



Bilkent University

Senior Design Project



Drector

Final Report

Shahriyar Mammadli, Nihad Azimli, Burak Özmen, Veysel Alperen Ceylan

Supervisor: Varol Akman

Jury Members: Çiğdem Gündüz Demir, İbrahim Körpeoğlu

Innovative Expert: Mustafa Sakalsız

Project Website: <https://drectorapp.github.io/Drector/>

Table of Contents

Introduction	4
Purpose of the system	4
Final Architecture and Design of the System	6
Package Diagram	6
Class Diagram	7
Impact of Engineering Solutions	8
Global Impact	8
Economic Impact	8
Environmental Impact	8
Societal Impact	8
Contemporary Issues	8
Engagement of Authorities on unnecessary reports	9
Misuse of chips	9
New Tools and Technologies Used	9
Ionic 3 Framework	9
Apache Cordova	9
Gradle	9
Git / Bitbucket / Sourcetree	10
Firebase Realtime Database	10
Google Cloud Functions	10
Nodemailer Module	10
Google Cloud Vision API	11
Visual Studio Code	11
Appendix: User Manual	12

Table of Figures

Figure 1. Package Diagram of <i>Drector</i>	6
Figure 2. Class Diagram of <i>Drector</i>	7
Figure 3. Login Screen	13
Figure 4. Signup Screen	14
Figure 5. Map Screen	15
Figure 6. Sliding Menu	16
Figure 7. Camera Mode of <i>Drector</i>	17
Figure 8. Drone Screen	18
Figure 9. Report Screen	19
Figure 10. Drone List Screen	20
Figure 11. Register Drone Screen	21
Figure 12. My Profile Screen	22

1. Introduction

Lately, drones and quadcopters started to get huge part of life in general. Almost in every industry people use drones or quadcopters for different purposes and usage of these flying objects is increasing day by day in vast level. However, there is no mechanism to get information about the objects flying around us. As stated above, drones can be used for multiple purposes; these purposes consist of both good and bad ones. Therefore, there is a demand for registering drones or keeping information about them. It is fact that governments want to learn specifications of unknown drones. Even more, anyone has right to learn about undefined flying object around him/her. Unfortunately, there is no way to get information about them. We plan to develop mobile application, *Drector* that will solve this worldwide issue. This cross-platform application will show in-demand information about drones and quadcopters. *Drector* is a mobile application which is implemented for both IOS and Android to keep track of drones and quadcopters that surrounds you. Users will be able to reach the information about drones or quadcopters using *Drector* which have two different modes. Users can inspect the drones or quadcopters that are nearby with map mode or camera mode of *Drector*.

1.1. Purpose of the system

Drector is planned to be global drone tracking service which makes available of getting real time information about drones all around the world. There are 3 different user types in the system which additional features are provided due to their role. Normal user, owner, and authorities are the user types. System has functionalities for the normal user that provides tracing drones all around the world with the help of map. User can click on the specific drone and see the information about both drone and owner. Furthermore, the drones, is in specific distance from the current location of user, can be selected and traced by the help of the map. That is, system provides users with the functionality that distance manually can be selected to trace the drones. It is much more useful in the manner of seeing that if there is alien or unknown drone around the user. For example, user can easily trace and get information about the strange drone in his or her garden. Furthermore, there is a camera mode which user can track the drone by the help of camera of the mobile phone. It is main usage is for the possibilities of having more than one drone that surrounds user and he or she wants to get and

information about the specific drone in certain direction. When drone is recognized in the camera, user can click and get information of the drone and the owner. Camera functionality is also a solution for the situation that user has more than one drone around which causes confusion when trying to recognize specific one in the map. Most important feature of the system is to provide users with functionality of the reporting the unknown and suspicious drone to the authorities. It is also purposive tool for the authorities to handle the security issue that are caused by the drones. Second user type is the owner which is provided with the same functionalities as normal user, however, additionally, has get license and added own drone to the system. Additional, UI is provided for the owner which by the help of this interface user it can be seen that if the drone is reported or not. However, data about the reported will not be given. Furthermore, when user flying the drone and area that drone is flid is problematic because of frequent reports, then, user immediately gets notification about it. Lastly, authorities can use the functionalities as above-mentioned roles, however, other than that; they can check reports to initiate the investigation about it. Authorities are able to reach the information about both reporter and reported person and their drones. By the help of system, security matter about the drones will be improved and issues that caused by drones will be minimized.

