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: ehabhamdy

Rakabny

Description

You are tired of congestions and waiting too long on the street to find affordable transportation to your home or work. Rakabny is here to help you. Scan all upcoming microbuses for empty places on your phone and reserve your seat comfortably right from where you were standing.

The mobile application shows you the number of microbuses in your area and the number of available seats in them, thus you can book your seat easily and don't stand too much to ride thus saving money and time; it's extremely cheap compared to other alternatives, you only pay the same fees and the price of your order.

Not sure how to write a good description? Search 5-star apps on the Play Store for inspiration.

Intended User

Microbus Passengers are main target users of the application

Features

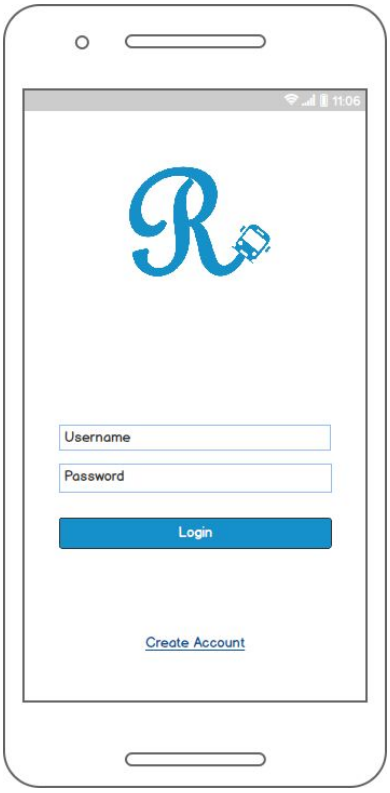
List the main features of your app. For example:

- A map view to scan nearby Microbuses and get information about their destinations
- Reserve a seat if available
- View user rides history
- View user's payment history
- An app for drivers to broadcast their location.
    - Show location of passengers with their identification number who bought a ticket on map
    - View payments status.
- An app for passengers
    - Display main stations on a map
    - Reserve a place in the bus
    - View profile (personal and payment details)

User Interface Mocks

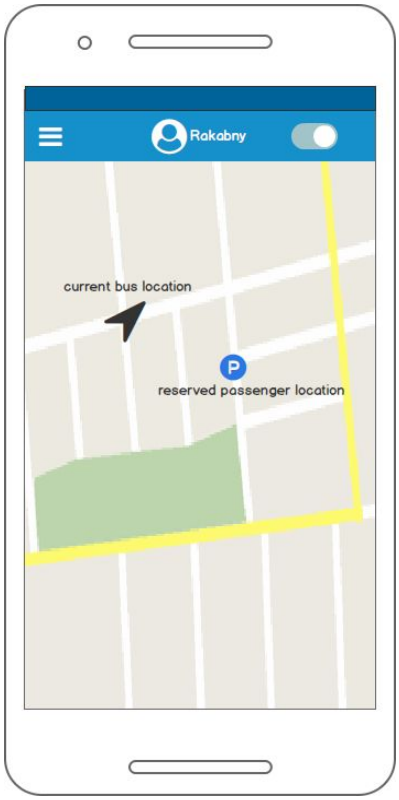
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1 Driver app



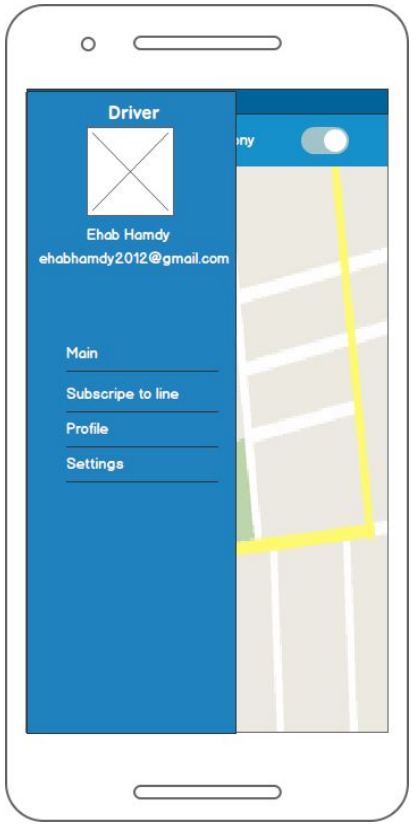
Login screen with firebase authentication

Screen 2 Driver app



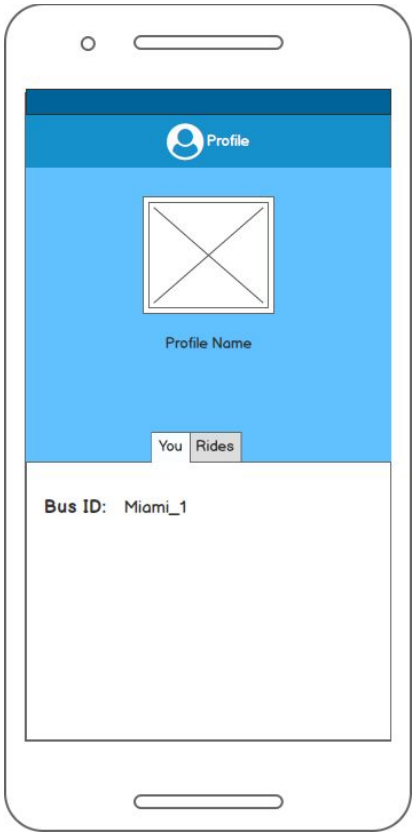
Main screen for the driver app to show his current location. This screen also allows the driver to share his location and to be discoverable by passengers

