DRIVE

Alven Enwya

Project Work in Android Development

Table of Contents

Introduction

The world is becoming more connected and with this comes the ease of use of cellphones, in many cities around the world a user can simply open their phone and click a button to call a taxi to their location for a quick and easy ride. Currently in Jönköping there are no good options for this, as companies such as Uber simply does not operate in the area. DRIVE aims to solve this predicament by offering a simple and easy to use taxi service.

DRIVE is for people who do not like the traditional taxi system or the messy and slow public transport systems in this city, DRIVE offers a quick and easy way to order a ride to go wherever you want to go, in no time at all.

How the DRIVE application work?

The application functions in two ways, either as a “taxi” tool to fetch and pickup customers or people who need a ride. But it also works as a tool for regular people who simply need a taxi to go somewhere. The application starts you off with a splash screen for some seconds, login screen once you log in you can then choose whether you are a Driver or a Customer. Choosing your role here allows you to operate the different functions the roles have.

Driver: The Driver checks in to the application. From “settings” Driver can chose Name, Phone number, photo, car brand and car size. After hitting “confirm” button the Driver will go back to the map. And now Driver can turn on switch “Working” to be online as available to customers. A Mission appear when Customer order a ride, and it consists of a Customer Name, Phone number, photo, And the Customer destination. On the map it shows customer current location.

After accepting a mission by hitting the “picked customer” button, the driver will be directed to google maps where a route has been laid out to get to the customer. Once the Driver has reached the customer the driver may switch over to a second view where the customers route of travel is laid out. After the Driver and Customer has arrived at the destination the mission is completed. And all the trip information it will be saved to both Driver and Customer in history. and the Driver will be automatically available to take another Mission.

Customer: A customer view is rather simplistic in comparison to the Driver windows. A customer simple from “settings” Customer can chose Name, Phone number, photo. After hitting “confirm” button the Customer will go back to the map. From map Customer can selects his or her wished Destination and choose size of car. The Customer then clicks “Call a Driver”, and a request is sent to a nearest Driver.

The customer gets a confirmation and then only must wait until the Driver shows up. Driver information will automatically appear on the screen such as Driver’s photo, name, car brand and Drivers ratting. The Driver will pick up the customer and drive to the destination. After the Driver and Customer has arrived at the destination the mission is completed. And all the trip information it will be saved to Customer

“History” and there can the Customer rating the trip and Driver. Customer now can please another order.

Overview

DRIVE uses Firebase databases to communicate between different platforms and units. A Customers personal travel data is sent to a server, and then a Driver can fetch said data. The Driver constantly gets updated by new information from the server, such as new incoming missions.

Personal information and mission information it can be saved and displayed for both Driver and customer.

When customer request accepted by Driver, personal information will be displayed for both so they can decide to continue the mission or just end it.

Mission information in “history” for both Driver and customer it contains destination, time and date, photo, name, phone, distance length in kilometers and the trip start and end in the map.

The application also heavily uses Google Maps API to communicate destinations, locations and routes to customers and drivers alike. In the DRIVE app the Customer that ordered a ride can always see very easily when a Driver has accepted their request, and where the incoming driver is currently located.

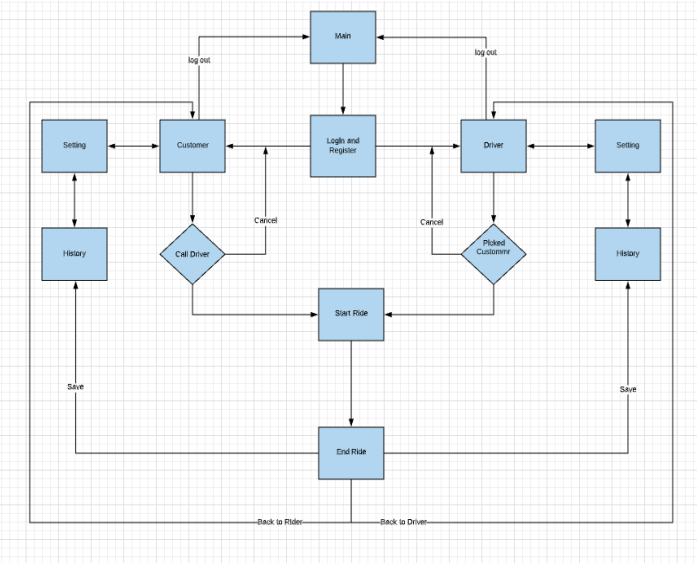
Ratting system at the end of the trip customer from “history” can evaluation the whole trip with this driver by give him stars from 1 to 5. When the new customer orders a ride the driver information will appear with the average of all evaluation the driver received.

