Bilkent University

Department of Computer

Senior Design Project

DRector

Analysis Report

Nihad Azimli, Shahriyar Mammadli, 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

Table of Figures

1. Introduction

Lately drones and quad copters started to get huge part of life in general. Almost in every industry people use drones or quad copters 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 a 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 will be implemented for both iOS and Android to keep track of drones and quad copters 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 is nearby with map mode or camera mode of *Drector*.

2. Proposed System

Drector is a mobile application to keep track of drones and quad copters that surrounds you. The aim of application is reaching the information about drone or quad copter. Firstly, when user launch the application, user will be asked to login to the system in order to use *Drector.* After the user authenticates himself, user will encounter map where he/she can see the drones or quadcopters that is nearby. By clicking the little icon of each drone user can reach information about that drone. This information consists of different things, such as name of drone, name of owner, cause of flight, specifications of drone (this information is optional owner of drone may choose to hide or show some of them). Also, user can view drones around him/her and get information instantly by camera of the phone. Such that, user can convert application to camera mode instead of map mode. In camera mode user can direct his/her camera to the drone and after this application detects drone that is flying, user can inspect drone in detailed way.

3. Overview

In this part of the report, system will be further discussed according to the functional requirements, non-functional requirements and pseudo requirements. Also the system model will be analyzed that include the scenarios, use case model, object and class model, dynamic model and user interface

3.1. Functional Requirements

- Drector will allow users to create their profiles.
- Users will be required to authenticate themselves before starting to use Drector.
- Drone owners must register their drones to Drector.
- Drone owners may hide some of their information from public.
- Users will be able to use Drector in two different modes which are map and camera modes.
- Users will be able to reach the information about the drone that is nearby by clicking its icon from map in Drector’s map mode.
- Users will be able to reach the information about the drone after Drector detecting it by camera in Drector’s camera mode.
- Drector will keep information of many drones in database to display it to users.
- Users will be able to send a report with a message to alert the authorities.

3.2. Non-Functional Requirements

- Drector will be implemented for both iOS and Android.
- Drector will not work without internet connection.
- Drector will have a user friendly graphical user interface and will be very easy to use.
- Drector will be reachable 7/24.
- Drector will keep user information safely in database and they will not be reachable by other users.
- Drector will have a low response time to display the information after detecting the drone.

3.3. Pseudo Requirements (Constraints)

3.3.1. Economical Constraints

The application will be on both App Store and Google Play Market for no price. Some amount of money will be paid to publish this application on both markets. Indeed, we will not pay for any development of software; we will use open-source communities and environments. Cost of GPS chips will be covered by companies. We will provide GSM module reader to get locations of drones.

3.3.2. Language Constraints

Our application will be released in English and Turkish for now; but according to demand of the application, other languages may be added.

3.3.3. Technological Constraints

Our application will be compatible with any device such as tablets, smart phones that has Android Operating System or iOS. Also, application needs Internet connection to run properly.

3.3.4. Ethical Constraints

As mentioned above, some special information will be private according to the choice of user and the owner of drones. However, some information cannot be private as we aim to publicize undefined flying objects.

3.3.5. Timing Constraints

The application will be fully implemented and tested by the end of Spring 2018 semester.

3.3.6. Social Constraints

Some people tend to hide their actions through drones. However, these groups of people are mainly the ones whose purpose is wicked actions. Indeed, our objective is limiting this kind of actions.

3.3.7. Implementation Constraints

To manage control system while in the development, Bitbucket and SourceTree will be used. The implementation will be conducted in Angular 4. System will be available for mobile devices with Android Operating Systems (2.2.3 or higher version) and iOS (9 or higher version). MySQL will be used for database manipulation purposes.

3.4. System Models

3.4.1. Scenarios

- Signup for new user

User opens the application. Clicks the signup button, and then enters the required information such as first name, last name, e-mail address (which will be verified through pass code) and password. Also, user fills some optional parts, namely, phone number, and postal code. Then, she clicks next where she asked whether she has drone or not. She chooses no for now option. Lastly, she gets verification e-mail and enters the code she got from server and becomes confirmed user.

- Authentication Error

User opens the application and enters required information to login; however, inputted data has inaccuracy. System responses with authentication error by alerting that “The username and password that you entered do not match with our records”.

- Tracing circumjacent drones

User wants to check if there are drones around his home. After entering to the application and passing authentication process, in homepage, map appears with the information of the surrounding drones. User clicks icon of specific drone and reads information of the drone.

- Get info about specific circumjacent drone

User opens the application and automatically logins to the system with remember me feature. Then, user encounters with map page where she can get details of drones flying nearby. She switches the program to camera mode. She clicks the drone and gets specifications of the drone accordingly.

- Reporting drone to the authorities

User sees drone that is in garden without user’s permission and opens application. System automatically accepts user as logged in user because user has used application before and clicked to remember me checkbox. User clicks button that directs to the choosing specific drone from the camera. User tracks drone and gets information of the drone, and then reports it to the authorities by clicking report and typing report details.

- Add certified drone to system

User enters the application and automatically logins to system by remember me feature. Then clicks profile icon on the top of screen to add certification for her drone. In her profile page user clicks the Add Drone. User enters the certificate number.

