Car-search

Gabrijel Okorn
Project documentation of Car-search application

July 2023

1 Introduction

The purpose of this document is to describe how to use the application *Car-search* and the functionalities that are provided.
Application has a trivial purpose - helps user find his parked car. Once a user parked their car, all they have to do is click *lock* button on the main screen and the application will lead them to car until they click *unlock*.

2 Application Requirements

2.1 Operating System

Car-search is built in AndroidStudio and requires a device to have an Android version *Kitkat* (Android 4.4) or newer.

2.2 Location Service

- The device must have GPS based location capabilities to accurately determine the user's location.
- It should have location services enabled, and the app should be granted the necessary location permissions.

2.3 Sensors

- The device should have an accelerometer and magnetometer sensors to provide orientation data for the app.

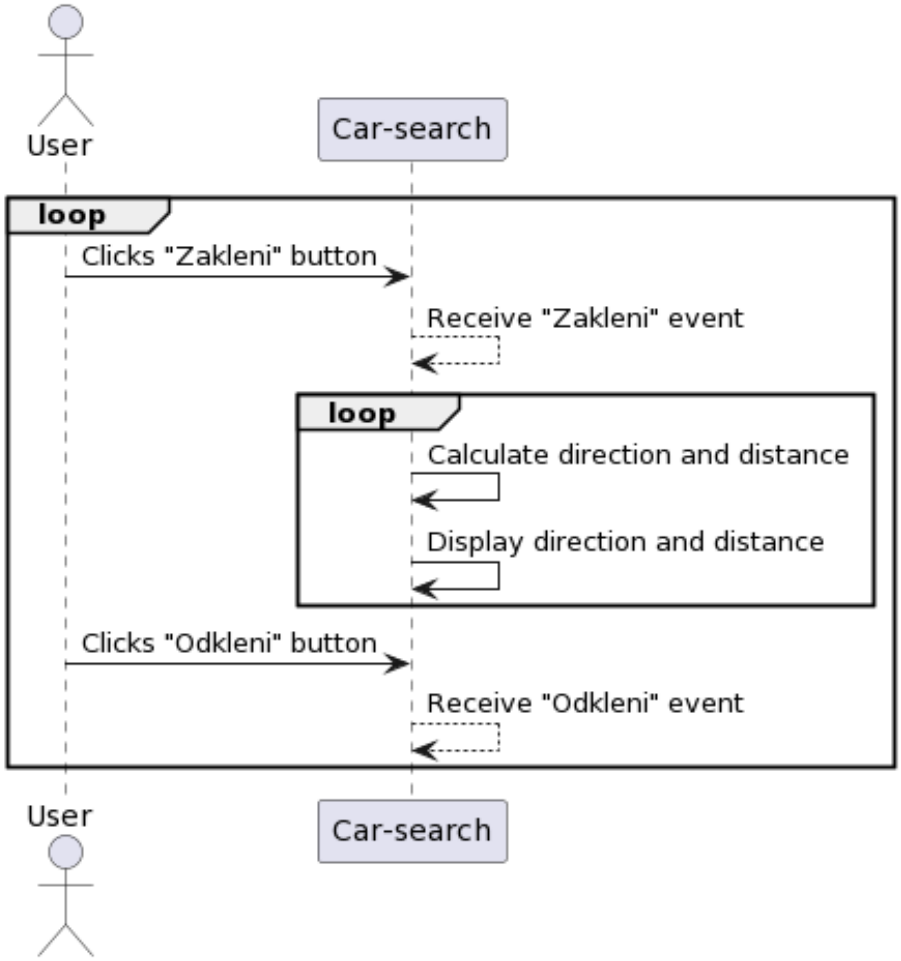
3 Application Workflow

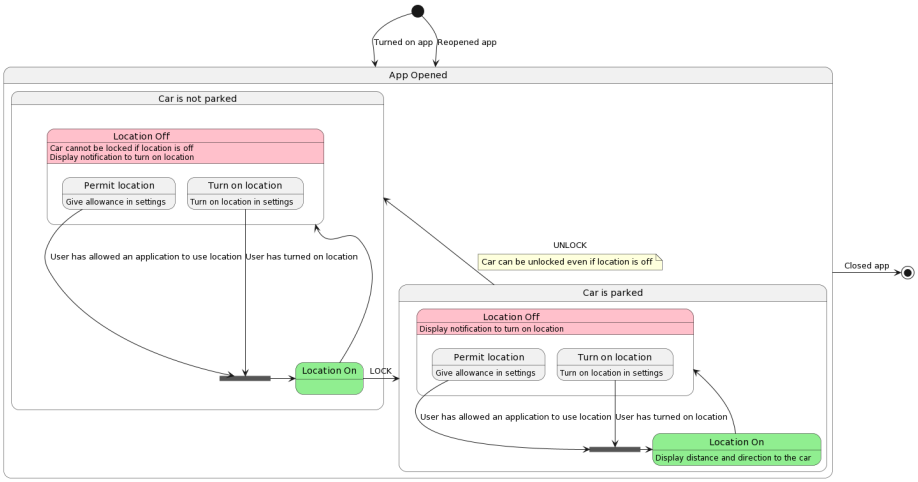
Program is written in java with it's main class *MainActivity.java*. It uses four other classes for it's reliable work.

- *Car.java* class effectively uses SharedPreferences to store and retrieve the car's state (locked or unlocked) and its location (latitude and longitude) persistently. It provides methods to lock and unlock the car, and it updates the UI to reflect the current state of the car using the provided LOCK/UNLOCK button.
- *Device.java* stores up-to-date newest information about user's device location.
- *Locationer.java* class is responsible for obtaining and handling locational information about the user's device. This class acts as a listener for location updates and responds appropriately when the location provider is enabled or disabled.
- *Orientationer.java* class is responsible for obtaining and handling orientation information of the user's device using the accelerometer and magnetometer sensors.
  The azimuth calculation (rotation around the Z-axis) is extracted from the orientation array and converted from radians to degrees.