Screen 3 Driver app

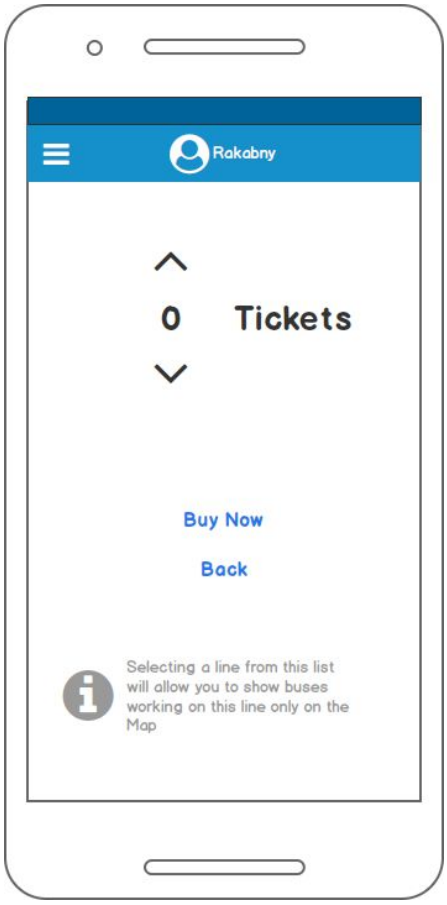


Main navigation menu to get to other important places such as profile or settings

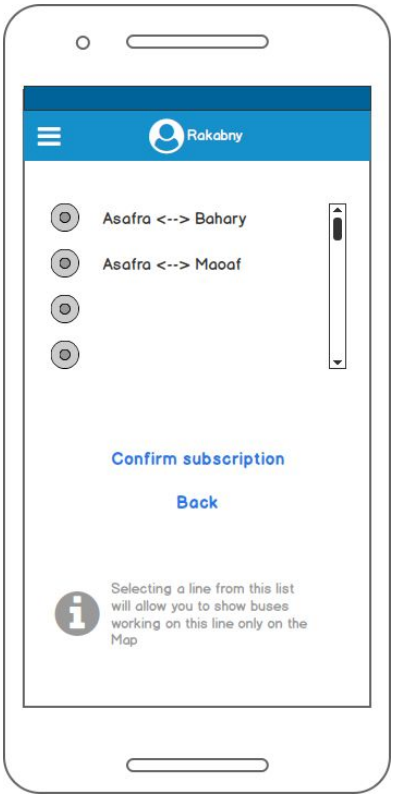
Screen 4 Driver app



Profile view to show personal information, payment status, and statistics about rides

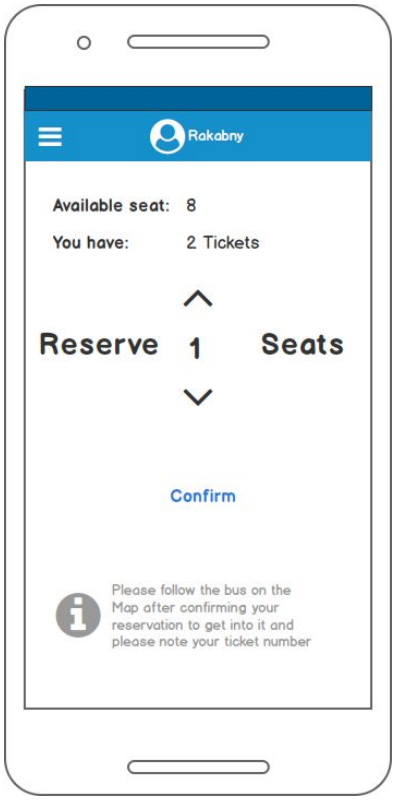
Screen 5 Passenger app main	Screen 6 Buy ticket screen
	
Main passenger screen, the screen shows the location of the buses according to the line subscription it also enable the user to buy tickets and show how many tickets they have	Tickets screen allow users to buy one or more tickets