3.4.2. Use-Case Model

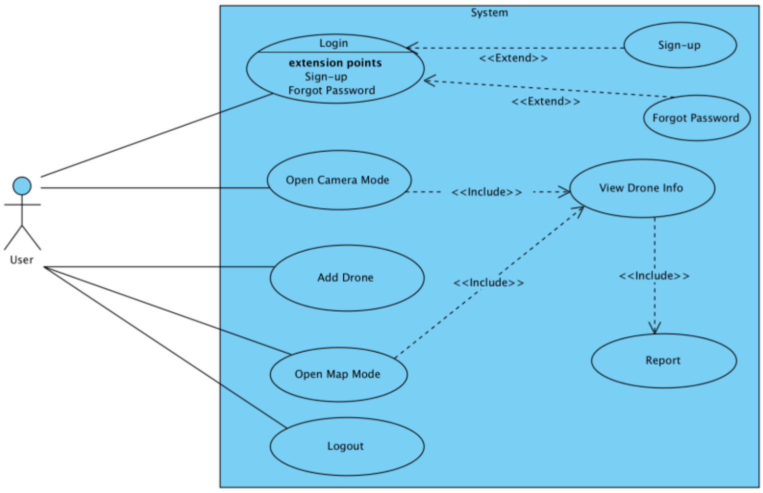


Figure 1 **Use Case Diagram**

3.4.3. Object and Class Model

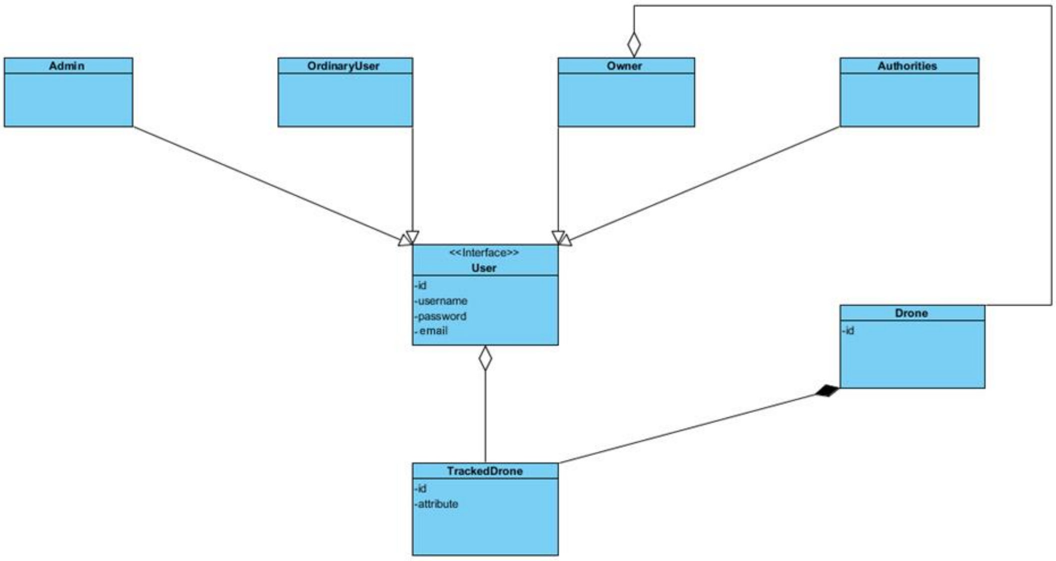


Figure 2 **Class Diagram**

3.4.4. Dynamic Models

3.4.4.1. Sequence Diagram

Scenario: User Selim detects flying object in the garden part of his house so he tries to detect this flying object. Selim detects the drone and he gets legal information about the owner of the drone, he sees that he has no legal reason to fly around his garden so he wants to report the drone. First he takes picture of drone, and then reports it along with picture to whom it may concern.

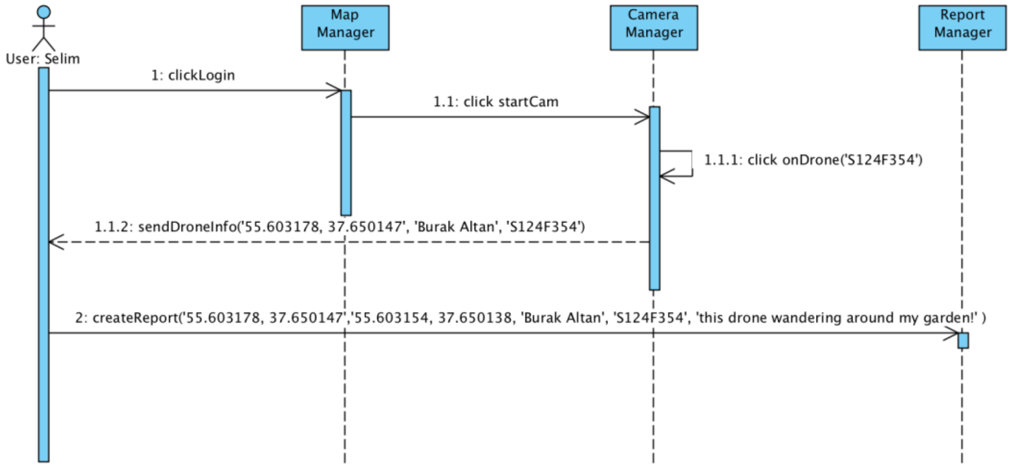

User: Selim
Map Manager
Camera Manager
Report Manager
1: clickLogin
1.1: click startCam
1.1.1: click onDrone('S124F354')
1.1.2: sendDroneInfo('55.603178, 37.650147', 'Burak Altan', 'S124F354')
2: createReport('55.603178, 37.650147','55.603154, 37.650138, 'Burak Altan', 'S124F354', 'this drone wandering around my garden!')