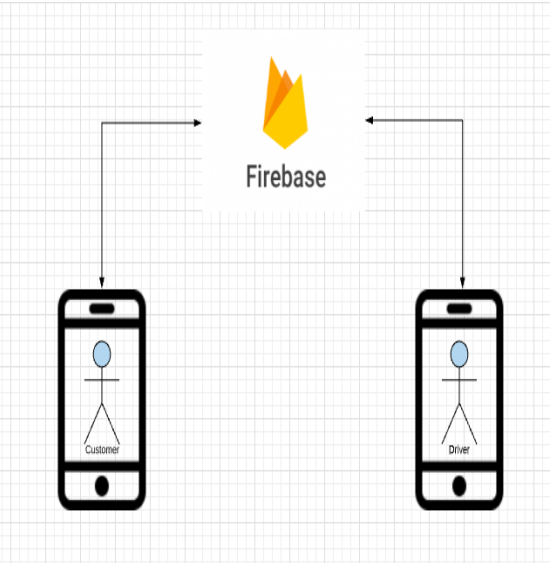
To use Drive app must have an account. Create a new account by writing e-mail and password then chose roll customer or driver and press “register” button to start Drive with new account. if user have a google account it will work too just choose the roll before signing in then the app will start.

Permission is needed to use GPS. Drive will ask for permission from user to use GPS and perform tasks that needed to.

When the customer orders a ride, the request will send to nearest driver who matches the size of car customer choose.

Below are some graphs of the DRIVER application, it contains ER-diagrams, to show how the application work.





Application

The DRIVER application is build using Android Studio programmed with Java. Application utilizes the Android API Oreo 8.0 due to the fact that Googles new guidelines for Google Map application requires that you use 8.0 or up to be able to upload the project to Google Apps Store.

DRIVER is compatible with any and all screen sizes, as long as it is big enough to see a map. The user experience is not changed by screen size at all since the app has scroll views and good graphical features. The app can be used in Vertical and Horizontal mode.

Drive application is supporting two language English and Swedish.

Data base Drive use Firebase Database to authenticate users' picture (DB7), storage to upload photos picture (DB6) and Realtime Database to save users info, history and start end missions.

Realtime Database

The data base name is (driver4-10631) picture (DB1) and it have 5 tables. Users, history, drivers Available, customer Request and driver Working.

Users table have two column Customers and Drivers picture (DB1).

Customers every customer have ID picture (DB2) and some value like history id to know which missions this customer is engage in, name, phone and profileImageUrl picture (DB3).

Drivers every driver have ID picture (DB2) and some value like history id to know which missions this driver is engage in, name, phone, profileImageUrl, rating and service (car size) picture (DB4).

history table have history ID picture (DB2) and every ID have value. customer ID, driver ID, destination, distance, rating, timestamp (date and time) and location from (longitude and latitude) to (longitude and latitude) picture (DB5).

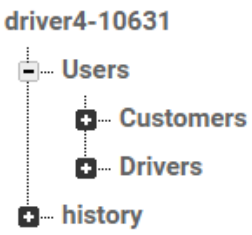
drivers Available have driver ID to show which driver is available value longitude and latitude for driver picture (DB8).

customer Request have customer ID to show which customer order a ride value longitude and latitude for customer picture (DB9).

driver Working have driver ID to show which driver is busy now with mission. Value longitude and latitude for driver picture (DB10).

