CS3252 - Industrial Computer Engineering

CS2982 - Automation Challenge II

Semester 4 Project

Report

Smart Vehicle Tracker

Version 2.0

140045E – S. Arunan

140062D – G. P. P. D. Bandara

140381E - R. B. Mahanama

140684K - T. K. Wijesinghe

Introduction **3**
Business Environment 3
System Vision 3
Business Model 4
Requirements **5**
Interfaces 5
System Requirements Specification. 5
Functional Requirements 5
Non-Functional Requirement 5
Modelling **6**
Event Table 6
Use Case Modeling 7
Activity Diagram 8
System Sequence Diagram 9
Package Diagram 10
Design **11**
Fault Tolerance 12
Fault Hypothesis 13
Hard deadlines and Soft deadline situations 14
Expressway Systems 14
Car Park Systems 14
4+1 Architectural View 15
Project Planning **16**
Cost Planning 16
Time Planning 17
Quality Planning 17
Risk Planning 17
Implementation **18**
Hardware Resources 18
Software Resources 18
Main Server 18
Local Server 19
Mobile Application 19
Testing **20**
Phase 1 20
Phase 2 20
Phase 3 20

1. Introduction

1.1. Business Environment

In the modern world, everyone has to travel and most of the time people tend to use their private vehicle for most of the journeys. Finding a place to park the vehicle and the congestion levels in car parks and streets are among the main issues that people may overcome with the use of their private vehicle.

Google traffic is an alternative for the purpose of the congestion analysis tools which is widely available. However, with google traffic, we come across many problems relating to the validity and availability of data.

Google traffic does not provide any information about the traffic level at a specific point, for instance a car park. On the other hand, google traffic does not provide an accurate prediction of the vehicle congestion as the data is purely based on the movement of the android devices on the specific route or street.

Another problem domain we are trying to address is expressway systems. Vehicle users had to wait in expressway entrance and exits to get a ticket and to pay the ticket. An automated system would enable the traffic to move on without waiting.

1.2. System Vision

The Smart tracker system would provide a timely, accurate and real time data on level of traffic congestion at a specific point of interest, for instance a road or a parking lot. Unlike other systems like google traffic, the system would provide more accurate information as it is based on the number plate recognition which intern a direct attribute related with the traffic congestion.

In addition, the system would also be able track down specific vehicles in terms of last seen locations a presence at a specific point.

The data stream generated by the system would be able to be fed in to various system ranging from mobile phone applications for traffic and finding parking spaces to internal control systems in a building or a place.

1.3. Business Model

The system would comprise of both a hardware product and a software product.

The hardware product will be extension device for an existing camera system which would capture the video stream from the camera feed and acting accordingly. The extension device will have the capability to connect with the internet and upload and download the information relating to the service such as vehicles to be noticed or noted vehicles.

The software component will comprise of a mobile application and a web platform for retrieving data generated through the devices in the service. The web platform would provide information about the congestion at a specific place or a street which will be consumed by the mobile application used by the users of the system.

2. Requirements

2.1. Interfaces

Device architecture and the capability of the device is a major consideration for the implementation of the device

2.2. System Requirements Specification.

2.2.1. Functional Requirements

- Ability to capture the number plate of a vehicle within a given range of the camera accurately.
- Ability to process the video stream and capture the number plate within the given time frame.
- The device being capable of timely update the server and server being capable of timely updating the mobile devices and other consumers of the web service.
- The device capable of retrieving necessary data from the server and acting accordingly.

2.3. Non-Functional Requirement

- The readings taken from the video stream should be accurate.
- Availability of user friendly interface in the mobile application.