Figure 3 **Sequence Diagram**

3.4.4.2. Deployment Diagram

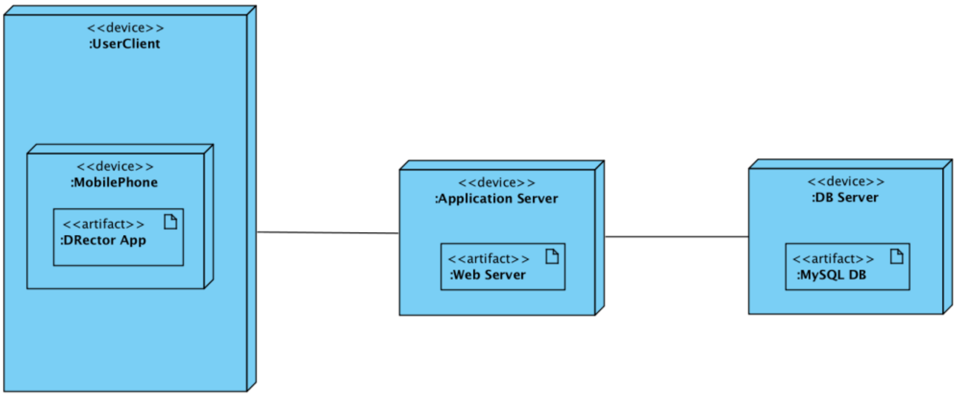


Figure 4 **Deployment Diagram**

3.4.5. User Interface

Drector’s user interface is easy to use.

3.4.5.1. Screen Mockups

3.4.5.1.1. Starting Page

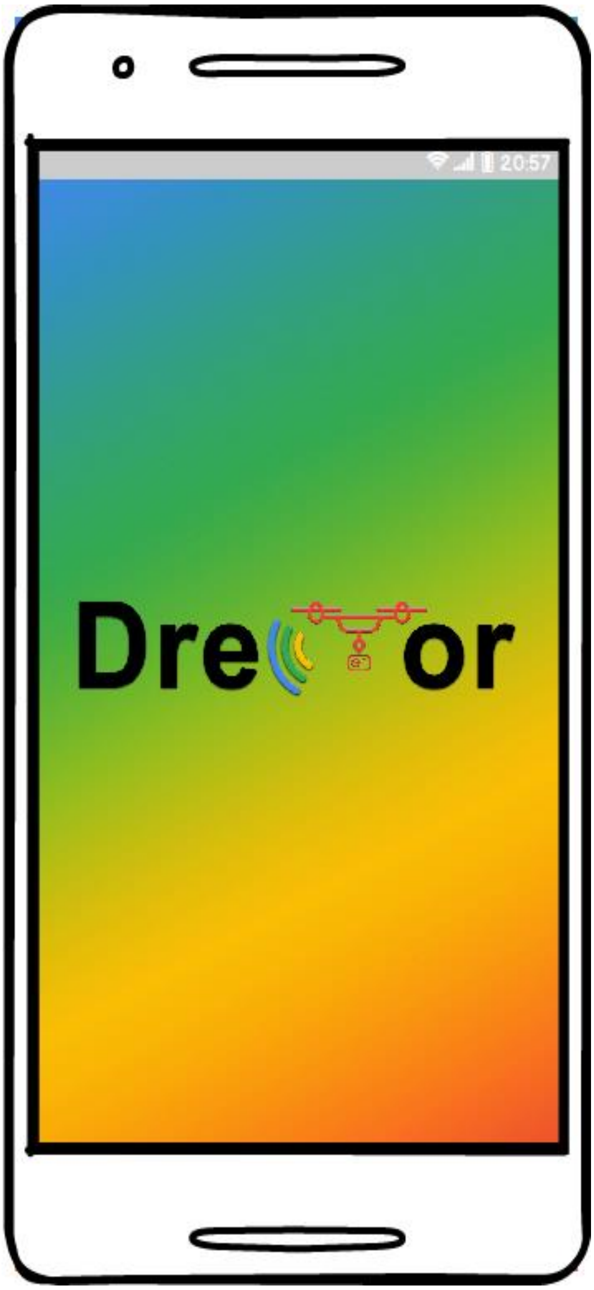


Figure 5 **Starting Page Screen Mockup**

In this part, the application logo is showed only a few seconds. Then, login page is opened.

