

# ARCHITECTURAL REQUIREMENTS

PROJECT: TRAFFIC CAMERA IMAGE ANALYSIS

CLIENT: DPSS, CSIR

TEAM: QUADCORE PRODUCTIONS

**Author(s):**

Mpho BALOYI

Hlengekile JITA

Mayimela MOSES

Mbhele THEMBA

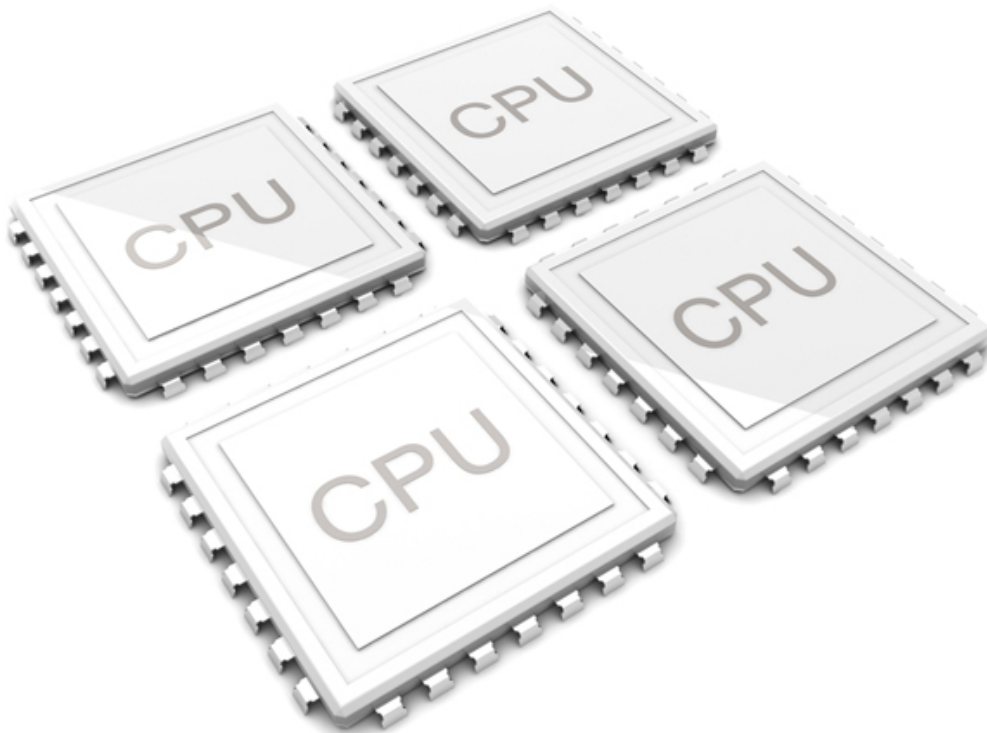
**Student number(s):**

14133670

14077893

14019702

14007950



University of Pretoria, Department of Computer Science  
29 July 2016

# Contents

<b>1</b>	<b>Versions</b>	<b>2</b>
<b>2</b>	<b>Introduction</b>	<b>3</b>
<b>3</b>	<b>Vision</b>	<b>3</b>
<b>4</b>	<b>Background</b>	<b>3</b>
<b>5</b>	<b>System Scope</b>	<b>4</b>
<b>6</b>	<b>Functional Requirements</b>	<b>4</b>
6.1	Use Cases . . . . .	4
6.2	Service Contracts . . . . .	5
6.2.1	Add Route . . . . .	5
6.2.2	Modify Route . . . . .	5
6.2.3	Delete Route . . . . .	5
6.2.4	Get Route Traffic . . . . .	5
6.2.5	Schedule Notification . . . . .	6
6.3	Required Functionality . . . . .	6
6.3.1	Add Route . . . . .	7
6.3.2	Modify Route . . . . .	8
6.3.3	Delete Route . . . . .	9
6.3.4	Get Route Traffic . . . . .	9
6.4	Process Specification . . . . .	10
6.4.1	Add Route . . . . .	10
6.4.2	Modify Route . . . . .	10
6.4.3	Get Route Traffic . . . . .	10
<b>7</b>	<b>Domain Model</b>	<b>10</b>
<b>8</b>	<b>Open Issues</b>	<b>10</b>

# 1 Versions

## 2 Introduction

This document describes the functional requirements and application design of the Traffic Camera Image Analysis System. The target system will run on a web server and be accessed by users on an Android Application which will provide the users with the necessary functionality to access real-time traffic information that assists them with things such as avoiding traffic and choosing the best alternative routes.