When the customer sends request and driver accepted the driver will no longer be available to take any other request. Therefore, change status for driver from drivers Available to driver Working to avoid taking any request during the mission picture (DB11).

Database screenshot

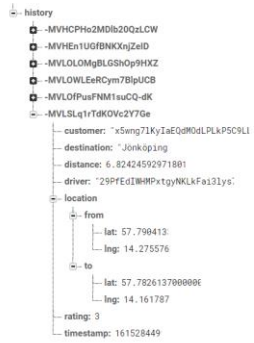


picture (DB1)

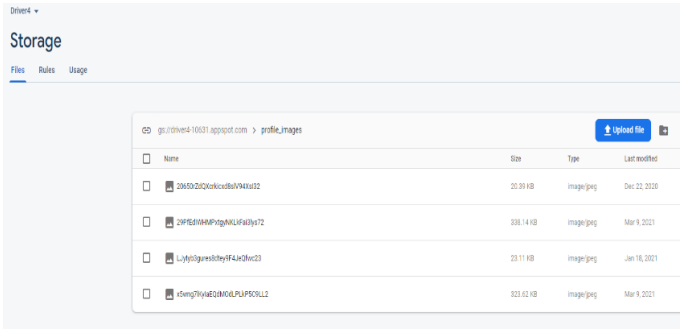
picture (DB2)

picture (DB3)

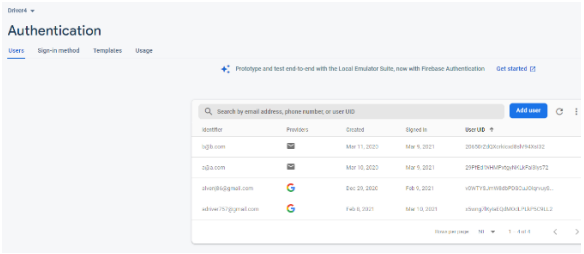
picture (DB4)



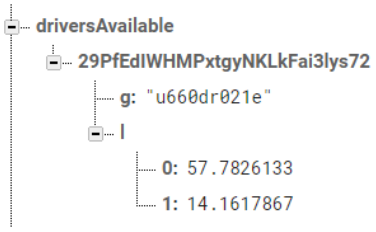
picture (DB5)



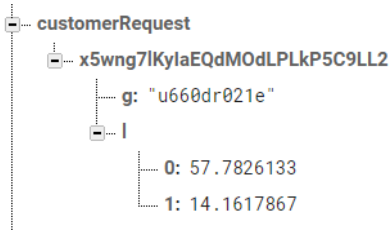
picture (DB6)



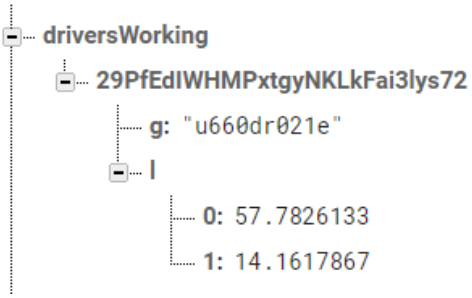
picture (DB7)



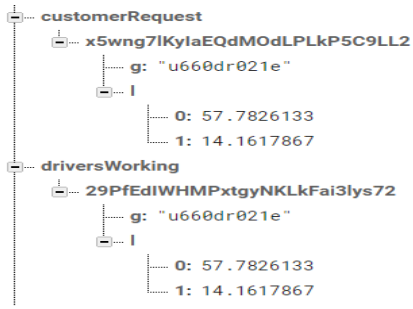
picture (DB8)



picture (DB9)