3.4.5.1.2. Login Page

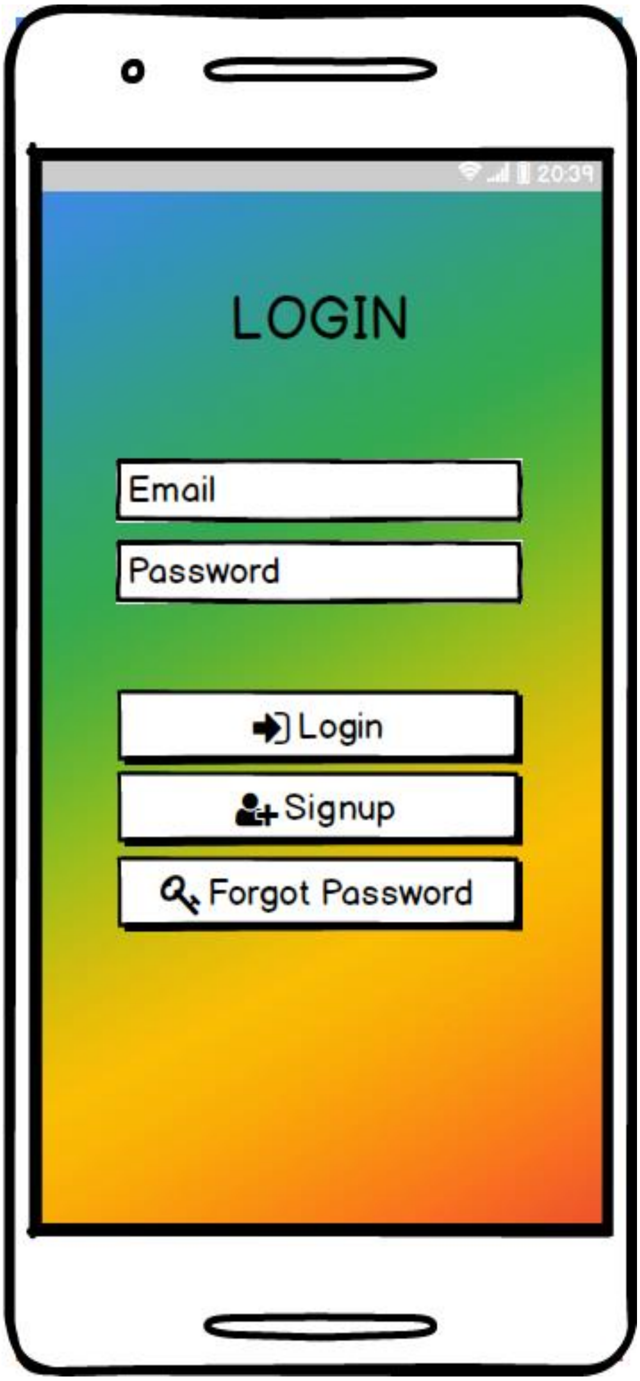


Figure 6 **Login Page Screen Mockup**

In this part user can login with username or email and password or if user does not have an account s/he goes for sign-up screen by clicking signup button. Also users have a chance to reset password by clicking Forgot Password button to go Forgot password screen.

3.4.5.1.3. Signup Page

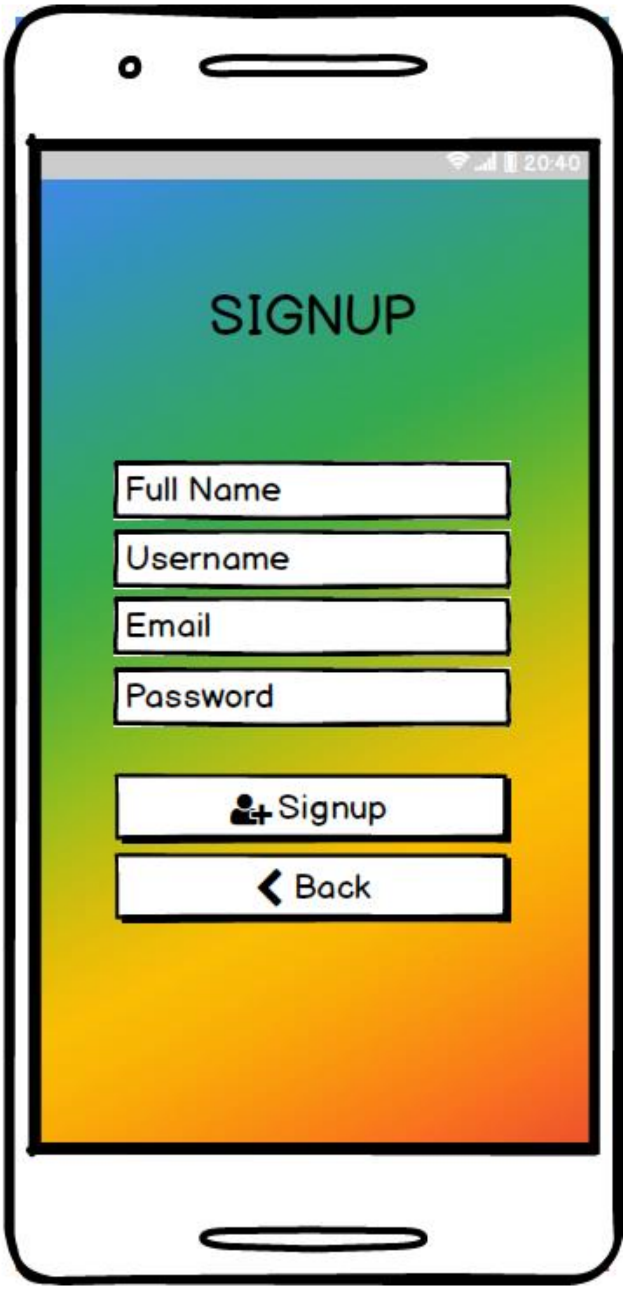


Figure 7 **Signup Page Screen Mockup**

User can sign-up with a full name, proper email address, user name and proper password in this screen. Back button is for going back to Login Page.