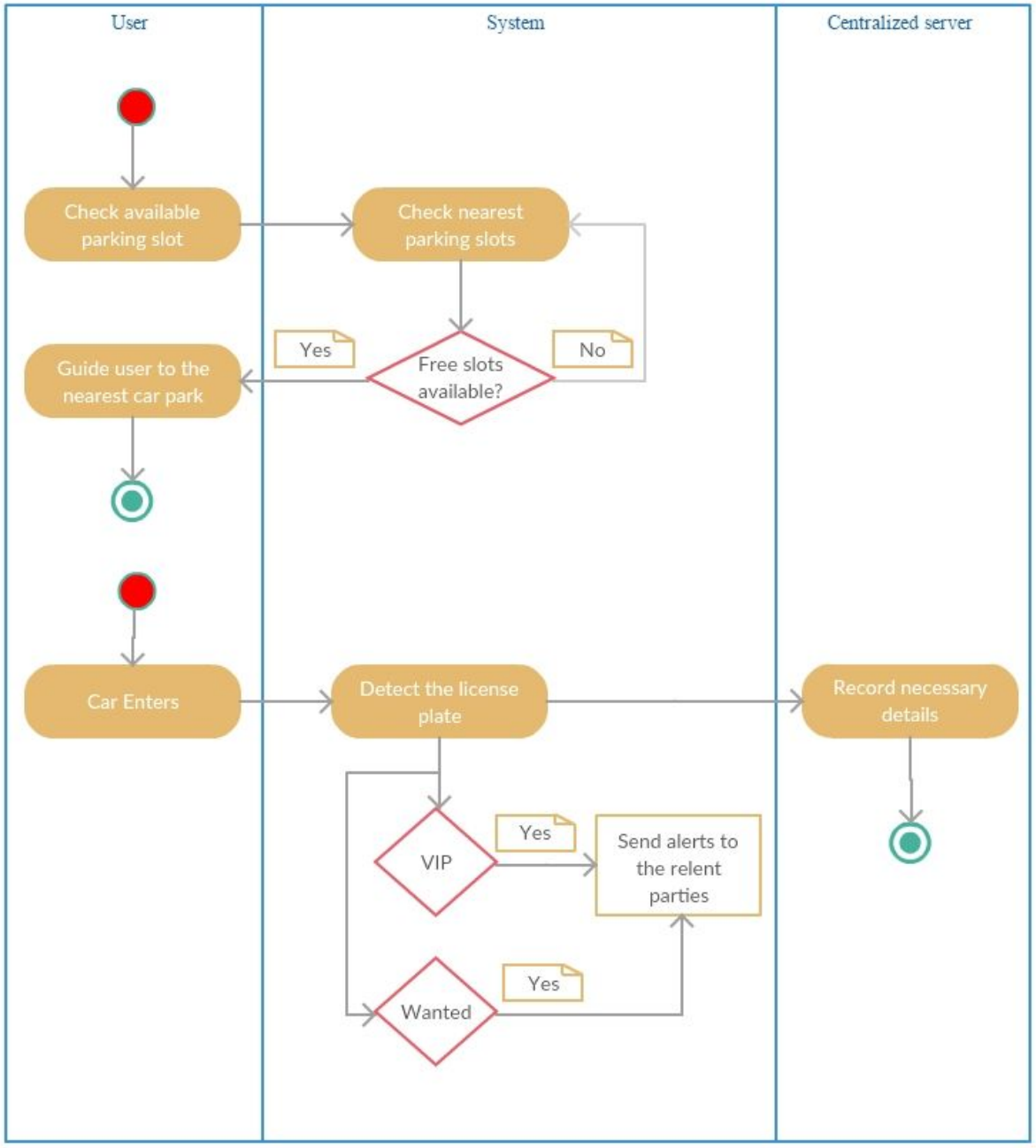
3. Modelling

3.1. Event Table

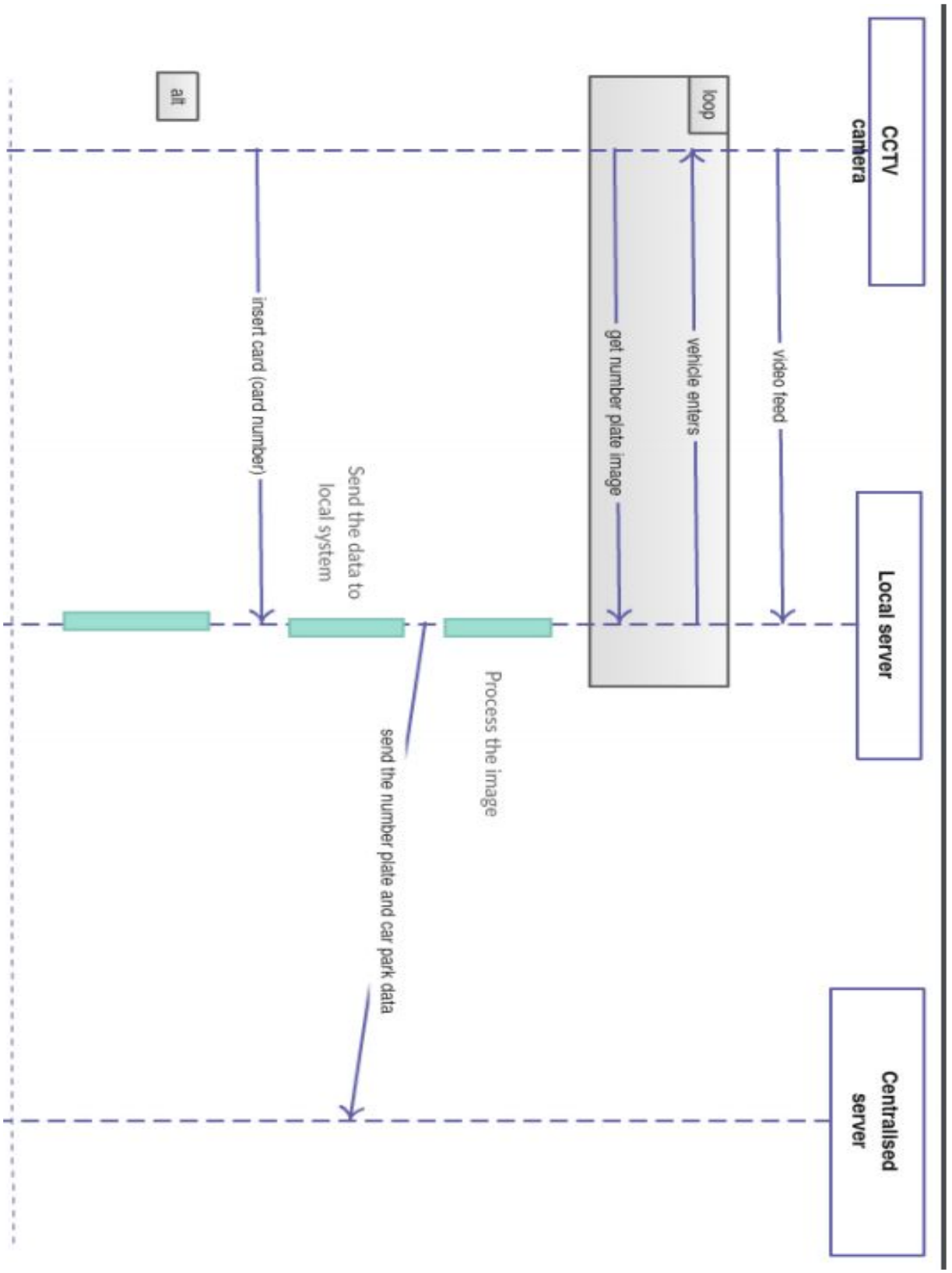
Smart Vehicle Tracker Event Table				
Event	**Source**	**Use Case**	**Response**	**Destination**
A new vehicle enters the gate	Camera	Read number plate	Number plate recognition data	Local vehicle registration service
Recognise the number plate	Number plate recognition system	Log license plate information	Number plate recognition data	Local logs, Database
Recognise the number plate	Number plate recognition system	Initiate payment	Number plate recognition data	Payment control system
Recognise number plate	Number plate recognition system	Check for VIP/Wanted	Vehicle identification data	Vehicle identification system
Number plate recognised as to be notified	Vehicle identification system	Alert the system	Alert data	Central alert system