In this specification, we cover the use cases, their service contracts, required functionality and process specifications of the target system. The above, will be built on over time as the software is developed, as we are following an agile development methodology. In addition the domain model of the application as a whole will be provided.

## 3 Vision

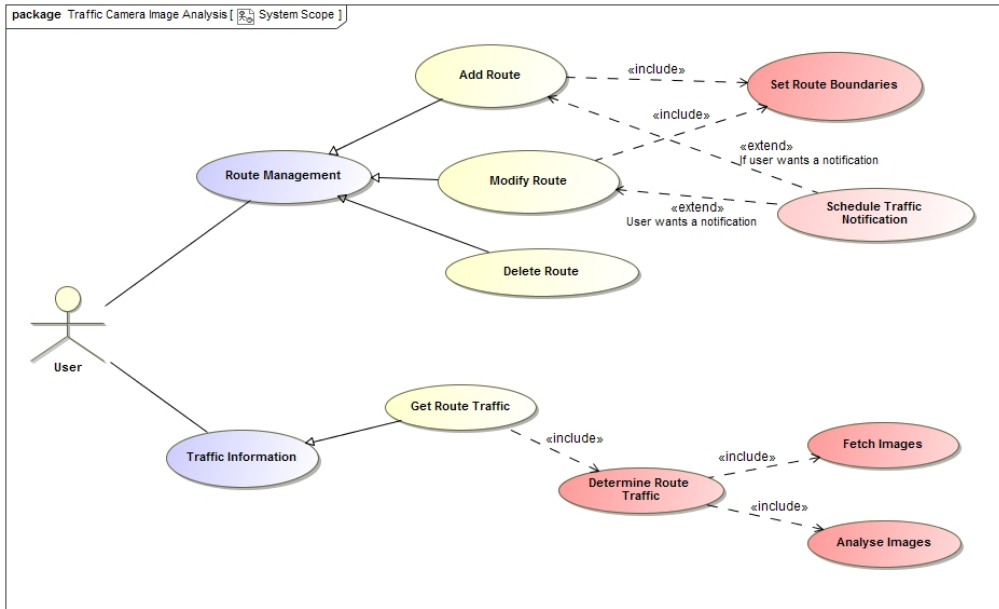
For this project we aim to achieve a system that makes use of images obtained from highway cameras to provide users with up-to-date real-time traffic information. The system should simplify the user's travels by providing traffic information and notifying them before they depart of traffic conditions, calculating arrival times based on traffic conditions and help them select the most suitable route for their trips using the traffic information and additional metrics. Our vision for the target system is that it should be reliable and perform relatively quickly, both for user satisfaction and in order to be the an up-to-date traffic information system.

## 4 Background

As a commuter, traffic is something that is a part of everyday life, and it is not one of the more pleasurable aspects of life. Already there is software in place that assists us in dealing with this problem, such as Google Maps. This software uses the crowd-sourcing of GPS data in order to provide their up-to-date traffic information.

In our system, we want to take an image analysis approach to solve the same problem. We will make use of the publicly available SANRAL highway cameras to get images. Processing these images we will perform image analysis and determine the traffic conditions in the area of a camera. Using this information we will be able to generate information pertaining to user specified routes in order to provide the information necessary to help them avoid traffic and choose the most suitable routes in order to do so.

## 5 System Scope



The purpose of this system as stated earlier, is to provide the user with traffic information based on image analysis from images accessed from highway cameras. Thus the user will need to be able to access services that allow the achievement of this goal.

The scope of the system thus includes the tasks the user can access such as add route, modify route, delete route and get route traffic.

