Design Document

Automatic Number Plate Recognition

BSc (Hons) Software Development Year 4

Student name: Michael Reid
Student ID: C00112726
Project supervisor: Mr. Nigel Whyte
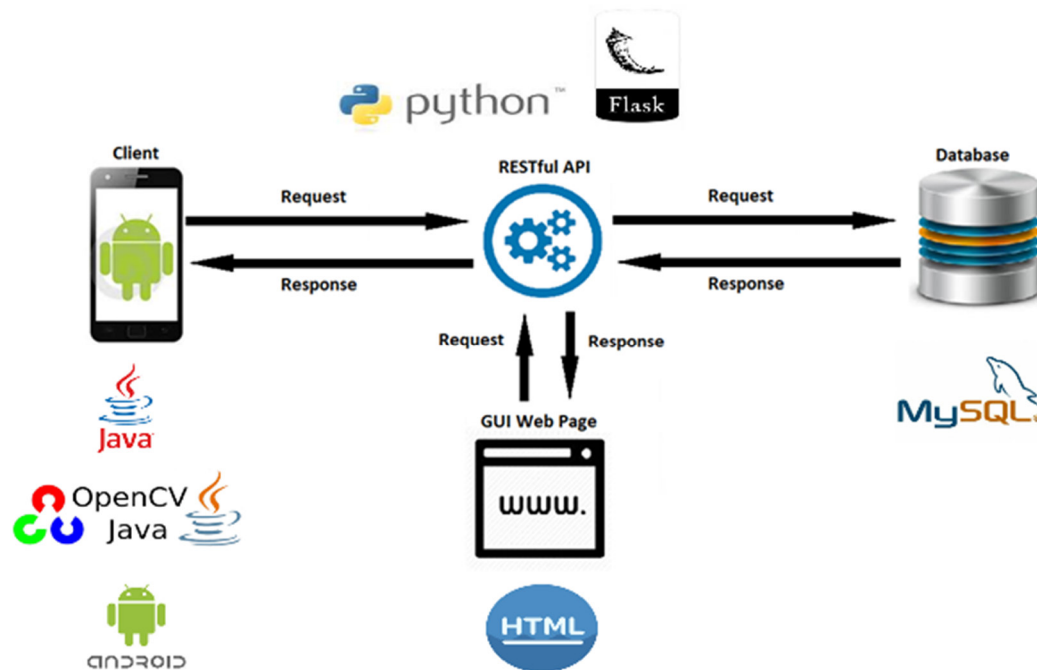
Table of Contents

1 Introduction

This project will involve creating a mobile phone application, which will allow a user to take pictures of vehicle registrations and check if that vehicle is permitted to be parked in the IT visitor's carpark. To do this, the application will analyse the image using various methods and algorithms, and extract a textual representation of the registration number. It will then query this text in an online database, and based on this result, will either give the all clear, send an automated email if the owner is a student or staff member asking them to move, or some other course of action if the owner is unknown to the IT. A web page will be available for staff to use to generate reports

2 Overview

The diagram below illustrates all the technologies used in this project.

2.1 Frontend

For the front end, 4 technologies will be used. These are:

1. Android
2. Java
3. OpenCV
4. HTML

Android, Java, and OpenCV will be used in the development of the Android app. HTML will be used to create the web page used for general CRUD operations as well as some report generating.

Java is the standard language used to develop applications on Android. As such, it will be used in this project. To aid with the image processing involved in this project, the OpenCV library will be used. It comes with an implementation for Java, although this is an interface which calls native optimised C++ code.

The web page will be coded in HTML with some minor CSS added for style. The web page will be highly limited in its functionalities as it is only an accessory to the app. The timescale of this project does not allow for a full web application to be developed.

2.2 Backend

For the backend, 3 technologies will be used. These are:

1. Python
2. Python Flask
3. MySQL

The WebService will be written in Python with aid of the Flask module. It will be in charge of interacting with the MySQL database at the request of either the Android app or the HTML web

page. By providing a service to handle the database, we improve overall security, as the database passwords will not be stored on a device or web page.

The WebService will contain two parts. One to allow the Android device to communicate, and one to allow the web page to communicate. Since the Android app will be sending and receiving JSON data, Flask_restful will be used to make manipulating the JSON easier. This would also allow for the future scalability of the project if that were to be an option.