3.2. Use Case Modeling

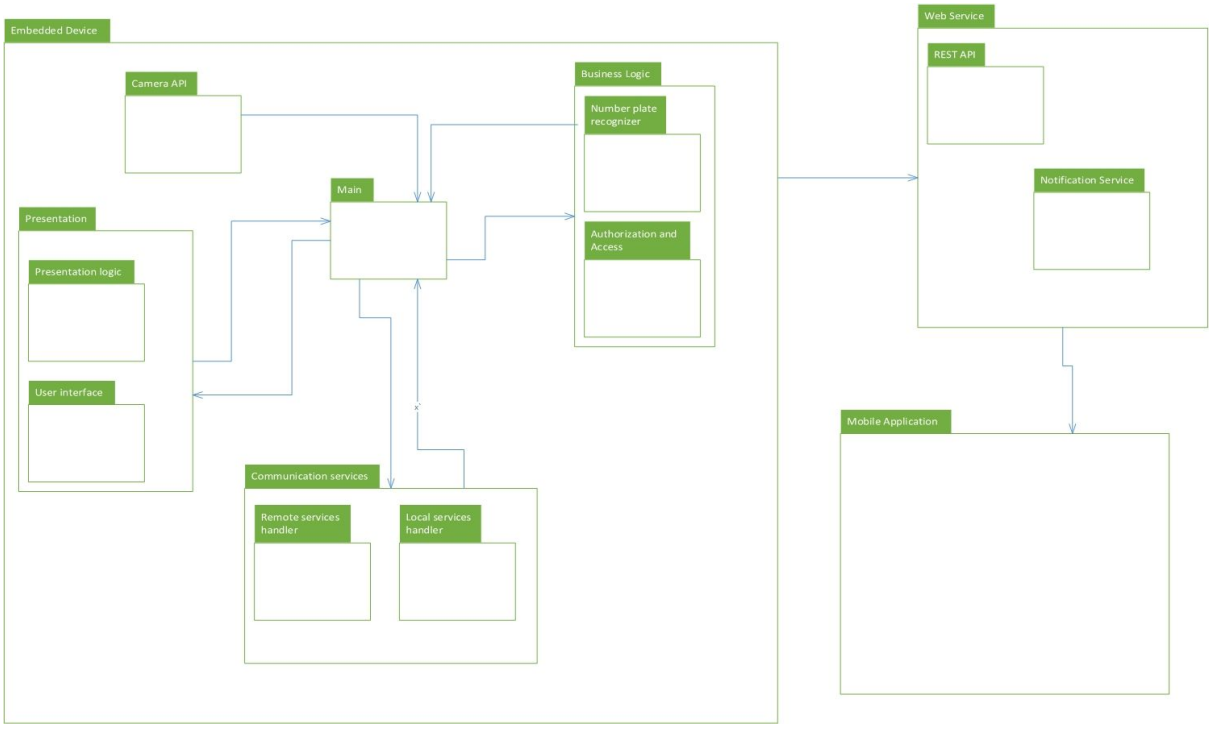
3.3. Activity Diagram



3.4. System Sequence Diagram



3.5. Package Diagram

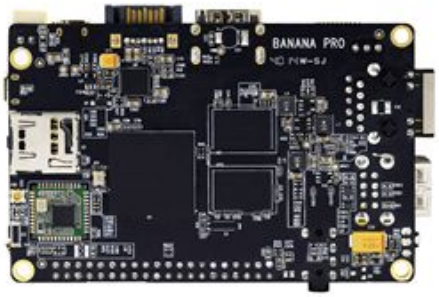


4. Design

The design of the system is to capture video footage from multiple cameras and process the video stream from the camera in order to detect license plates in the vehicles. The system is expected to be built in an iterative manner and the expected development plans for each iteration have been included under the construction section.