3.4.5.1.4. Forgot Password Page

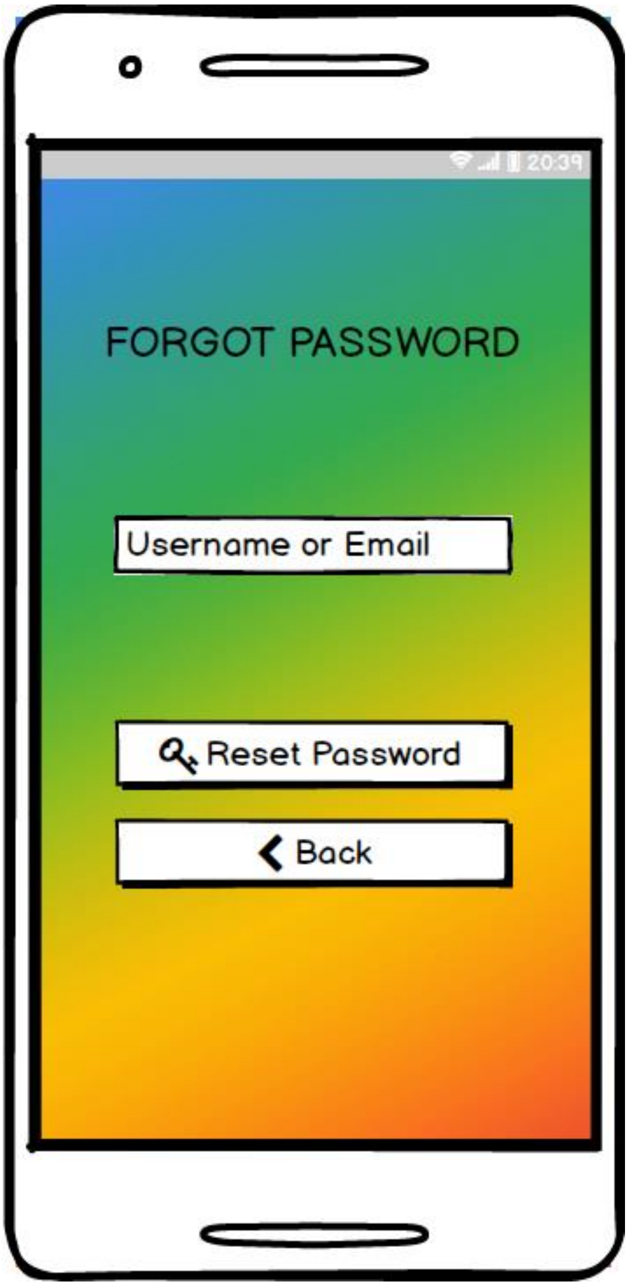


Figure 8 **Forgot Password Page Screen Mockup**

If users forgot password, can reset their password with their email address or username. Back button is for going back to Login Page.

3.4.5.1.5. Map View Page

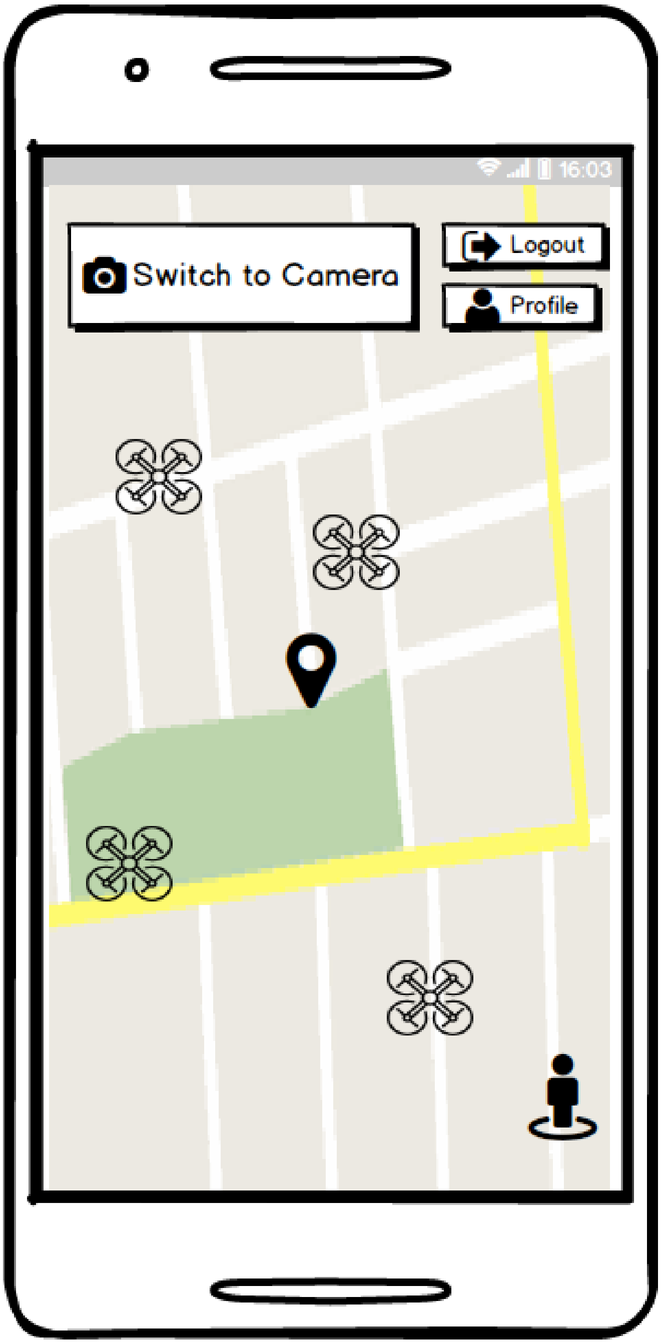


Figure 9 **Map View Page Screen Mockup**

Users can see the drones which fly near them in the map. Also they can change to the camera view by clicking Switch to Camera button. The logout button is for going back to login page. The profile button is for going to profile page to change the profile information and add drones.