2. Final Architecture and Design of the System

2.1.Package Diagram

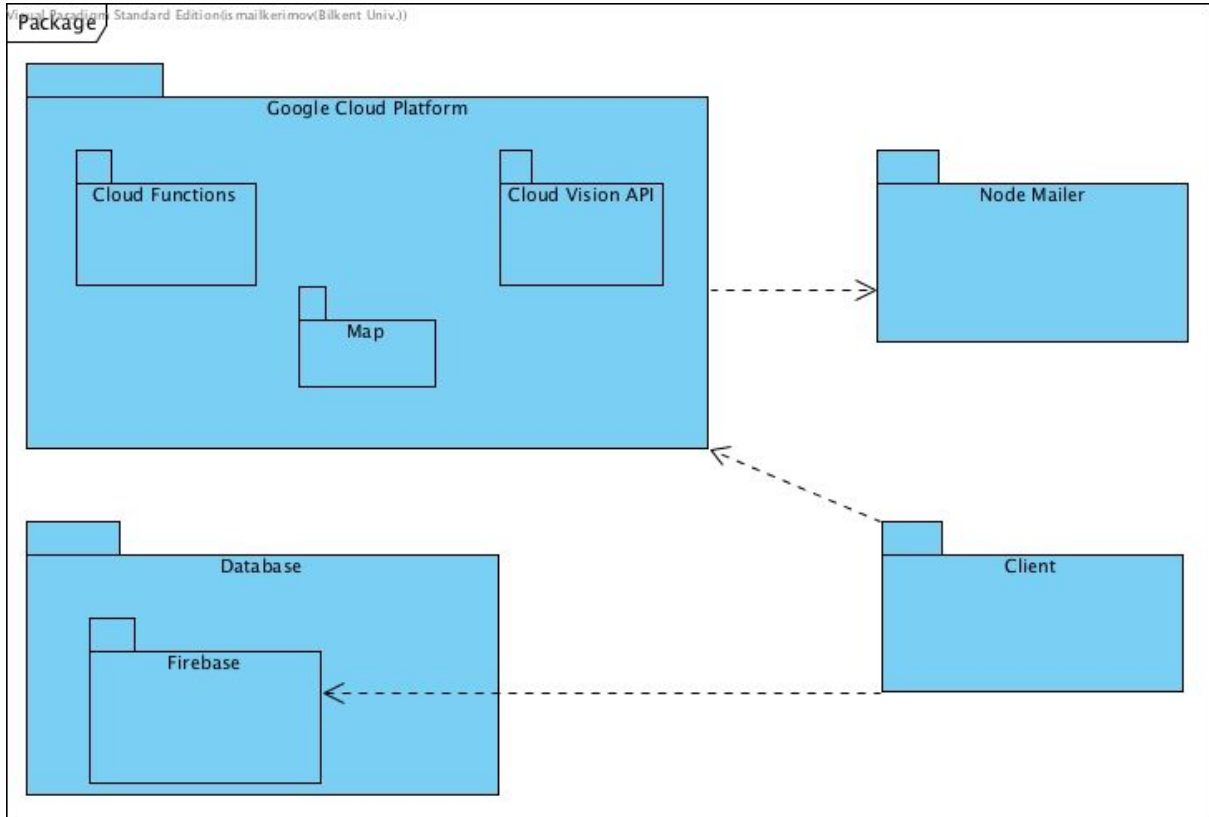


Figure 1. Package Diagram of *Director*

2.2. Class Diagram

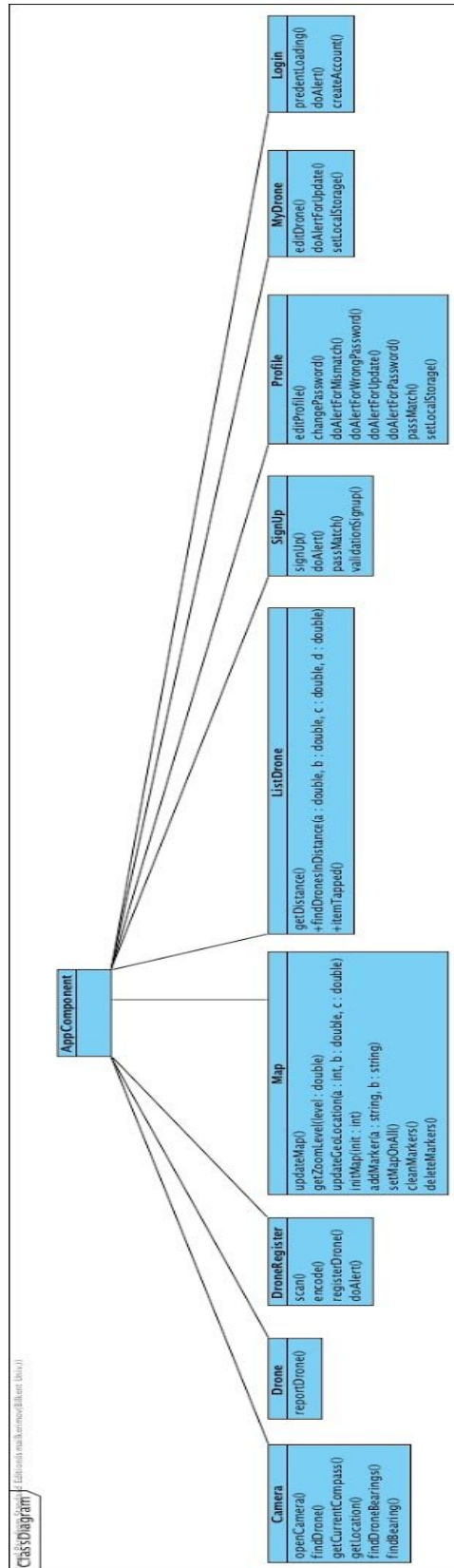


Figure 2. Class Diagram of *Drector*

3. Impact of Engineering Solutions

3.1. Global Impact

Our application mainly focuses on to become bridge between drone user and people around the world. *Drector* mainly focuses on to increase and improve security over drone related problems. Report and Detection features major proof of the this patterIt also helps to governments and secretly flying objects to ensure their anonimity.

3.2. Economic Impact

Economically *Drector* gives users no significant cost. Our plan is when someone goes to Directorate of General Civil Aviation (DGCA) for getting flight permit for their drone, they will get new chip (the chip probably will be Sim 908-C GSM + GPS module with IMEI registered) which will cost approximately 50 Turkish Lira. The staff from DGCA will integrate this chip to the drone. That is why, everyone with drone in Turkey may be traced where they fly their drone.