SharedPreferences is a file of key-value data stored in android filesystem and can be found in */data/data/PACKAGE/shared-prefs/NAME-OF-PREFERENCES.xml*.

Additionally a state and sequence diagrams are provided for a better understanding of the program lifecycle.





4 Technical Detail

Both distance and angle of

4.1 Distance Calculation

In order to calculate distance from point A to point B we use longitude and latitude of those points.
Let's define ϕ as latitude and λ as longitude of points. By Haversine formula:

a = $\sin(\Delta(\phi\ /2)) + \cos(\phi_1) * \cos(\phi_2) * sin^2(\Delta\lambda/2)$
c = $2 * atan2(\sqrt{a}, \sqrt{1-a})$
distance = R * c,

where R is the Earth's radius (= 6371m).

4.2 Angle Calculation

Besides longitude and latitude, angular calculation from point A to point B also requires device's orientation. It is provided by devices magnetometer and accelometer. We use those to calculate the angle between A and B by the following formula:

ϕ = atan2(sin $\Delta\lambda$ * cosϕ_2 , cosϕ_1 * sin ϕ_2 - sinϕ_1 * cosϕ_2 * cos($\Delta\lambda$))

where ϕ_1,λ_1 is the start point, ϕ_2,λ_2 the end point $\Delta\lambda$ is the difference in longitude.

5 Conclusion

While programming the application I struggled most when trying to configure the location service on my android device. I was keep confronting the issue of app crashing, but I could not figure out what the cause for crash was. As this was my first Android-application and I have never been deep in Android operation system and it's general working I had no idea how to solve this problem. Soon enough I traced the error down and figured out that the app was crashing only when I changed location access of device while using the application. The library I was using of listening to location changes had a functionality to *throw error* when location provider was enabled or disabled. I solved the problem by overriding the functions to ignore the location service access as I expect users to turn on and off location service while using my application.
In addition I would add a functionality to application to provide a link to google-maps with a focus to their parked car.