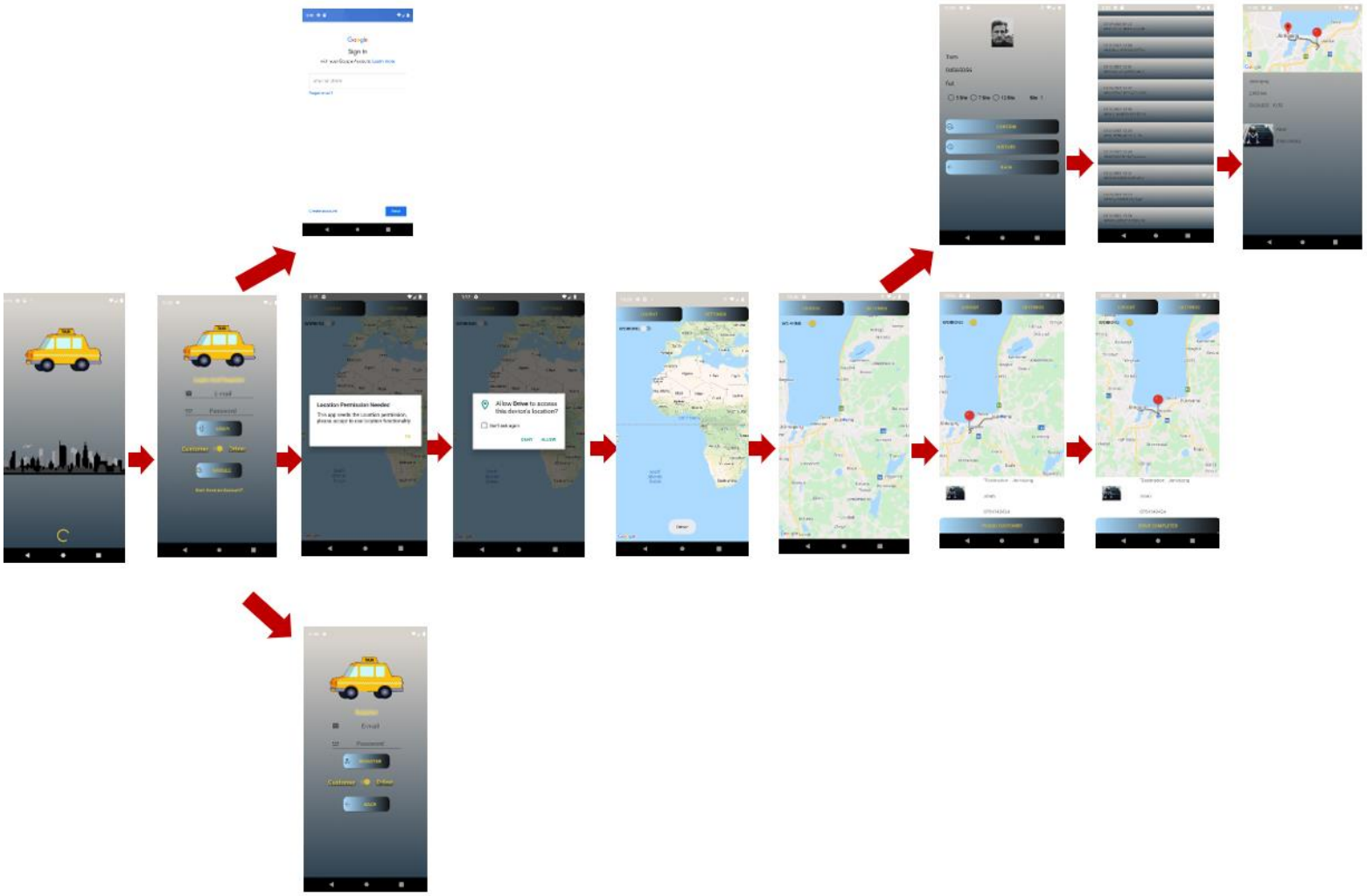


picture (DB10)



picture (DB11)

Driver screenshot



Customer screenshot