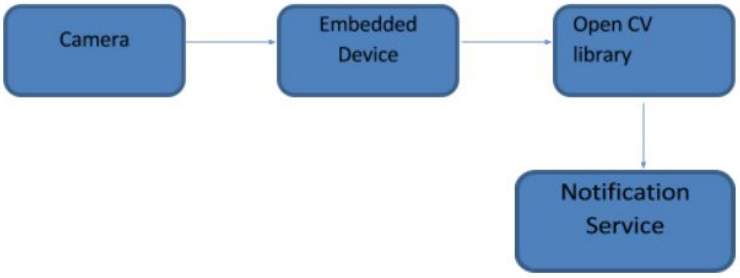
CCTV Camera

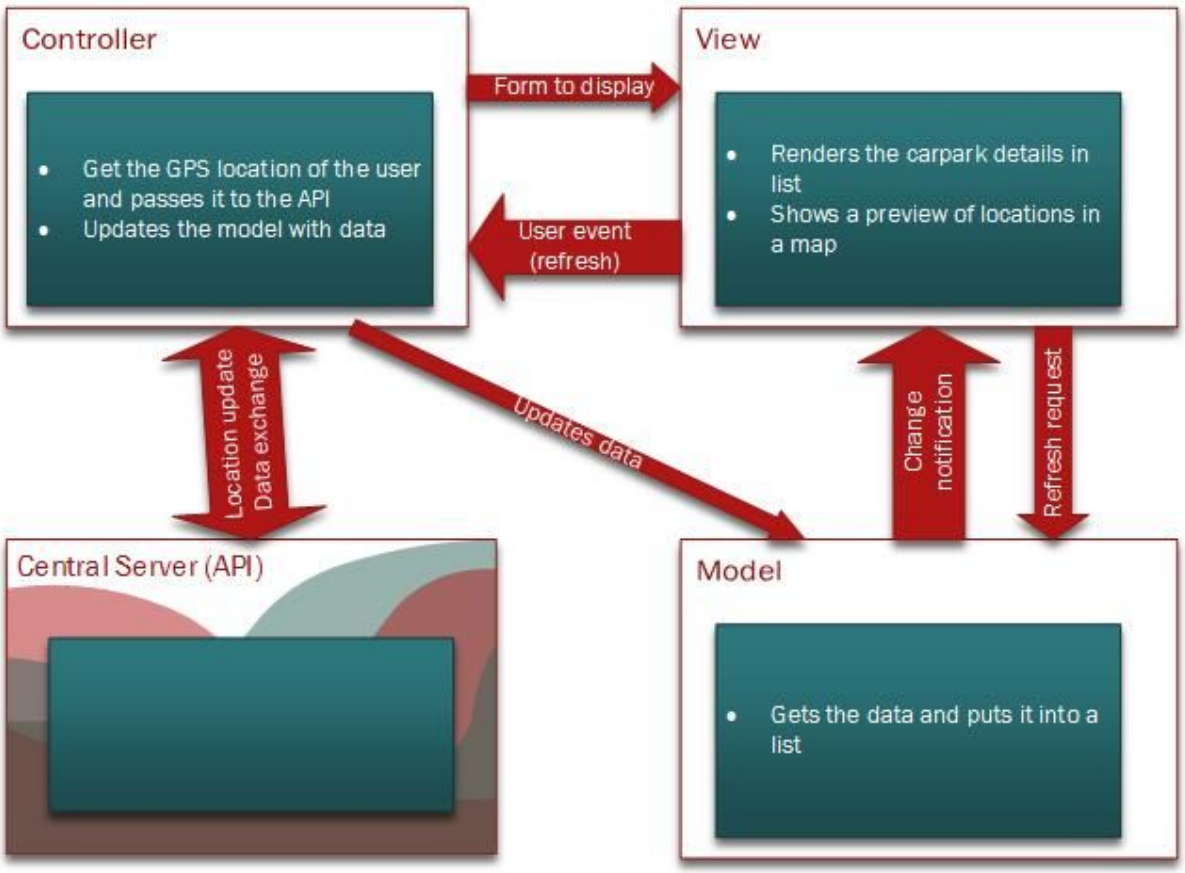
**See the note below



Banana Pro

Storage for operational purposes





4.1. Fault Tolerance

Since our application is working on real time certain measures have to taken in the system if unexpected faults occur. Otherwise the system will be compromised.

4.2. Fault Hypothesis

Possible Fault occurrence events	**Solutions**
Number plates not getting detected properly due to some grains in the number plate.	Gates would not open for the vehicle to pass and the vehicle should be manually validated.
Cameras not properly sending image feeds to process or the feed is not clear	If data is not processed continuously or frequently, then the administrator will be given an alert saying that there may be a problem in the camera.
Fault occurs in the connection.	There should be an alert for the manual operator to take over.
A sudden power failure	There has to be a backup generator installed or a manual operator who can continue on the process.

4.3. Hard deadlines and Soft deadline situations

We have two main applications for this project. Both have different hard, soft deadlines in terms of operation. So we are going to analyze it individually.