3.3. Environmental Impact

There is no environmental impact of *Drector*. Since our chips are chips that can be recycled and there is no need to change them constantly. Medium life span of our chips are 15 to 20 years.

3.4. Societal Impact

Drector app is designed to minimize drones' both intentionally and unintentionally infringe of people's privacy. Moreover, *Drector* can be used in military level. It is intended to improve security issues that caused by drones. Also, abuse of drones will be diminished significantly.

4. Contemporary Issues

Drector is not designed to cover all the users of smart phones. People who owns drone must use it. Also, people who are engaged with drones or interested in drone tecnology and the ones who gets offended by drones mostly will use it. However, following issues may occur according to misuse of the application.

4.1. Engagement of Authorities on unnecessary reports

Some people may send reports for just fun, or to get revenge from someone who has another problem with someone. Authorities may get engaged on wrong reports or cases that will waste their times. Also, after some cases of misuse authorities believe on application will diminish and they will not take serious of the cases that reported by *Drector* users.

4.2. Misuse of chips

Some fraudulent people may misuse the chips for tracking people constantly to track location of someone.

5. New Tools and Technologies Used

5.1. Ionic 3 Framework

Ionic 3 framework is a platform to build hybrid mobile applications for iOS, Android, and Windows Phones. Hybrid applications are build for cross platform using web tools so that we can develop the application once and deploy to multiple platforms without having to change the code base. We chose Ionic 3 Framework in order to gain its benefits such as shorter development, less complexity, and single code base. Even though there are speed differences between native mobile applications and hybrid mobile applications, Ionic 3 Framework has been optimized to minimize the speed difference.

5.2. Apache Cordova

Apache Cordova is an open-source mobile development framework. Cordova wraps our HTML/JavaScript app into a native container which can access the device functions of several platforms.

5.3. Gradle

Gradle is an open-source build automation tool focused on flexibility and performance. Gradle is used as build tool in this project.

5.4. Git / Bitbucket / Sourcetree

Bitbucket is a web-based version control repository hosting service owned by Atlassian, for source code and development projects. We use Bitbucket hosting service as our version control project repository. Sourcetree which is also owned by Atlassian, simplifies how we interact with our Git repositories so we can focus on coding. We use Sourcetree as our Git GUI since it visualizes and manages our Bitbucket repository through its simple Git GUI.

5.5. Firebase Realtime Database

Firebase Realtime Database is a cloud-hosted NoSQL database. It stores data as JSON and synchronizes data in realtime across all clients. Also, data remains available even if the app goes offline. Firebase Realtime Database is used in this project to store all our data including user informations, drone informations and locations, and reports.

5.6. Google Cloud Functions

Cloud Functions for Firebase lets us run our backend code that automatically responds to events triggered by Firebase features and HTTPS requests. The code is stored in Google's cloud, and runs in a managed environment. Therefore, there is no need to manage and scale our own servers. Realtime Database Trigger of Cloud Functions is used in this project. The function is triggered when a user tries to register a drone to the system. Then, the function uses Nodemailer Module which creates transport mechanism such as SMTP to send e-mails to the user.

5.7. Nodemailer Module

Nodemailer is a module for Node.js applications to allow easy email sending. Nodemailer is a single module with zero dependencies and code is easily auditable, as there are no dark corners. Also, it is heavily focused on security. Because of these features, Nodemailer module is used in this project in order to send e-mails to the users who wants to register their drones.

5.8. Google Cloud Vision API

Cloud Vision API provides powerful Image Analytics capabilities as easy to use APIs. It enables application developers to build the next generation of applications that can see and understand the content within the images. The service enables us to detect a broad set of entities within an image. Cloud Vision API's label detection feature is used in this project in order to recognize the drone that user wants to see its information via camera mode of *Drector*.

5.9. Visual Studio Code

Visual Studio Code, which is a source code editor developed by Microsoft for Windows, Linux and macOS, is used as a source code editor in this project. It includes support for debugging, embedded Git control, syntax highlighting, intelligent code completion, snippets, and code refactoring. In addition to these features, integrated terminal is the most important reason why we chose Visual Studio Code as a source code editor since terminal is frequently used while developing applications with Ionic Framework.

Appendix: User Manual

Appendix A: User Manual for Android Mobile Version

A.1 Installation of *Drector* on Android

Drector can be installed on Android using apk file. Users will be able to download the apk file which will be on the website of *Drector*.

