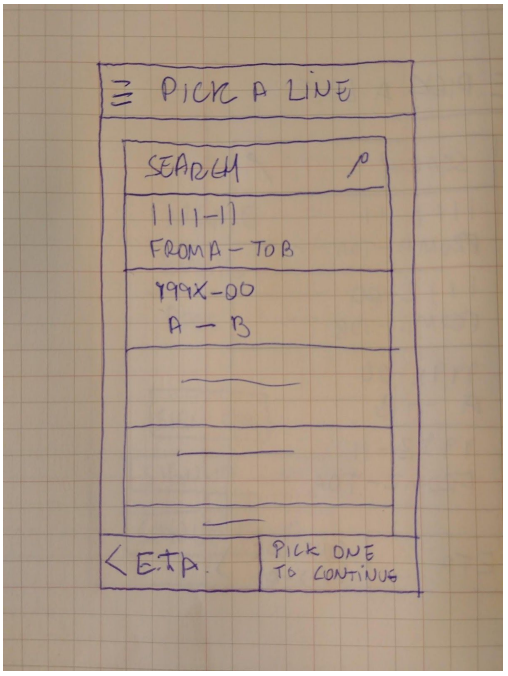


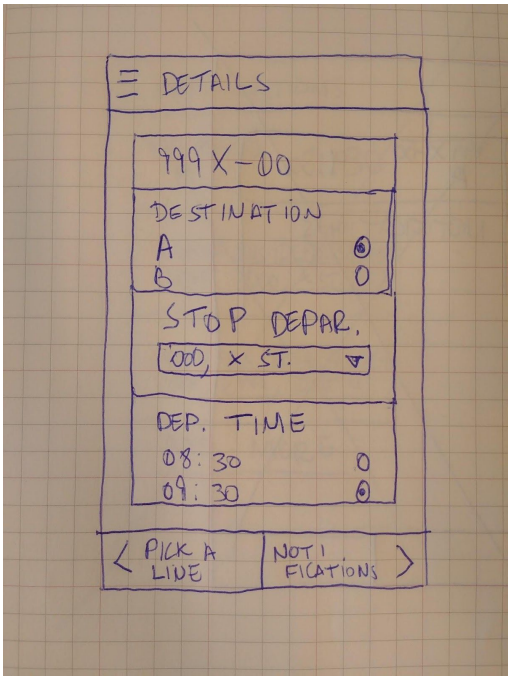
Here the user will be able to:

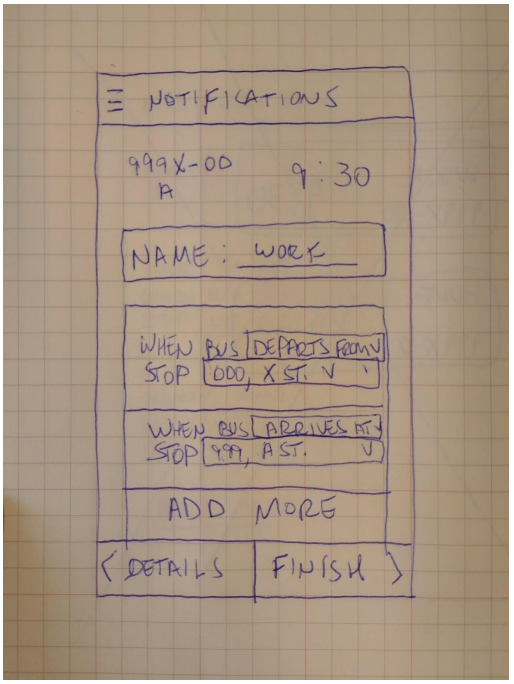
- Check the notifications available;
- Edit or delete notifications;
- View the details of each notification;
- Add a new notification;
- View Quaiscalingudum menu;
- Sign out from profile.