3.4.5.1.6. Camera View Page

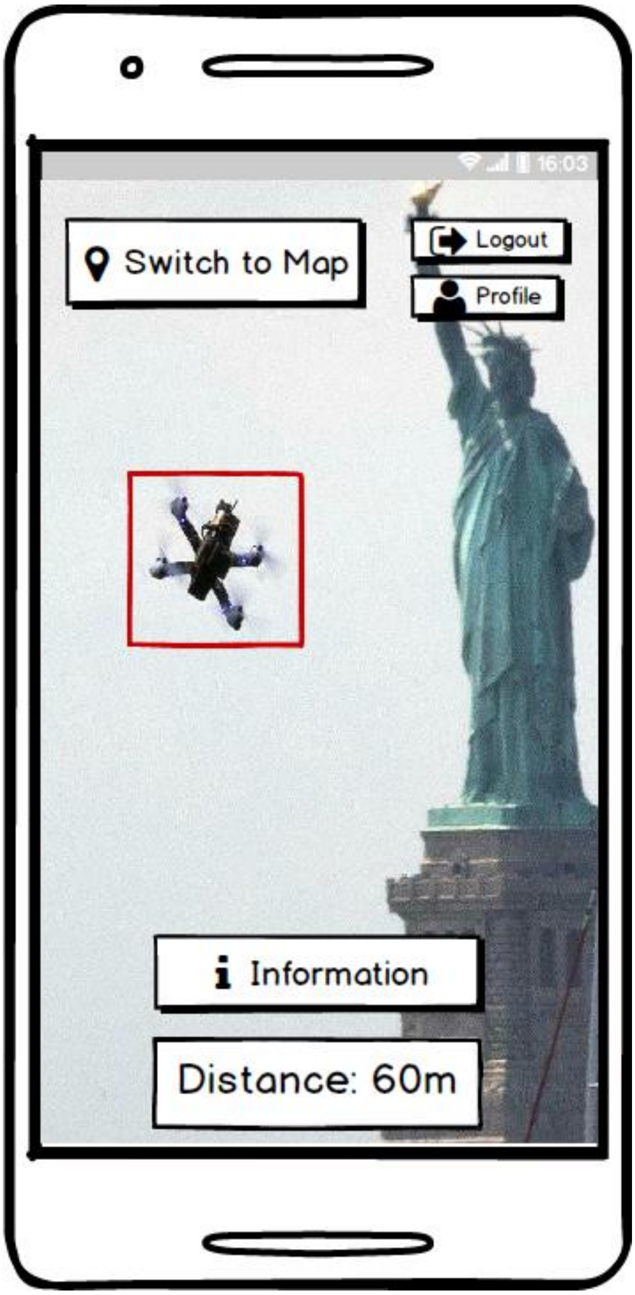
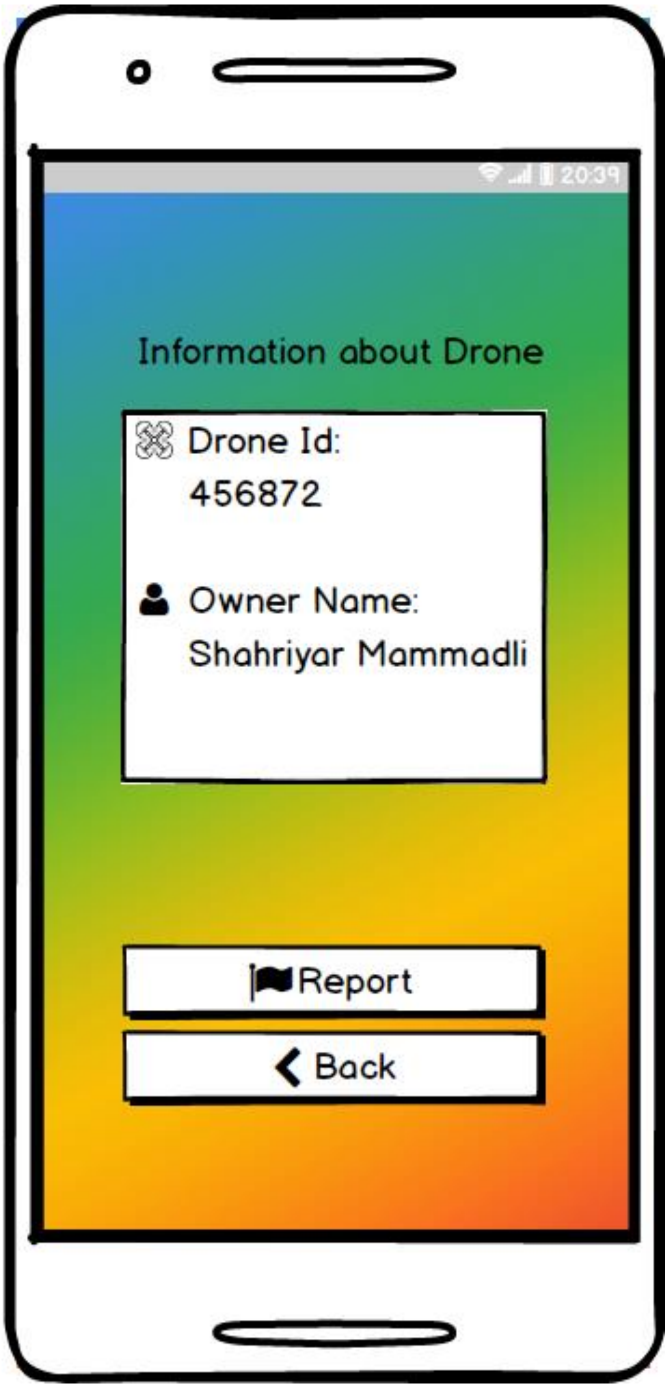


Figure 10 Camera View Page Screen Mockup

Users can detect the drones with camera view and see the information about drones by clicking information button. Also users can change to the map view by clicking switch to map button. The logout button is for going back to login page. The profile button is for going to profile page to change the profile information and add drones.