A.2 Login Screen of *Drector* on Android

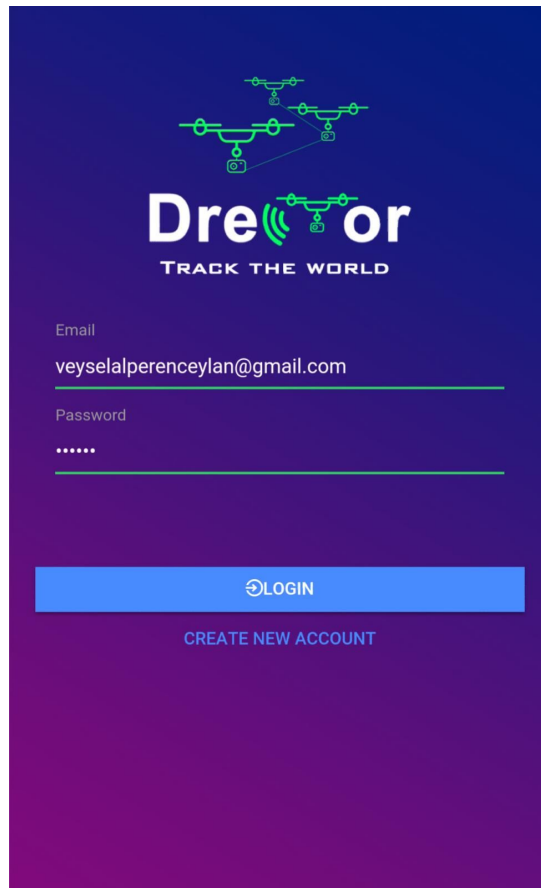
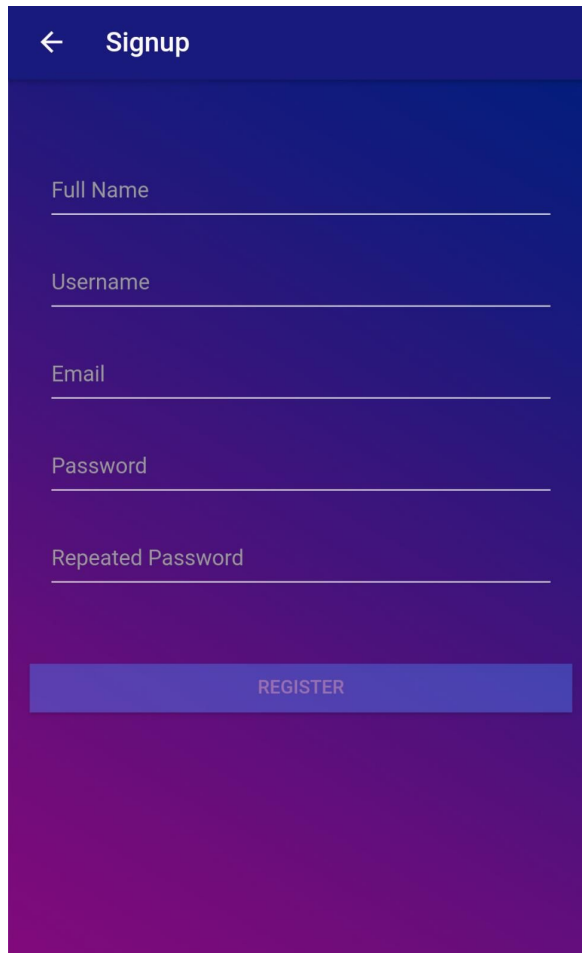


Figure 3. Login Screen

When user opens *Drector*, Login screen will be opened automatically as first page of the application. User can login to the system using his e-mail and password which are registered to the system. If user is not registered to the system, he can go to register page using “CREATE NEW ACCOUNT” button.

A.3 Sign Up Screen of *Drector* on Android



The image shows a mobile application interface for a signup screen. The background is a dark blue gradient. At the top, there is a header bar with a white back arrow on the left and the word 'Signup' in white text. Below the header, there are five text input fields, each with a light blue label and a white underline. The labels are 'Full Name', 'Username', 'Email', 'Password', and 'Repeated Password'. At the bottom of the form, there is a solid blue button with the word 'REGISTER' in white capital letters.

Figure 4. Signup Screen

On the Signup page, one should fill all necessary information which are Full Name, Username, Email, Password and Repeated Password. After filling the required fields, registration process can be completed using REGISTER button.

A.4 Map Screen of *Drector* on Android

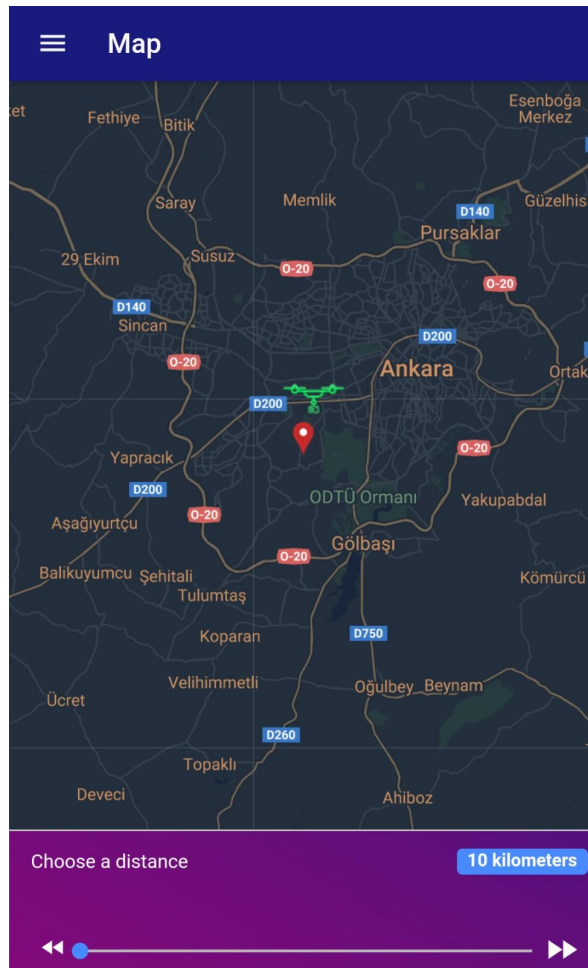


Figure 5. Map Screen

After new user registers to the system or existing user logins to the system they will be directed to the Map Screen. In this screen user can see his location and inspect the drones near him. User can click any drone icon flying around and get the necessary information about it such as brand name, owner name, flying purpose and so on in Drone Screen. By sliding to the center from left of the screen user can go to other pages with sliding menu.