4.3.1. Expressway Systems

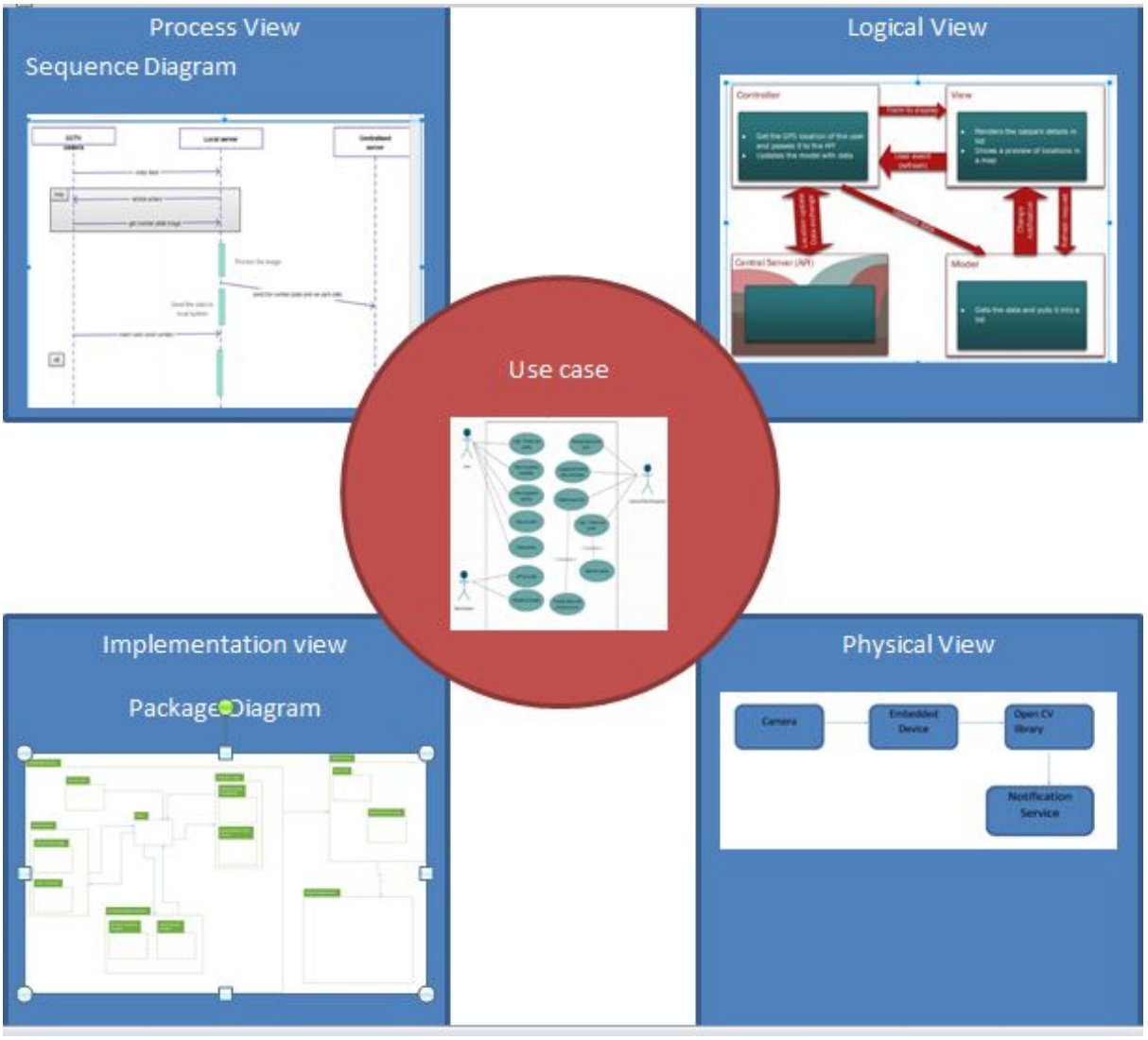
In expressway system our main motive is to eliminate waiting time in toll booths. Our idea is to allow the vehicle to cross the toll booth at a desired level speed. (Not too high speed or low speed) Once the vehicle approaches the gate, the local subsystem (embedded device) will process the video and send the number plate to expressway controller system. The system has to process everything within the time the vehicle crosses the toll booth. In theory the real time deadline will be the distance between the vehicle and the gate divided by the average speed of the vehicle.

To reduce the time, we are sending the processed data to the expressway controller directly without sending it to the centralized server. Centralized server does not have anything 4+1 vmodto do with the expressway controller system. The embedded device consists of a Banana Pro board which have the necessary processing power. When deploying it commercially the processor can be replaced with higher capabilities according to the need.

4.3.2. Car Park Systems

In car parks system the mobile application will notify the users with available parking slots in a nearby car park. There will be a soft deadline for the centralized servers to update in the mobile applications because the data is still useful. There can be a latency time of 30-60 seconds in updating the central server. A reasonable real-time deadline would be 1-2 minutes.

4.4. 4+1 Architectural View



5. Project Planning

5.1. Cost Planning

Since the project is to build the system as an extension to an existing system, the system to be developed should cost have to be managed in order to attract the customers significantly.

For the implementation, a banana pi development board has been used as the development platform. In addition. A banana pro camera module has been used to simulate a feed from a camera system.

The hardware costs associated with the costs are as follows,

Description	Quantity	
Banana PRO development board	1	9950.00
Banana Pro camera module	1	5500.00
Total		15450.00

The software components for the first phase have been taken from free and open source vendors.