## 6 Functional Requirements

### 6.1 Use Cases

The use cases for our system are described below. These use cases are the services that provide the user of the application some value. The following uses are listed in order of prioritization:

- Add Route (Critical)
- Get Route Traffic (Critical)
- Modify Route (Critical)
- Delete Route (Critical)
- Schedule Notification (Important)

- Route Optimization (Nice To Have)

## 6.2 Service Contracts

For each of the above services described above, there are pre-conditions that must be met for the service to be carried out and post-conditions that are met to indicate the successful completion of a service. In addition the service contracts indicate what exceptions will be thrown in case the service fails.

### 6.2.1 Add Route

**Pre-Condition:** Valid route boundaries.

Route does not already exist.

**Post-Condition:** New route is created and stored.

This use case creates a new route and stores it so that it can be used in requests for traffic information. It first checks that the route to be created is valid and does not already exist in storage, then it creates and stores the route. In the case that the route exists an exception is raised.

### 6.2.2 Modify Route

**Pre-Condition:** Route exists.

**Post-Condition:** Route is modified and the modified route is stored.

This use case makes changes to existing routes and stores the changes. In order to do this, this use case first checks if the route exists, if so it then makes the necessary changes to the route and then stores the route. If the route does not exist an exception is raised.

### 6.2.3 Delete Route

**Pre-Condition:** Route exists.

**Post-Condition:** Route is removed

This use case removes an existing route. Before it does this it checks if the route exists in storage then removes it. In the case that the route does not exist an exception is raised.

#### 6.2.4 Get Route Traffic

**Pre-Condition:** Route consists of camera locations.

Camera locations on route has valid images.

**Post-Condition:** Traffic information for route is generated.

This use case generates traffic information based on a route provided with the request. To do this it first checks if the route covers any of the camera locations, if not no traffic information is generated and an exception is raised. Otherwise it continues and retrieves the images for the camera's within the route, thereafter it checks if these images are valid, if all are invalid, no traffic information is generated, it then performs image analysis on the valid images and determine the traffic levels and generates traffic information for the route.

#### 6.2.5 Schedule Notification

**Pre-Condition:** Valid times

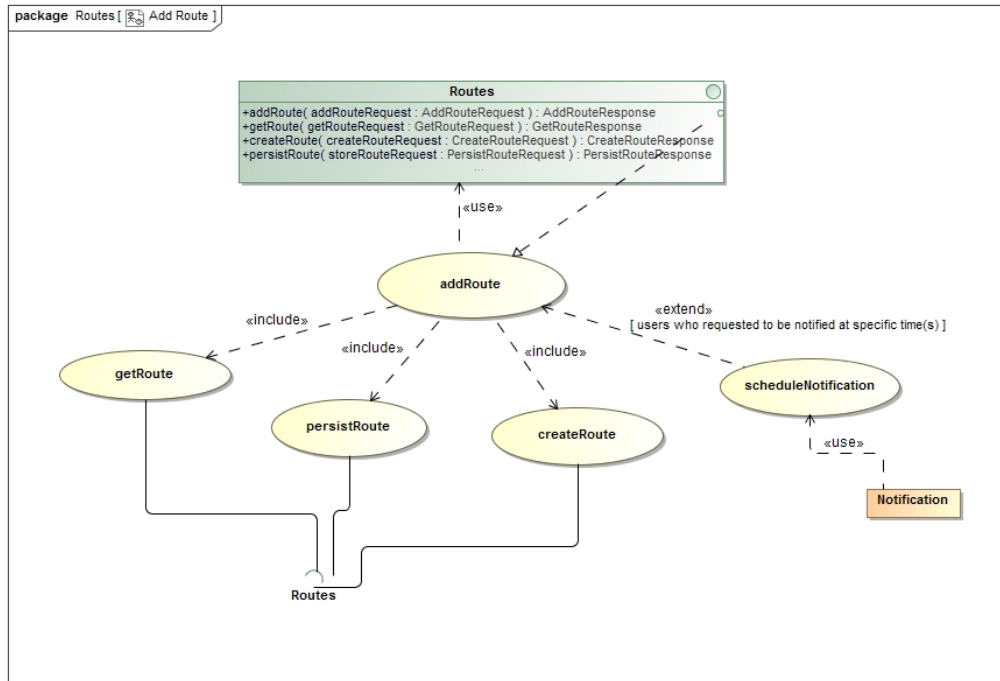
**Post-Condition:** Notification is scheduled

This use case allows the user to schedule notifications of traffic information at specific times. This is done by allowing the user to specify a time to be notified about a route when adding or modifying it. This time is then used to schedule automated traffic information requests.