A.5 Sliding Menu of *Drector* on Android

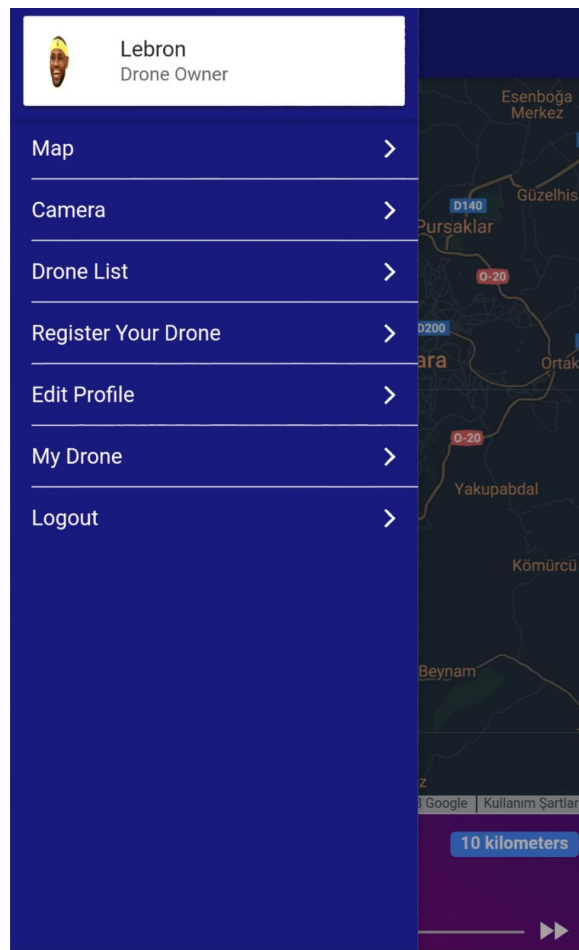


Figure 6. Sliding Menu

This menu will help you to navigate through the application. User can choose one of the following options, such as, Map, Camera, Drone List, Register Your Drone, Edit Profile, My Drone, Logout. Map option is described above, however, other options that is represented in the sliding menu will be discussed in the next couple of sections. Lastly, user can reach his profile from top of the menu where his profile picture and username is represented.