3 Domain Model Diagram

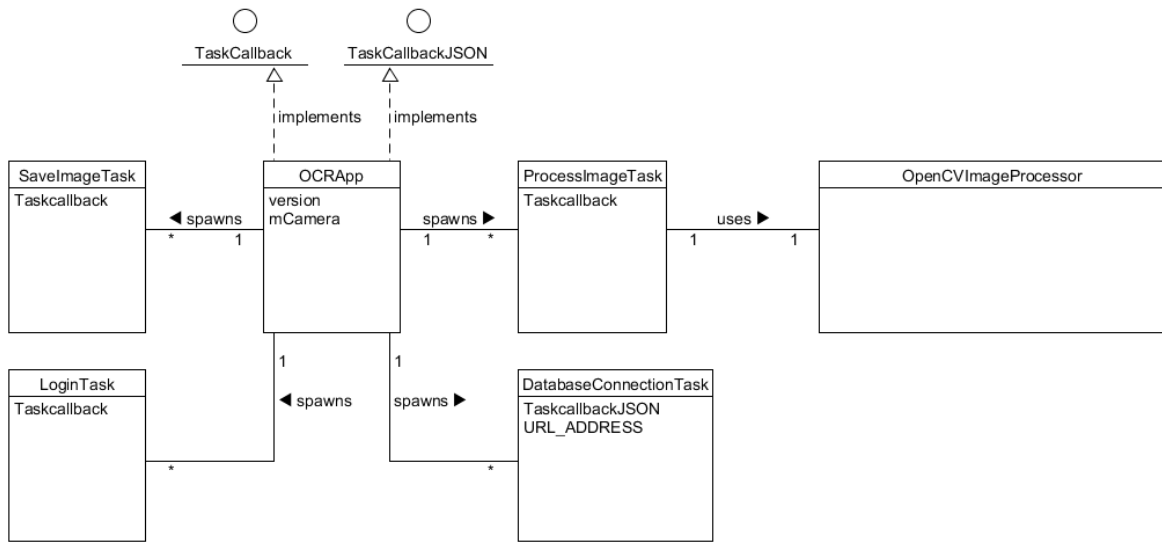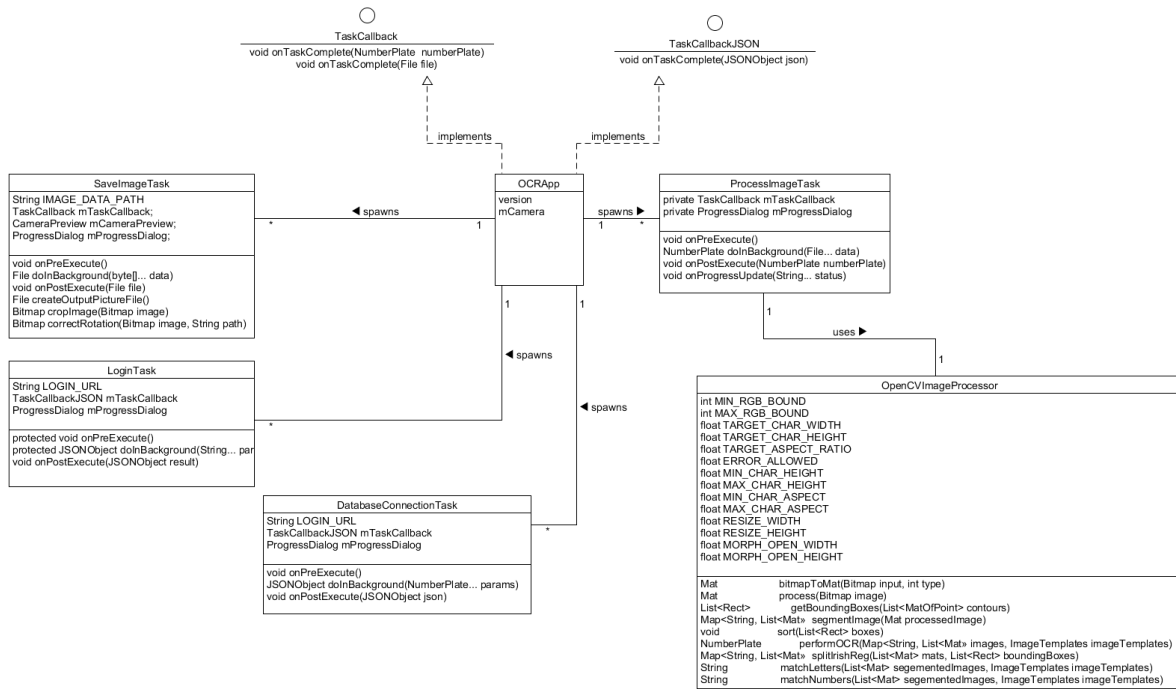The following domain model shows the general structure of the Android app.