### 6.3 Required Functionality

The use cases make use of other lower level functions in order to achieve their purpose. In the below diagrams for each high level use case, the lower level use cases that are required are described.

### 6.3.1 Add Route

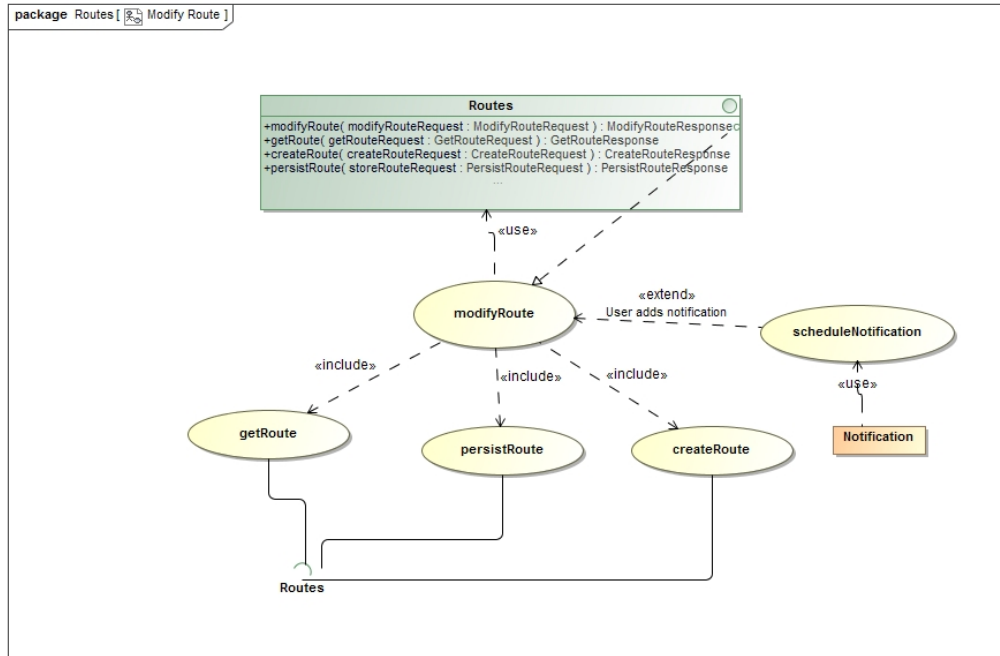


The Add Route use case makes use of the following lower level functions to add a route to the system:

- **getRoute** - This checks if the route exists already or not.
- **createRoute** - This creates a route, or a set of directions, based on the route boundaries (departure and destination points).
- **persistRoute** - This stores the route so that it is available for later use.