A.6 Camera Mode of *Drector* on Android

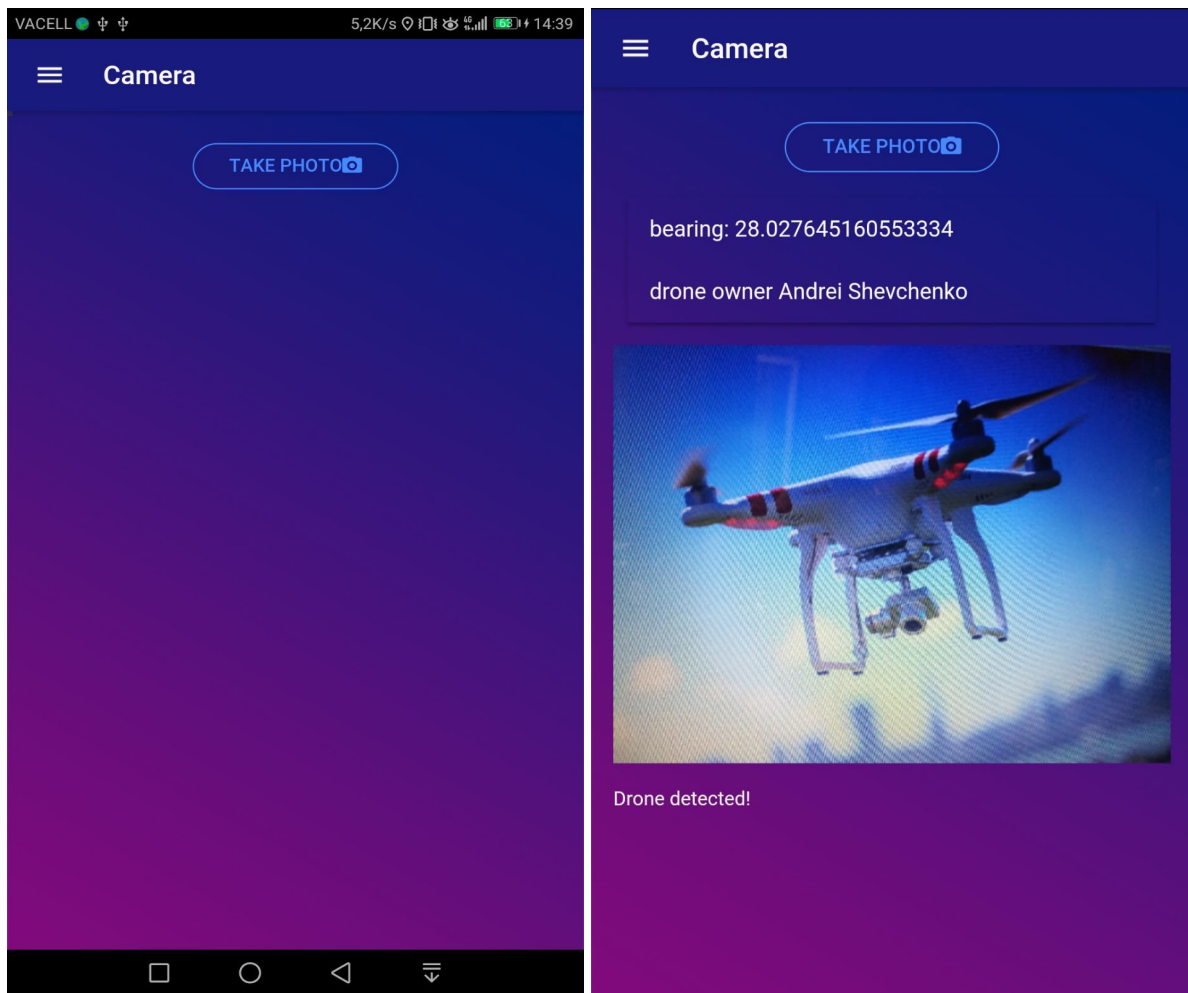


Figure 7. Camera Mode of *Drector*

This screen can be opened from sliding menu by clicking to the camera, after camera mode opens up their camera automatically opens. If user canceled camera, can click to the *take photo* to get their camera reopen. The illustration of it depicted in the left photo. After they take photo screen changes to the right one and user can see the photo the he took and necessary information about it. Also user can click information card and get the necessary information about it such as brand name, owner name, flying purpose and so on in Drone Screen. If drone not detected user will see the message as *Drone* is not detected.

A.7 Drone Screen of *Drector* on Android

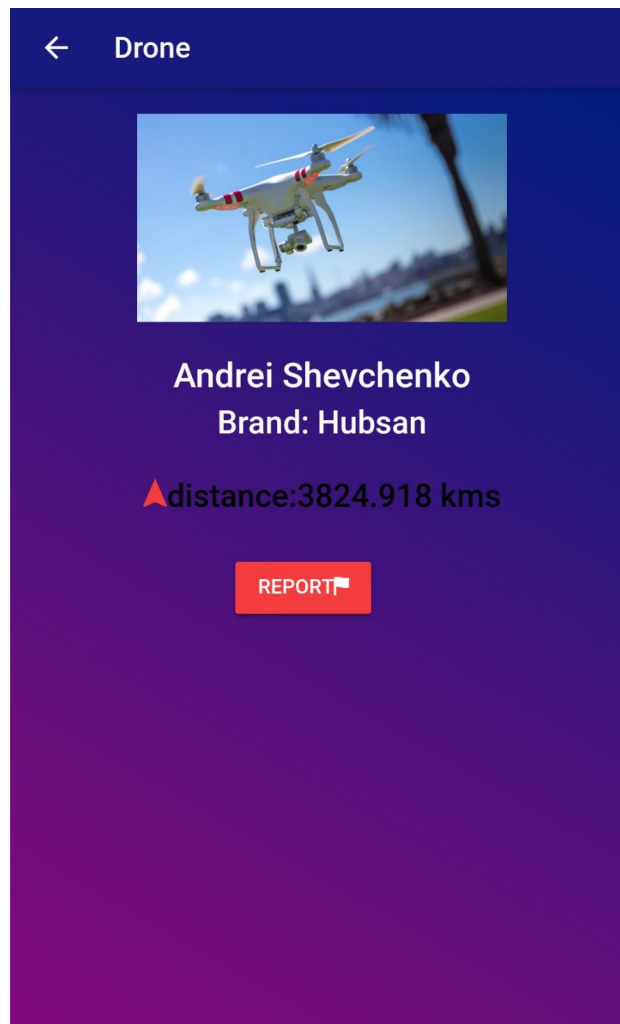


Figure 8. Drone Screen

If user from the previous page clicks the name he saw in the screen after drone is detected or user clicks the drone icon in map or click one of the drone in Drone List Screen, he is directed to this screen where user can see information about specific drone owner. If user wants to report the specific drone owner, he can click to saw and can report the drone owner the report part depicted more detailed way in following steps.