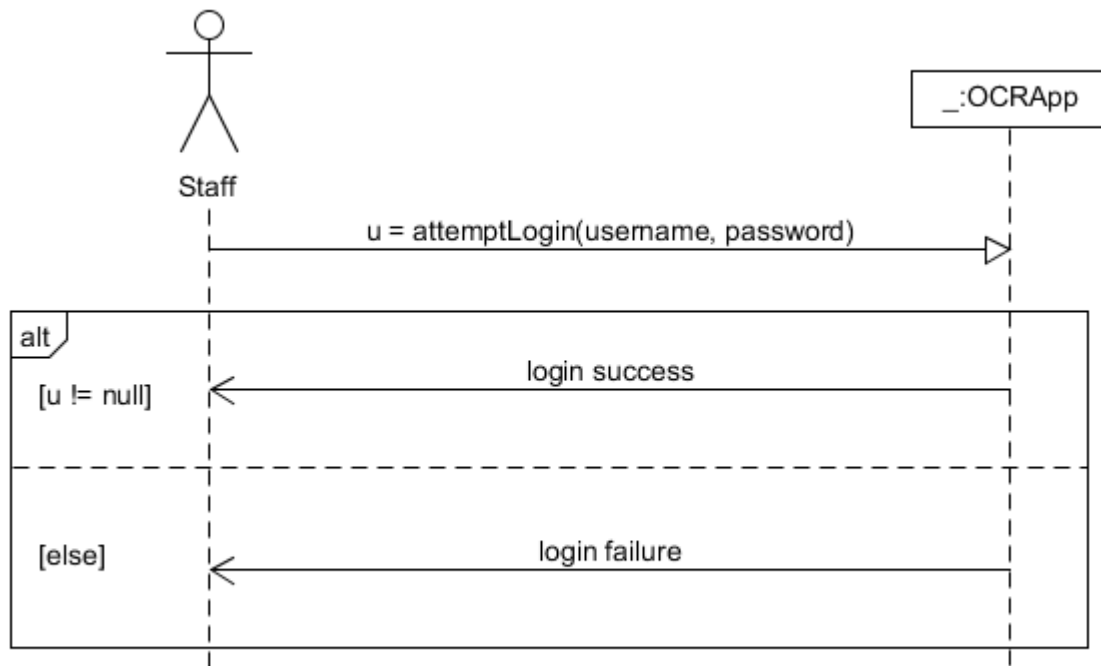*Figure 1 Domain Model for Android App*

4 Class Design Diagram

TaskCallback

void onTaskComplete(NumberPlate numberPlate)
void onTaskComplete(File file)

TaskCallbackJSON

void onTaskComplete(JSONObject json)

implements

implements

SaveImageTask

String IMAGE_DATA_PATH
TaskCallback mTaskCallback;
CameraPreview mCameraPreview;
ProgressDialog mProgressDialog;

void onPreExecute()
File doInBackground(byte[]... data)
void onPostExecute(File file)
File createOutputPictureFile()
Bitmap cropImage(Bitmap image)
Bitmap correctRotation(Bitmap image, String path)

OCRApp

version
mCamera

ProcessImageTask

private TaskCallback mTaskCallback
private ProgressDialog mProgressDialog

void onPreExecute()
NumberPlate doInBackground(File... data)
void onPostExecute(NumberPlate numberPlate)
void onProgressUpdate(String... status)