For the second phase of the project, the system is expected to be implemented for multiple cameras with a central service station managing the data store of the number plates detected in through the camera system. Furthermore, second phase will also require the local service centre (image processing unit) to have more computing capacity. Hence a desktop CPU has to be used for the implementation of the second phase.

The costs associated with the second implementation are as follows,

Description	Quantity	
CPU - local server for a single system	1	50000.00
Web camera	3	9000.00
Total		59000.00

5.2. Time Planning

The first phase of the project is expected complete in a time frame of 6 months. During the first phase all the basic components in terms software needed for subsequent phases are expected to be developed.

The second phase of the project is expected to be completed in a time frame of another 6 months where the system will be developed to handle multiple cameras in a system.

5.3. Quality Planning

Since the application is expected to be deployed in commercial environments, the quality is a key factor in the project. The quality of the information generated by the system is planned to be improved using optimizations in terms of accuracy.

Software quality of the project is planned to maintain through code reviews and refactoring at the end of each phase in order to ensure maximum customizability of the system.

5.4. Risk Planning

Risk description	**Potential impact on project**	**Likelihood of Occurrence**	**Overall threat**
Connecting multiple cameras to the embedded device	High	Low	High
Number plates not being printed in the format expected by the system	High	Medium	High
Power failure	High	Low	Medium
System, drivers and hardware malfunctioning	High	Low	Medium
Time frame	High	Medium	High
Loss of a member in the team	High	Low	Medium

6. Implementation

The device is a commercially viable product as the product can be implemented as an extension to an existing system as well as covers a wide range of use cases that many people come across in the day today lives. The system will also generate data streams that will be available for the a wide range of audience ranging from general users to the planning authorities and security authorities. The automated monitoring will also eliminate any costs that were incurred for monitoring and notifications.

The tools to be used throughout the project in the project are as follows,

- OpenCV - Image Processing
- Android - Mobile application
- LAMP stack - Server side processing
- GitHub - Version Controlling
- Banana Pro - Development of hardware component
- Creately - Diagramming tool

6.1. Hardware Resources

The embedded system will require a development board capable of supporting image processing. Thus the embedded device will be developed using a Banana Pro.

During the initial phase the camera module for the banana PRo will be used for the implementation and testing purposes. Subsequently, the system is expected to test the compatibility with different classes of cameras.

6.2. Software Resources

6.2.1. Main Server

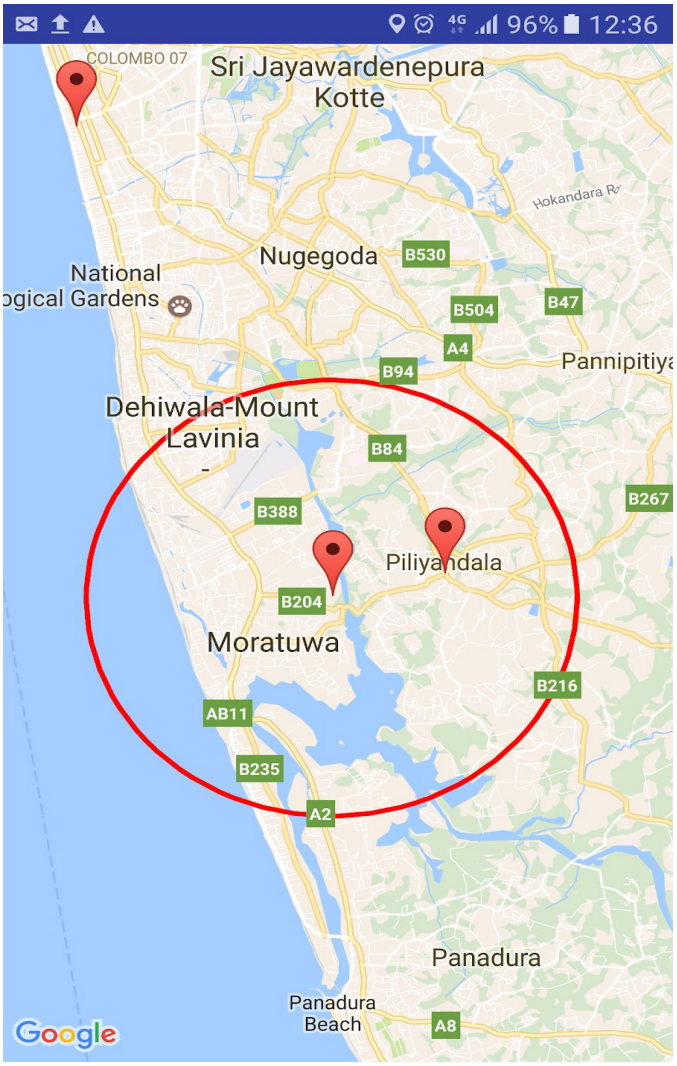
A server machine with advanced capabilities has to be used to manage the traffic. There has to a centralised secure server which could be used to store the data. Currently the server is built in our own laptops for testing purposes.