A.8 Report Screen of *Drector* on Android

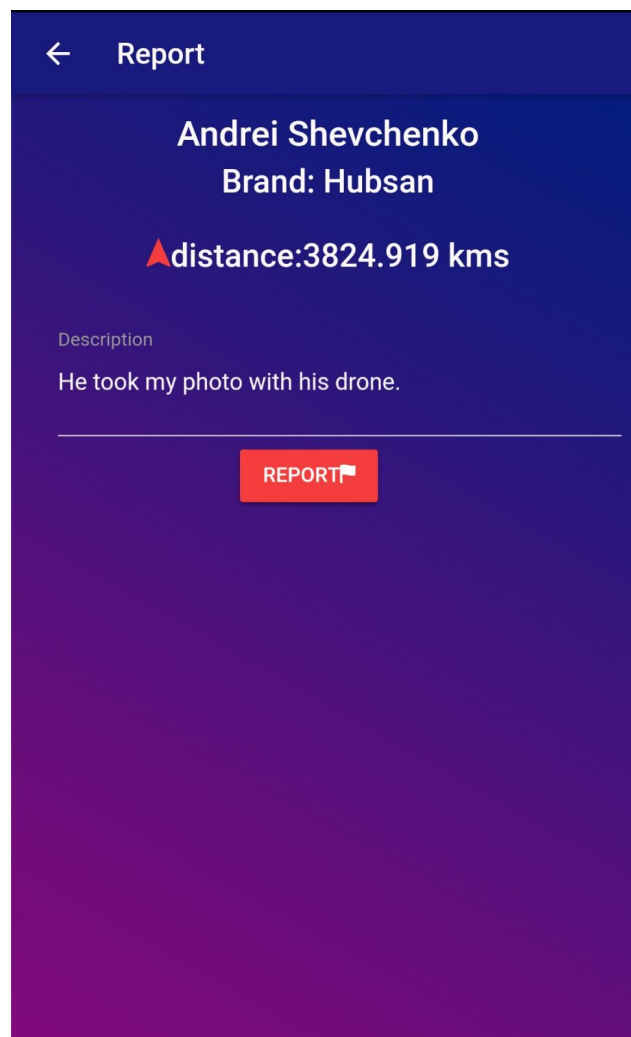


Figure 9. Report Screen

In this page user can make report about flying drone username and brand field automatically filled by the program. User just enters the message to report the drone to authorities about the non appropriate behavior done by drone owner. After report is completed user will be directed to Home Page which is Map mode.

A.9 Drone List Screen of *Drector* on Android

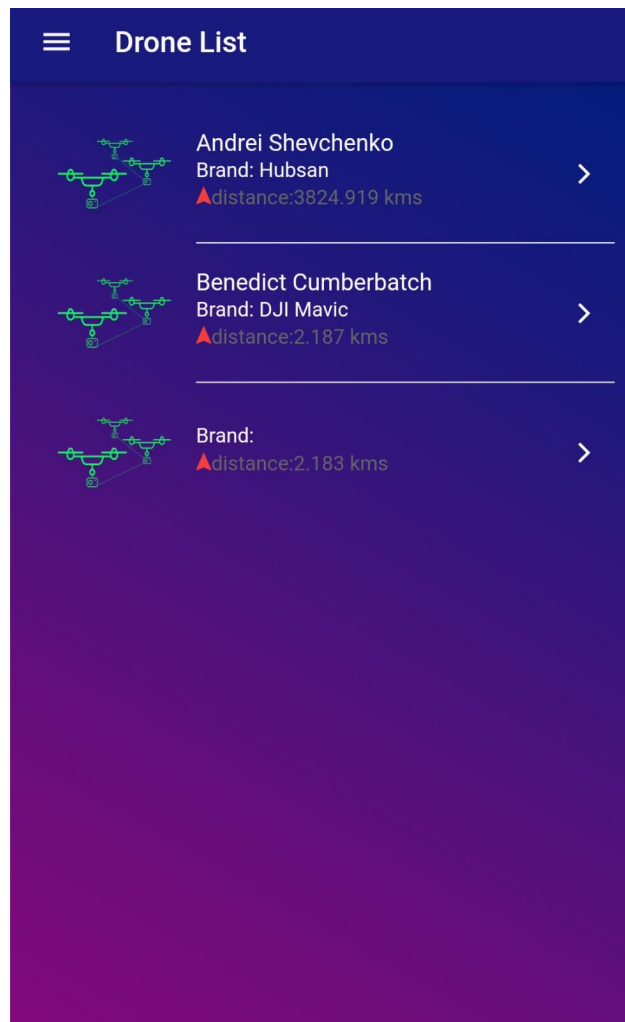


Figure 10. Drone List Screen

Drone list is also the place that user can navigate from Sliding menu. User can see the nearby drones from this page and inspect their picture, information and so on.