Screen 7 Line selection

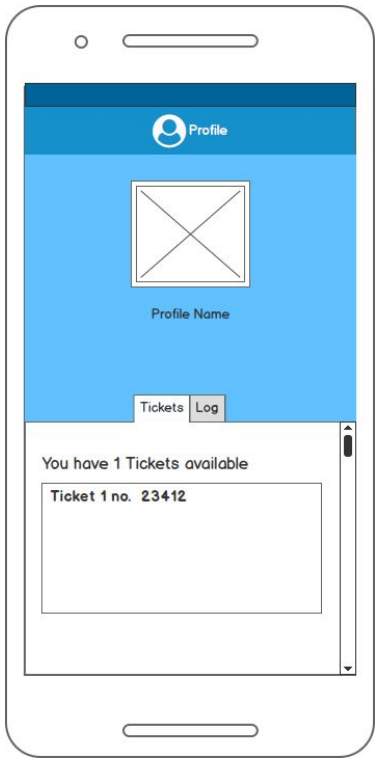
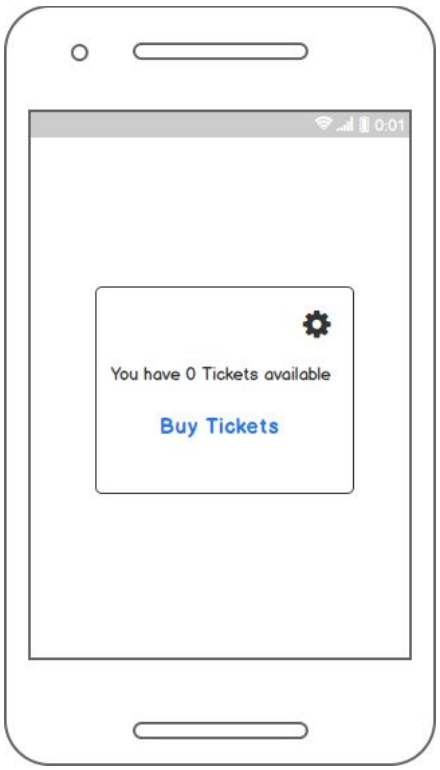


The user can choose a particular line according his desired rout to show all buses on that line.

Screen 8 Reserve confirmation



The user confirm his reservation to a particular bus. The user reach this screen when he click on a bus on the map.

Screen 9 Passenger Profile	Screen 10 app Widget
	
Profile view for the user to display common information.	App widget that show information highlight and easy access to common functionalities

Key Considerations

How will your app handle data persistence?

I am planning to handle data persistence using firebase real-time database and firebase storage. I am also using pubnub API to allow location sharing capabilities.

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.

Mainly I will be using Pubnub and Firebase to support the main functionality of the application. Firstly, Pubnub provide a very elegant solution for sharing geographic location across different platforms. It also provide the concept of publishing and subscribing to channels with minimal sync time. Second, I use firebase for user authentication and to store profile information.

Describe how you will implement Google Play Services.

My app is relying on Google maps in many situations and it is considered the main point of interaction. In addition to that, Location services plays a critical rule in the application. Locations are captured from gps, depicted on the map and published throughout pubnub channel to be ready to received by clients. I am also planning to introduce ads with admob.

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

- Setting up backend to obtain API Keys
    - Creating Google API Console project and enabling Google Maps
    - Creating Firebase project
        - Database info. For driver:
            - Bus ID
            - Number Seats Available
            - Location of passengers reservations with this Bus ID
        - Database info For passenger
            - Name
            - Phone number
            - Email
            - Tickets (list of ticket numbers)
    - Creating Pubnub project
- Setting up Android Studio Projects.
    - Create a project for the Driver and another one for the passenger usage.

Task 2: Implementing the main screens

- Building the main activity for the driver app
    - Setting up Google maps to display a map view
    - Setting up Location services to capture the current location periodically

    - Setting up Pubnub to publish the current location, Each bus publish location in a unique channel with the following naming schema (line-name_bus-number)
    - Adding a drawer menu.
- Building the main activity for for the user app
    - Setting up Google maps to display a map view
    - Setting up Location services to capture the current location periodically
    - Configure Pubnub to receive the location of buses
    - Add riding location hints on the main Map screen
    - Add a link to another screen for popular places to get rides
    - New ride locations are added in Firebase database.
    - Subscription channel names for tracking buses is also saved on the firebase database and the user can view all buses in a particular line and new buses that are joining the service will appear to all user without updating the app itself
    - Adding a drawer menu to access profile, history, payment details
    - When a user click on a green bus marker (green means there are available places) the user will be redirected to a screen that show number of available seats and allow him to reserve seats according to number of tickets he bought
    - Data fetching from Firebase database are always done asynchronously in a background thread also I setup firebase listener to continually listening to database changes and updates the UI accordingly.

Task 3: Implementing user authentication

- Setting up Firebase project and enable email authentication
- Building the login and signup activities.
- Integration with the main screens

Task 4: Implementing buying tickets screen

- User determine number of tickets he wants to buy
- User directed to enter payment details
- User confirm payment

Task 5: Developing user profile screens

- Setting up Firebase real-time database.
- Building the UI for the profile screen.

Task 6: Developing a Payment workflow simulation

- Simulating the process of payment for the restrictions of time constraints given my unfamiliarity with implementing payments

Implementation:
https://github.com/ehabhamdy/DriverBroadcast
https://github.com/ehabhamdy/Rakabny

(For testing purposes I provided signed apk so the process is fast and no need to worry about api keys for the pubnub or firebase)

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**"