◀ spawns

spawns ▶

uses ▶

LoginTask

String LOGIN_URL
TaskCallbackJSON mTaskCallback
ProgressDialog mProgressDialog

protected void onPreExecute()
protected JSONObject doInBackground(String... par
void onPostExecute(JSONObject result)

◀ spawns

◀ spawns

DatabaseConnectionTask

String LOGIN_URL
TaskCallbackJSON mTaskCallback
ProgressDialog mProgressDialog

void onPreExecute()
JSONObject doInBackground(NumberPlate... params)
void onPostExecute(JSONObject json)

OpenCVImageProcessor

int MIN_RGB_BOUND
int MAX_RGB_BOUND
float TARGET_CHAR_WIDTH
float TARGET_CHAR_HEIGHT
float TARGET_ASPECT_RATIO
float ERROR_ALLOWED
float MIN_CHAR_HEIGHT
float MAX_CHAR_HEIGHT
float MIN_CHAR_ASPECT
float MAX_CHAR_ASPECT
float RESIZE_WIDTH
float RESIZE_HEIGHT
float MORPH_OPEN_WIDTH
float MORPH_OPEN_HEIGHT

Mat   bitmapToMat(Bitmap input, int type)
Mat   process(Bitmap image)
List<Rect>   getBoundingBoxes(List<MatOfPoint> contours)
Map<String, List<Mat>> segmentImage(Mat processedImage)
void   sort(List<Rect> boxes)
NumberPlate   performOCR(Map<String, List<Mat>> images, ImageTemplates imageTemplates)
Map<String, List<Mat>> splitIrishReg(List<Mat> mats, List<Rect> boundingBoxes)
String   matchLetters(List<Mat> segementedImages, ImageTemplates imageTemplates)
String   matchNumbers(List<Mat> segementedImages, ImageTemplates imageTemplates)
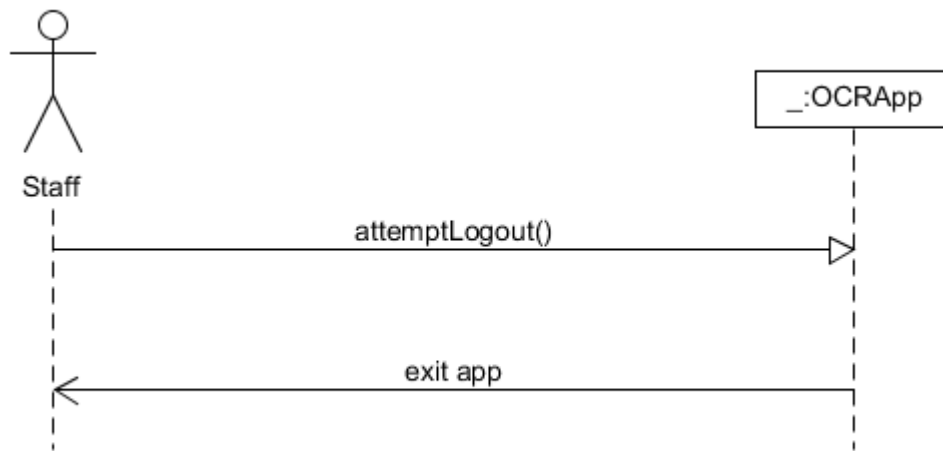
5 System Sequence Diagrams