A.10 Register Drone Screen of *Drector* on Android

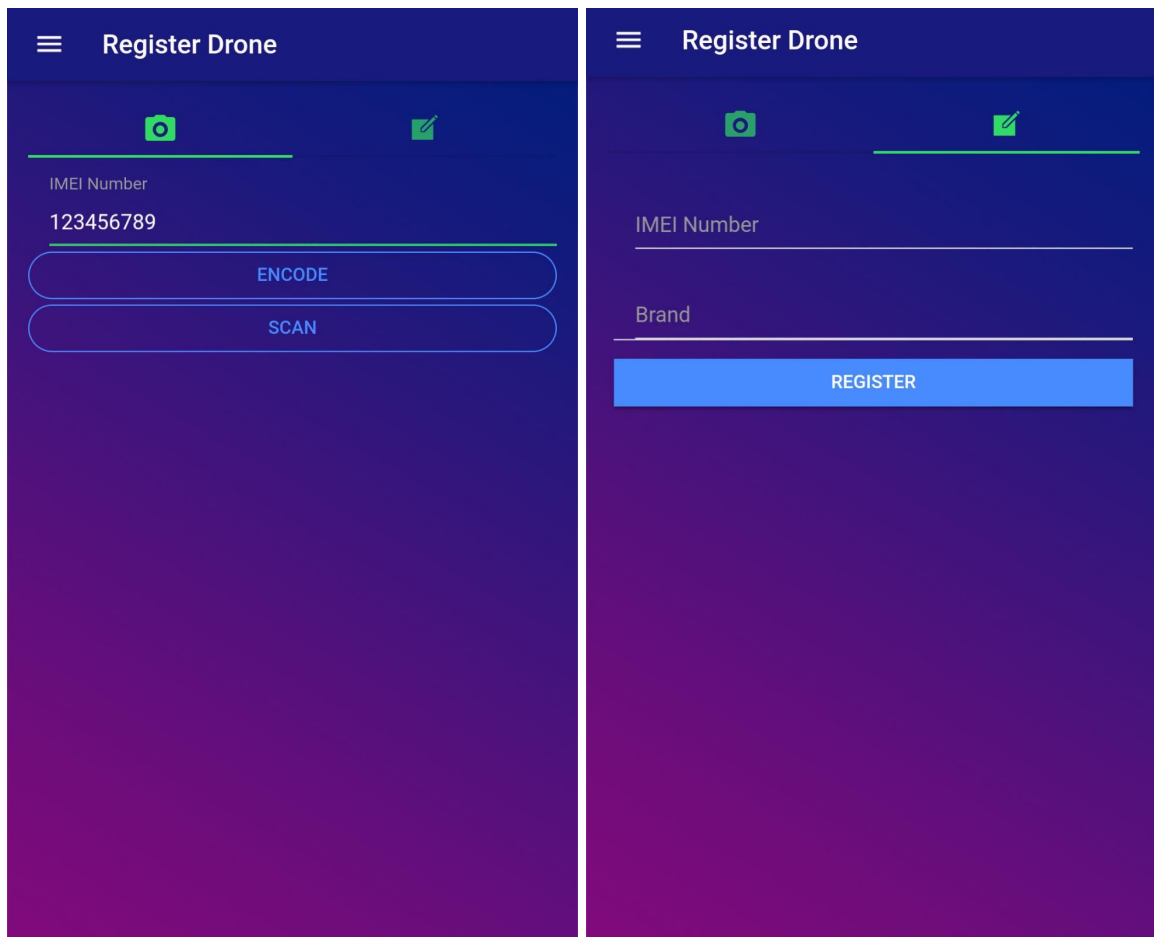


Figure 11. Register Drone Screen

Register Drone Screen is also the place that user can navigate from Sliding menu. User can register his/her drone with Register Drone Screen in two ways. One way of the register drone to system is filling the inputs that *IMEI Number* and *Brand* area, the other is that every drone has barcode that can be encoded from their IMEI number. User can scan the drone's barcode with barcode scanner by clicking the *SCAN* button of the first picture.

A.11 My Profile Screen of *Drector* on Android

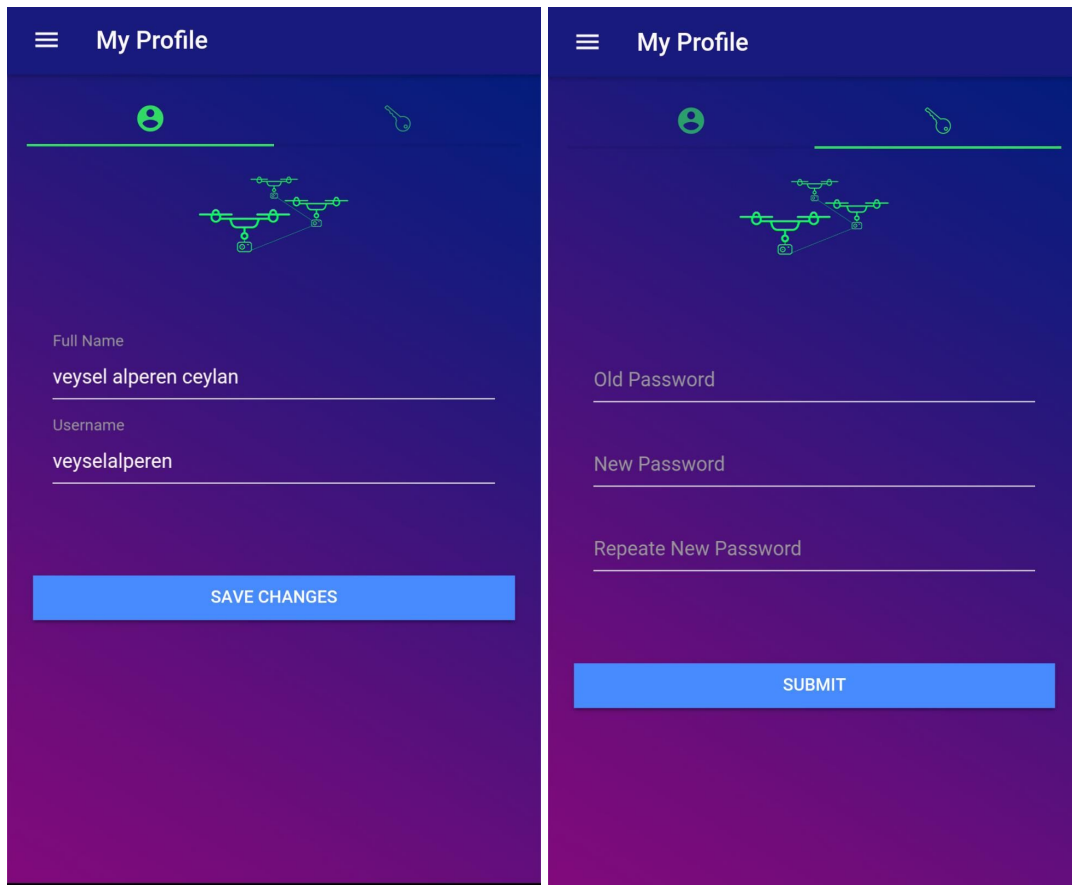


Figure 12. My Profile Screen

User can reach this page using sliding menu. This page is where user can edit his profile such as his Full Name, and Username. Also, user can change his password using this page.