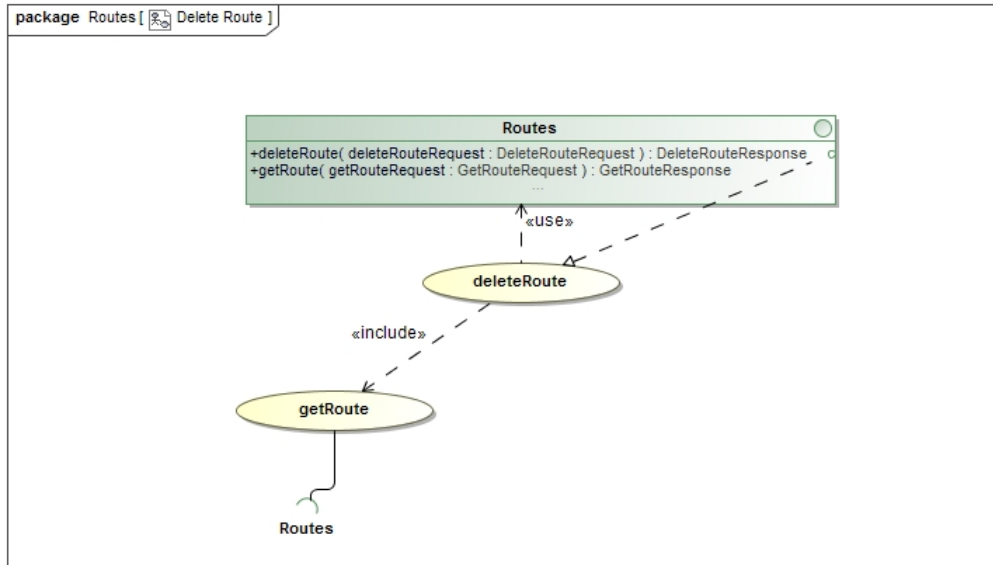
### 6.3.2 Modify Route



The Modify Route use case makes use of the following lower level functions to make changes to an already existing route.

- **getRoute** - This checks if the route exists already.
- **createRoute** - This creates a route, or a set of directions, based on the route boundaries (departure and destination points). Only if the changes are in the route boundaries.
- **persistRoute** - This stores the modified route.

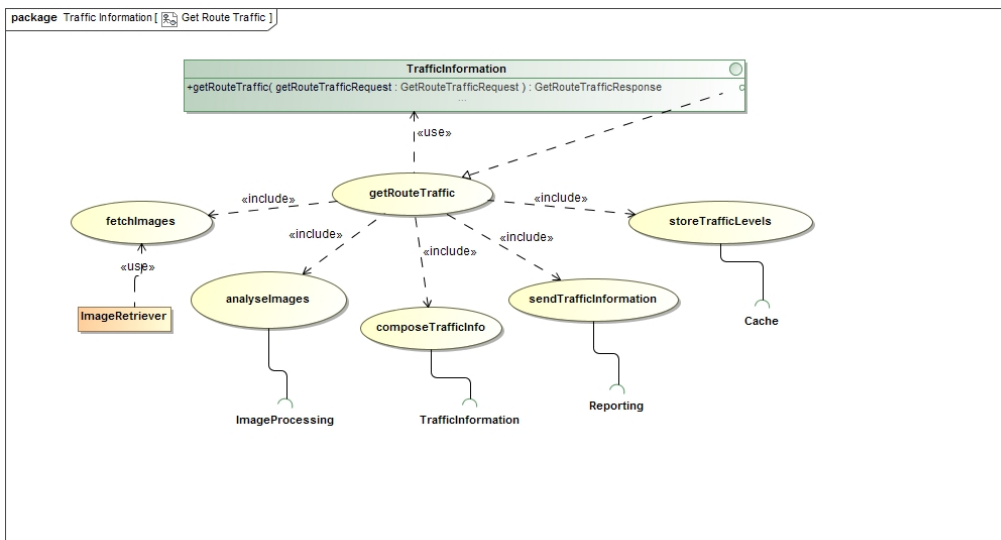
### 6.3.3 Delete Route



The Delete Route use case makes use of the following lower level functions to assist in removing the route from storage:

- **getRoute** - This checks if the route exists already .

### 6.3.4 Get Route Traffic



The Get Route Traffic use case makes use of the following lower level functions, both within and outside the Routes module to determine the traffic of

a route and send it back to the user:

- **fetchImage** - This function fetches an image from the camera at the specified location.
- **analyseImage** - This function performs image analysis in order to determine the traffic level in that location
- **composeTrafficInfo** - This function composes all the traffic levels that make up the route and constructs a preliminary "traffic info report".
- **sendTrafficInformation** - This function creates a traffic information report and sends it to the user who requested it.
- **storeTrafficLevels** - The traffic levels are stored for any other user to be able to access it if necessary instead of performing analysis again.

## 6.4 Process Specification

This section describes the process for completing the Add Route, Modify Route and Get Route Traffic Sections.

### 6.4.1 Add Route

### 6.4.2 Modify Route

### 6.4.3 Get Route Traffic

## 7 Domain Model

## 8 Open Issues