3.4.5.1.7. Information Page



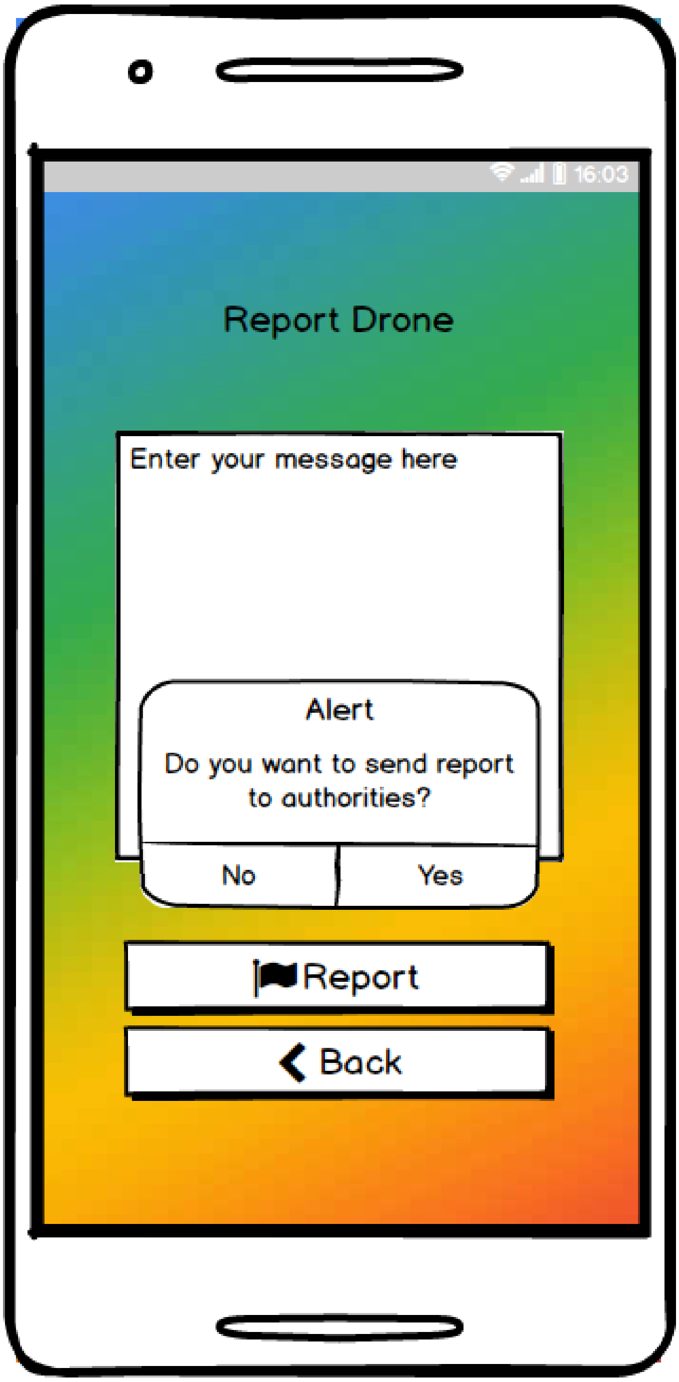
Drone Id:
456872

Owner Name:
Shahriyar Mammadli

Figure 11 **Information Page Screen Mockup**

Users can see the information about the selected drone. And they can report to the officers by clicking Report button by going report page. Back button is for going back to view pages.