This application performs the majority of its functionality in the background. As such, there is minimal interaction between the user and the app. Detailed below are the general interactions the user will make with the app.
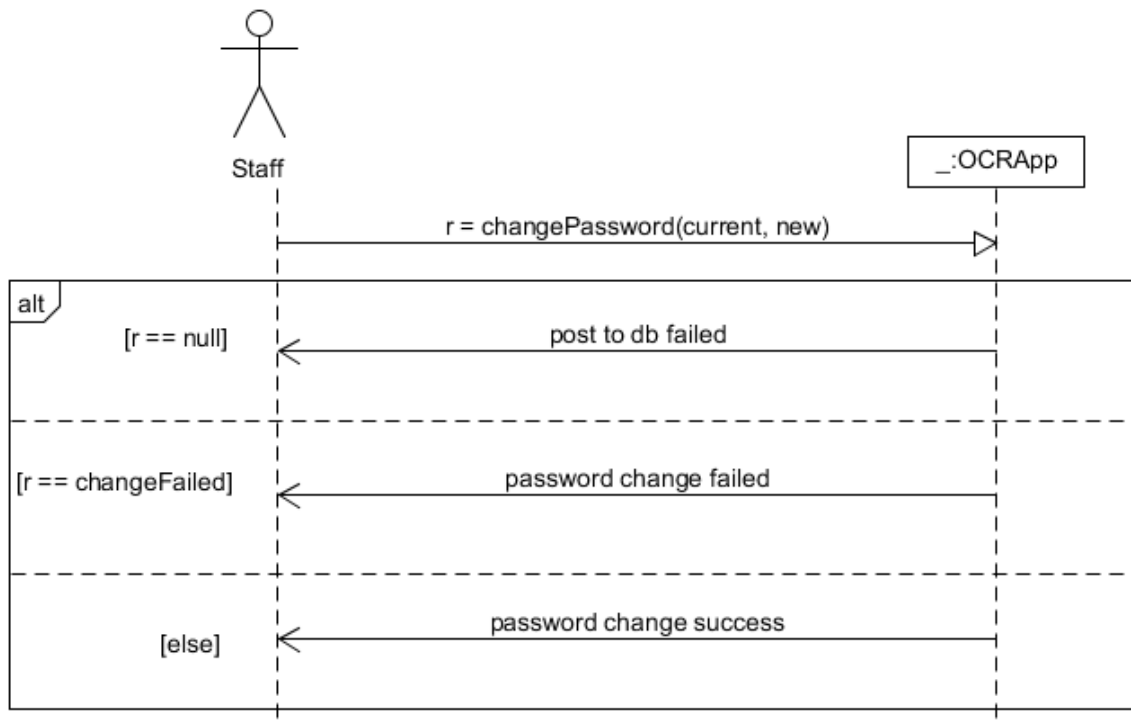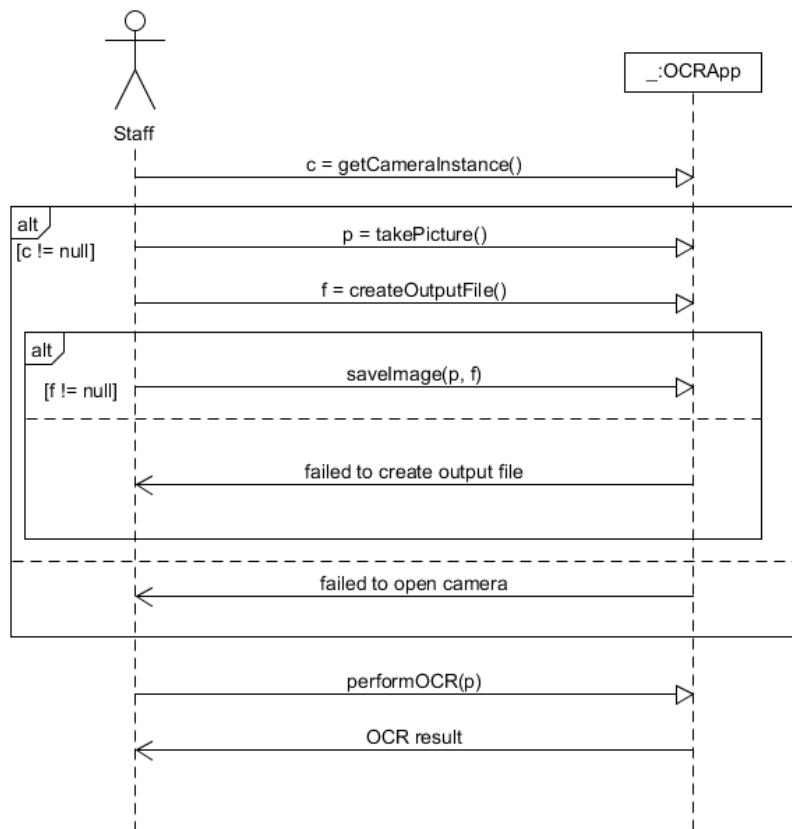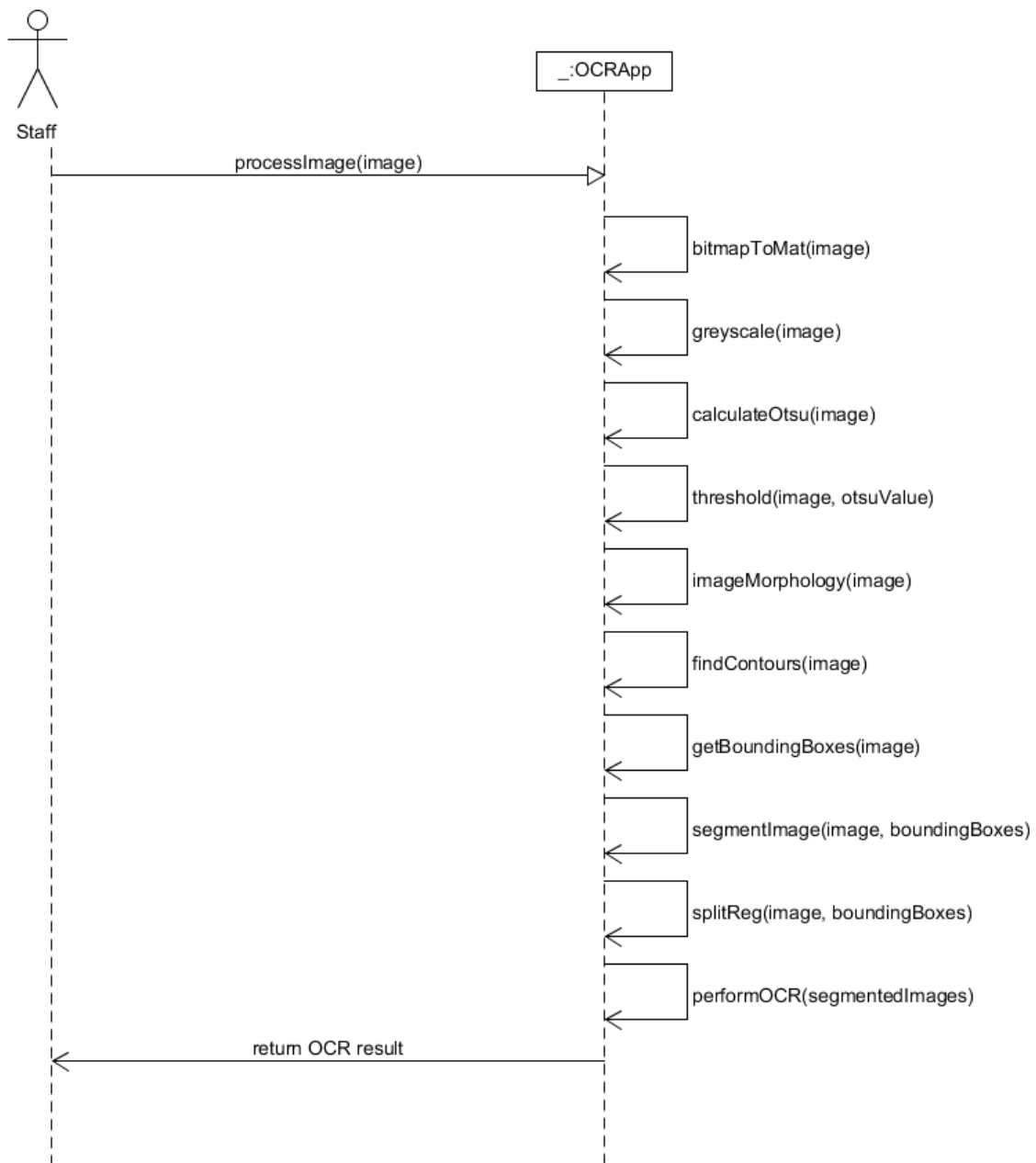
5.1 Login