6.2.2. Local Server

The embedded system will require OpenCV library to python for the processing of images and PyQt library for graphical user interface rendering. The library has to be installed with the required dependencies for the project on the Banana Pro. Furthermore the applications stack on embedded device will also require libraries that can pull data from and push data to a remote server.

6.2.3. Mobile Application

The mobile application should be developed for main mobile phone platforms like IOS, Android and Windows. There also a possibility to use cross platform build tools like ionic but building natively would make application more flexible and speedy. For the model purpose we developed the mobile application in Android.



7. Testing

7.1. Phase 1

- Testing for images simulating the real world scenarios was done using a set of images
- The test images set included 90 images of which 50 was identified by the system.
- Further tuning has to be carried out in order to improve the recognition of number plates. For instance to detect between the similar looking characters in number plates such as ‘O’ and 0.

Tests which were carried out on the prototype of the first iteration has proven substantial success in terms of accuracy.

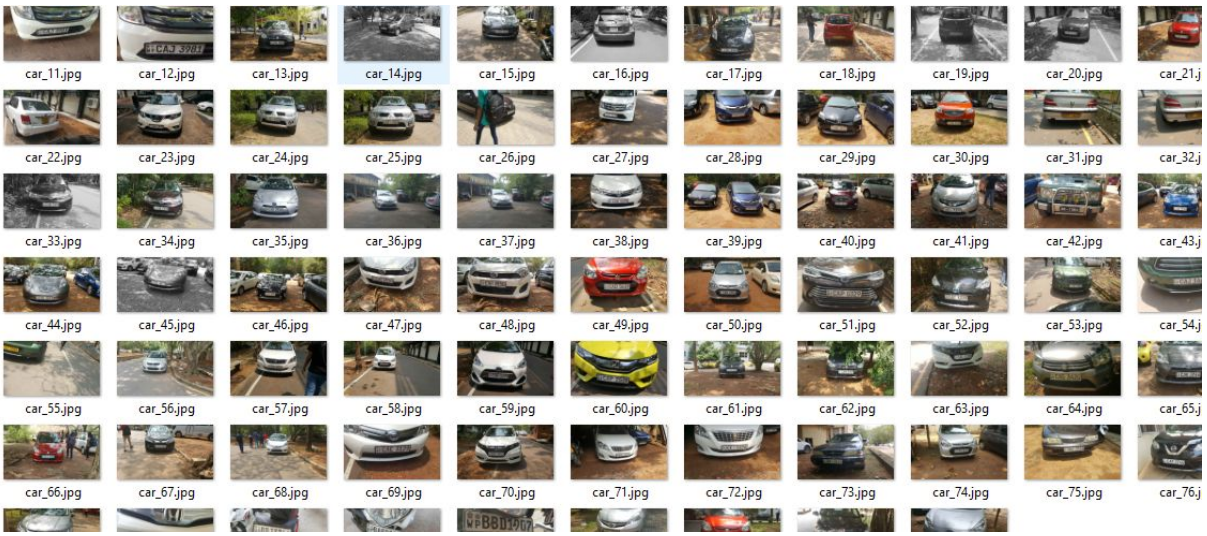
7.2. Phase 2

- In the second iteration, we tested our product with the previous test images while providing pictures of the vehicle with the recognized number plate through the GUI.

7.3. Phase 3

- In the third iteration, we tested our product with the previous test images while providing pictures of the vehicle with the recognized number plate through the GUI.

All the testing have been done by our project which consists of four members. For testing purposes we took images and videos around our university premises and collected image datas to analyze the performance of our algorithm.



7.4. Statistics of our testing phase

Number plates tested	85
Number plates identified (100%)	41
Number plates identified (One character misplaced)	24
Number plates identified (2 or 3 chars misplaced)	9
Number plates identified (4 or 5 character misplaced)	6
Number plates identified (At least one char got correct)	5
Number plates completely not identified	None
Pass rate	48.23%