3.4.5.1.8. Report Page



Report Drone

Enter your message here

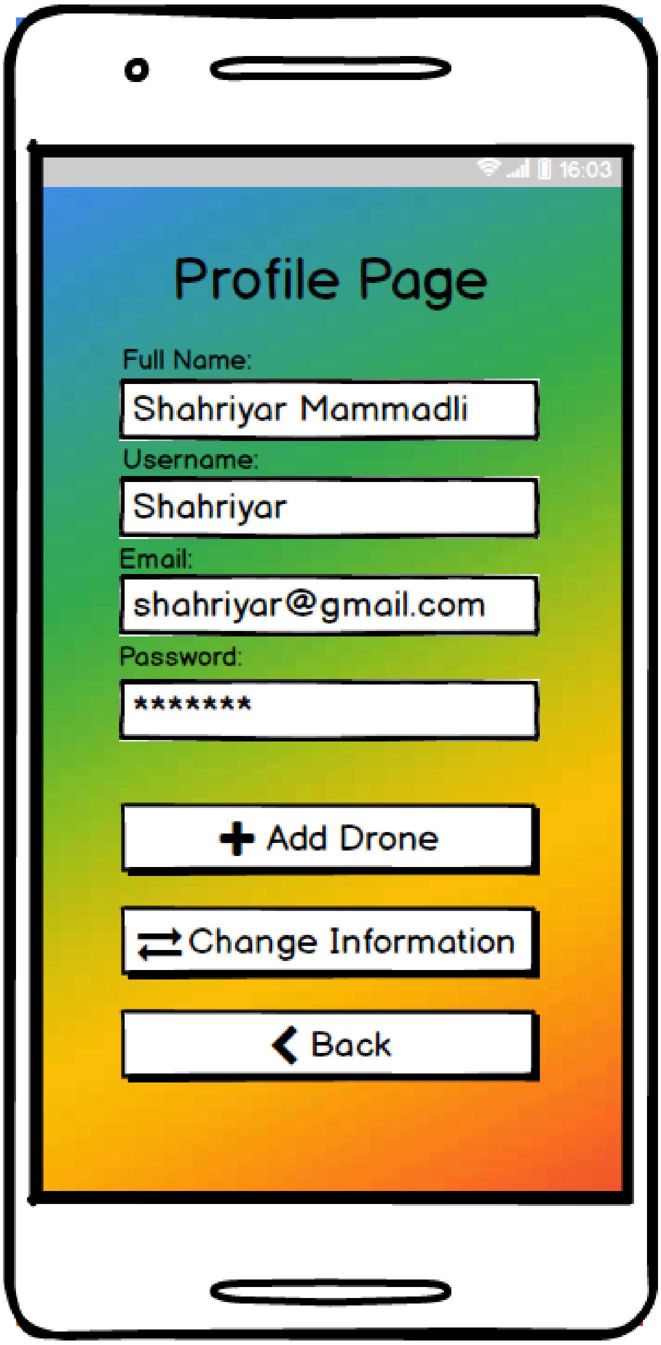
Alert

Do you want to send report to authorities?

Figure 12 **Report Page Screen Mockup**

Users can report the drone with a message to the authorities by clicking Report button. When they clicked the Report button, a popup message box showed up for approving. Back button is for going back to information page.

3.4.5.1.9. Profile Page



Full Name: Shahriyar Mammadli

Username: Shahriyar

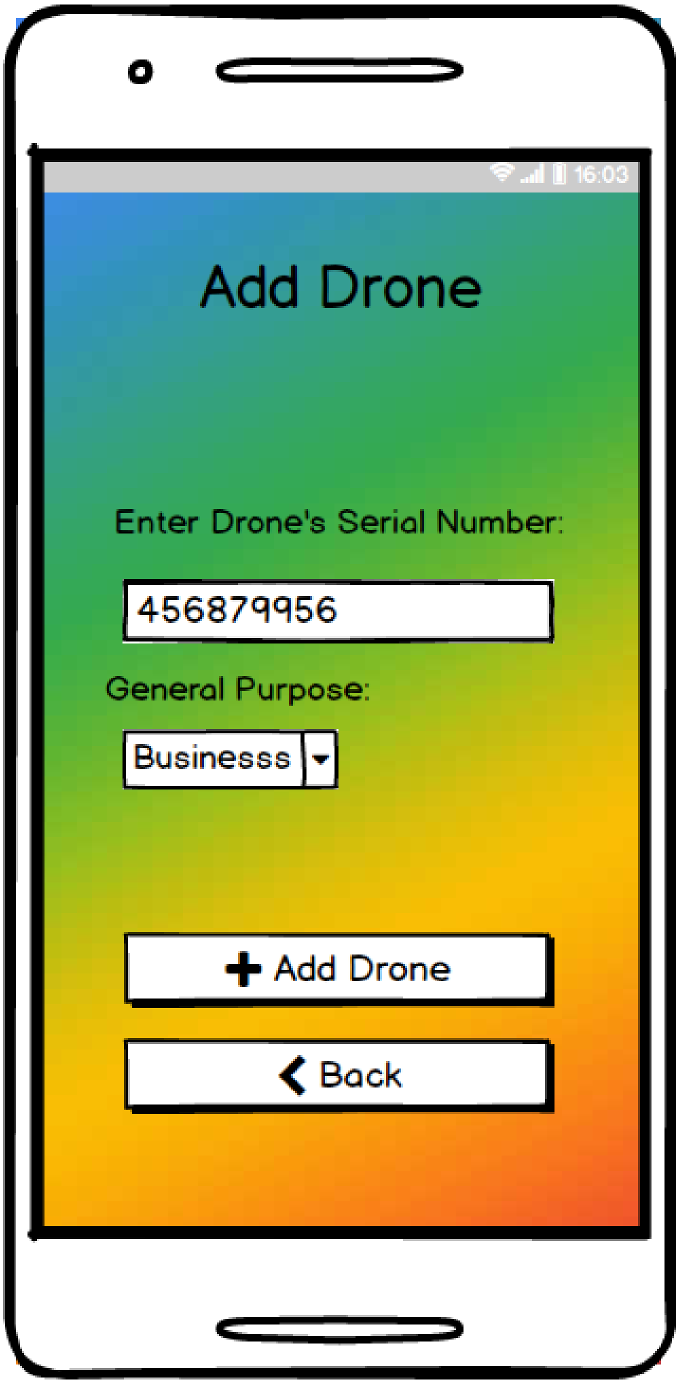
Email: shahriyar@gmail.com

Password: *******

Figure 13 **Profile Page Screen Mockup**

Users can see and change their profile information. They approve the changes when they click the Change Information button. Also users can add drone in their accounts by clicking the Add Drone button. Back button is for going back to view pages.

3.4.5.1.10. Add Drone Page



Enter Drone's Serial Number: 456879956

General Purpose: Businesss

Figure 14 **Add Drone Page Screen Mockup**

Users can add drone in this page. They have to enter drone's serial number and general purposes. After that by clicking Add Drone button, a popup message box showed up for approving. Back button is for going back to profile page.

4. Glossary

MySQL: My Structured Query Language

Android OS: Android Operating System

GPS: Global Pointing System

IOS: Iphone Operating System

GSM: Global System for Mobile Communications

5. References

[1] Object-Oriented Software Engineering, Using UML, Patterns, and Java, 2nd Edition, by Bernd Bruegge and Allen H. Dutoit, Prentice-Hall, 2004, ISBN: 0-13-047110-0.

[2] "Object Modeling." Google Drive. Accessed November 06, 2017. https://drive.google.com/file/d/0B9ApNnKlfgcHV29PYUw3a0gwUGc/view.