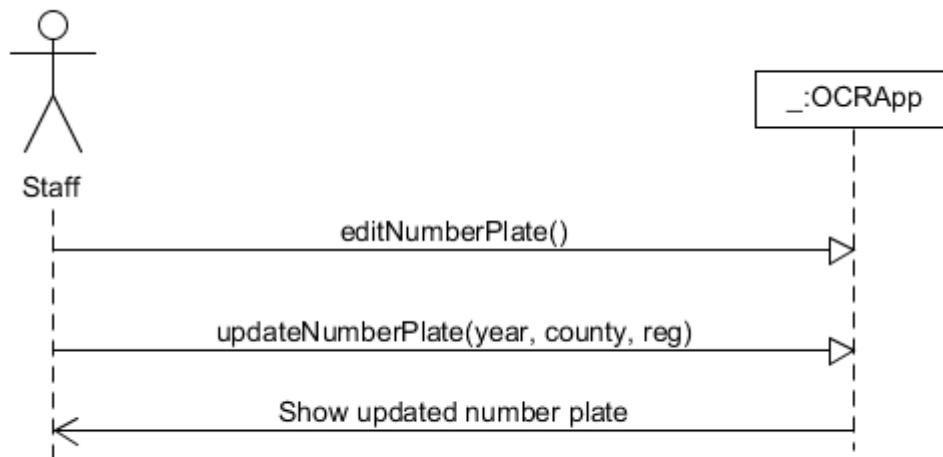## 5.2 Logout

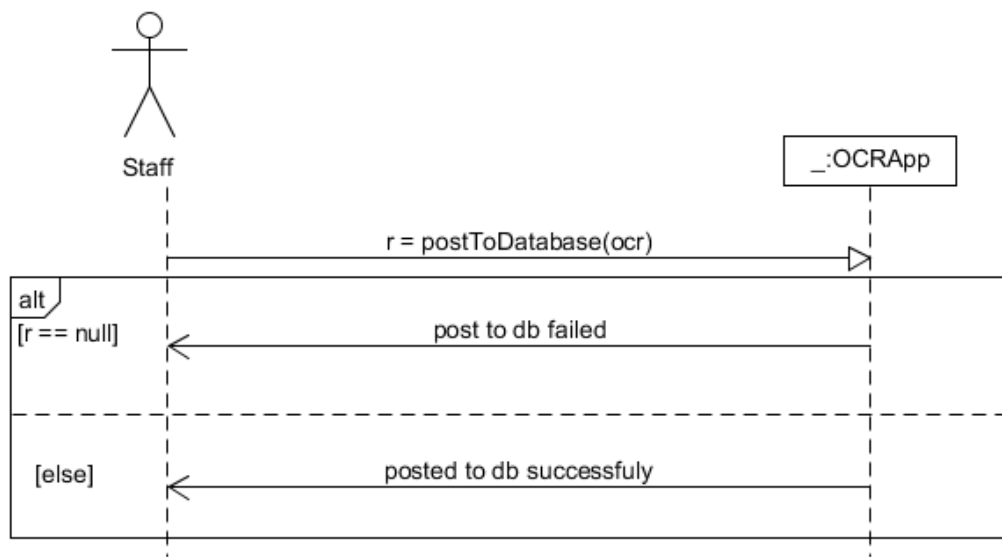## 5.3   Change Password

5.4 OCR Session

5.5 Process Image

5.6 Edit Number Plate

5.7   Post Registration to Database

6   Database Design

Described below are the tables in the project's database.

6.1   Staff

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	AUO_INCREMENT
username	varchar(255)	NO		NULL	
password	varchar(255)	NO		NULL	
email	varchar(255)	NO		NULL	

6.2   MainCarpark

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	AUO_INCREMENT
last_name	varchar(255)	NO		NULL	
first_name	varchar(255)	NO		NULL	
vehicle_registration	varchar(255)	NO		NULL	
email	varchar(255)	NO		NULL	

6.3   Infraction

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	AUO_INCREMENT
vehicle_registration	varchar(255)	NO		NULL	
time	timestamp	NO		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP
logged_by	varchar(255)	NO		NULL	

6.4   AdminStaff

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	AUO_INCREMENT
username	varchar(255)	NO		NULL	
password	varchar(255)	NO		NULL	

6.5   Log

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	AUO_INCREMENT
guess	varchar(255)	NO		NULL	
actual	varchar(255)	NO		NULL	
time	timestamp	NO		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP
logged_by	varchar(255)	NO		NULL	