Screen 3 - Add notification





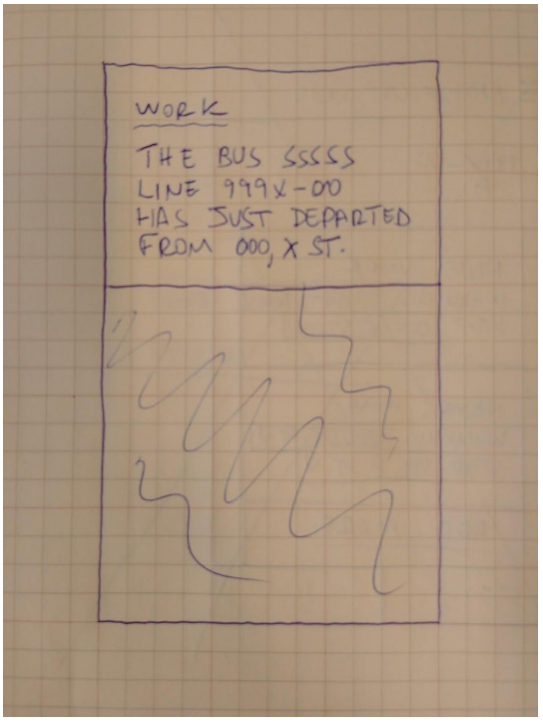




Here the user will be able to:

- Choose an estimated time to arrive on the bus, including if it will repeat any day of the week;
- Choose the desired bus line;
- Choose the one of the destinations of the line, the arrival stop and the departure time (based on the ETA);
- Name the new notification and add them based on the bus position.

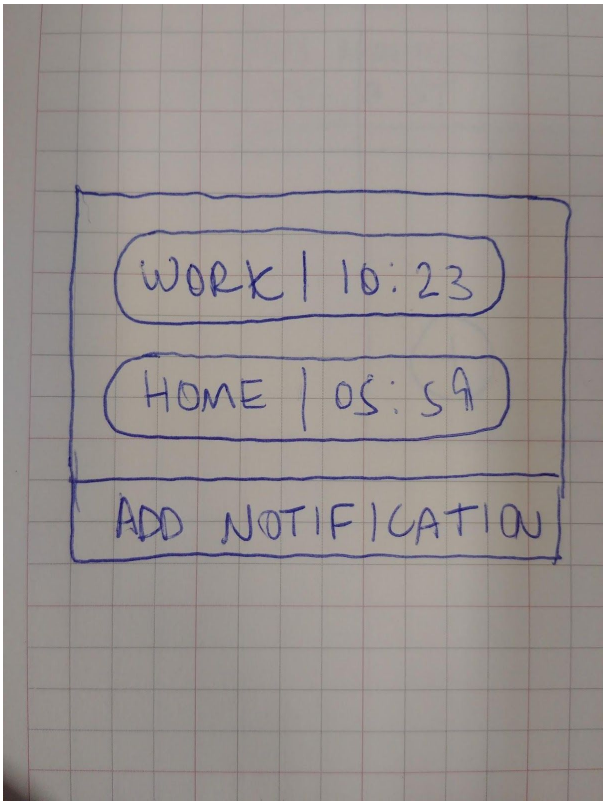
Screen 4 - Notification



Here the user will be able to:

- Choose an estimated time to arrive on the bus, including if it will repeat any day of the week;
- Choose the desired bus line;
- Choose the one of the destinations of the line, the arrival stop and the departure time (based on the ETA);
- Name the new notification and add them based on the bus position.

Screen 5 - Widget



Here the user will be able to:

- Check the active notifications and the remaining time to start tracking;
- Add a new notification.

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

Key Considerations

How will your app handle data persistence?

App will handle data persistence by using Firebase Realtime database to store the user notifications.

Describe any edge or corner cases in the UX.

When using the app without network at the time of the tracking, the user will only be notified that the app will notify him based on the GTFS (General Transit Feed Specification) data, and it may not represent the reality.

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

- Picasso
    - Retrieve, cache and show images, mostly due to its easiness to use and cache feature that will combine perfectly with the offline use of the app;
- ButterKnife
    - Reduce view boilerplate;
- Volley
    - Handle Olho Vivo api calls;
- FastCSV
    - Read GTFS data.

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

- Firebase Realtime Database
    - Store the user notifications;
- Firebase Authentication
    - Access the Firebase Realtime Database to save the user notifications.

Next Steps: Required Tasks

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

Task 1: Project Setup

- Make sure all libs that will be used are in gradle;
- Create initial folder structure.

Task 2: Add Firebase Authentication Module

- Add and connect Firebase to project;
- Build UI for login activity;
- Connect Firebase authentication with the app by using Gmail, Facebook and E-mail authentication;
- Build UI for sign up activity.

Task 3: Develop app UI and logic

- Develop main activity and it's behaviors;
- Develop menu activity;
- Develop add notification activities;
- Develop Firebase Realtime Database connection;
- Develop tracking services.

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