CERTIFICATION OF APPROVAL

REDUXGO – CONTEXT AWARE MOBILE RECOMMENDER SYSTEM TO REDUCE STOP-AND-GO SCENARIO DURING HIGHWAY CONGESTION

by

Nor Fadhilah Binti Abdul Samat

A project dissertation submitted to the
Business Information System Programme
Universiti Teknologi PETRONAS
in partial fulfillment of the requirement for the
BACHELOR OF TECHNOLOGY (Hons)
(BUSINESS INFORMATION SYSTEM)

Approved:

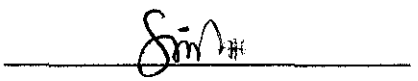
Mrs. Nazleeni Samiha Haron
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

September 2011

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.



Nor Fadhilah binti Abdul Samat

ABSTRACT

This research paper will discuss about current problem that is common to urban citizens in the whole world, which is traffic congestion and the solution on how to overcome it. Emphasized has been given on the traffic congestion that occurs without bottleneck, especially in highway. Study has been made on the Anatomy of Traffic Congestion to identify the main cause of the problem and at the same time identifying possible ways to overcome the situation with the help of current technology tools. Research also has been made on the available options currently being used by public to avoid traffic congestion in the urban area which includes GPS, website portal, radio and others.

Though there are many technology tools that available to help public with traffic congestion, however there are no specific applications that help the user to reduce the Stop-and-Go during traffic congestion. Most of the current applications are emphasizing more on giving current updates to user in order to avoid the congested area. ReduxGO will be developed to help the driver to cope and reduce the congestion by maintaining suitable gaps distance between owner's vehicle and preceding vehicle.

Context-Aware Recommender System (CARS) has been chosen as the base platform to develop special application for mobile phone users. Location, which refers to vehicle's position, has been chosen as the main Context input to be considered in the application. The application will function to recommend suitable speed to the user in order to maintain the appropriate gaps distance between vehicles. This report will discuss in detail regarding the procedure and tools required to develop the application.

The methodology undertaken to conduct the project development will be Extreme Programming Methodology which involve four (4) main phases which are Planning, Analysis, Design and Implementation. This report is presented to deliver the background, problem statements, objectives, scope of study, literature review, and methodology of choice in the development process.

ACKNOWLEDGEMENTS

There were many people's involved throughout completing this Final Year Project, which has contributed to give help, guidance, assistance, motivate, advice, supports, and also supervise. It is a good experience going through all the documentation, presentation and development of a project called ReduxGO – Context Aware Mobile Recommender System to Reduce Stop-and-Go Scenario during Highway Congestion. I would like to express my appreciation and highest gratitude to the individual that have taken the time and effort to assist me in completing the project. Without the cooperation and some motivation of these individuals, no doubt I would have to face some problem throughout the course.

Special thanks go to my supervisor, Ms. Nazleeni Samiha Haron for her guidance, assistance and motivation, without it I would face a lot of obstacle. To both my internal examiner, Dr. Rohiza Ahmad and Ms. Mazeyanti Ariffin, thank you for all the comment and recommendations that been shared for the project improvements. Also thanks to the Final Year II Project Coordinator, Ms. Emelia Akashah Bt Patah Akhir for giving all of final year student all the information especially on timeline and guideline that required along the project of final year project. To the Head and Academic Executives of Computer and Information Sciences Department, thank you for your cooperation and assistance in completing this project. Not to forget, my fellow friends and colleagues who share their knowledge and information in completing the report and development of the project.

My parent:

- Mr. Abdul Samat Bin Daud
- Ms. Hafsah Binti Hussin

To both my parent thank you so much for your understanding upon completing the Final Year Project. All your guidance, support and assistance are very important in completing my study.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

1.1 Background

Rapid changes in modern lifestyle have affected much of our daily basic tasks. People are now tight with busy schedule which oblige them to minimize the time spend on certain task, including basic chores in life such as having lunch, shopping for household goods, paying for utility bills and others. However, with the needs to be fast in most of the task in daily life, modern people nowadays have to face a new headache which is the traffic congestion problem, especially in the urban area.

This research project will basically discussed more on how the traffic congestion occurs and how to tackle the problem by preventing it from getting worse. As far as we are concern, there are many reasons that had causes the congestion to occur. Generally, public would agree that the bottleneck in the traffic congestion occurs because there may be road works, accident or police roadblock that will slow down the traffic because the driver will slow down their car to see on what happen during the particular events. However, the real origin of traffic congestion often has nothing to do with obvious obstructions such as accidents or construction work but is simply the result of there being too many cars on the road [1].

The traffic congestion problem has been severe nowadays, especially when number of vehicles keeps increasing each year. What actually action that can be taken to overcome this problem? Common ways that generally agreed by public are by maximizing the use of public transports, improve the highway infrastructure by providing broad lanes and also practice car pool culture to work. However, rapid development in technologies has given the public with extra option to overcome the situation. There is a need to develop an application that will help driver to cope with

the congestion and decrease the stress level by reducing the Stop-and-Go scenario during highway congestion.

The basic idea of the proposed application is that, it will eventually help the driver especially in urban area where traffic volume is at the maximum level and one cannot simply find a way to reduce it since everybody seems need to have their own vehicle and drive to work. The application will help driver that was trapped in the middle of traffic to control the vehicle speed in order to avoid the Stop-and-Go scenario during traffic congestion. The Stop-and-Go scenario can be overcome by maintaining a suitable distance between each vehicle on the road. Thus, we need an application that will suggest a suitable speed that driver should maintain to overcome the situation, after taking into consideration certain context such as weather, level of traffic congestion, time of the day and others.

Context-Aware Recommender System (CARS) is focusing on recommending the most relevant items to users by taking into account any additional contextual information, such as time, location, or the company of other people [2]. This concept is a combination between context-aware computing and recommender system. Recommender systems (RSs) are information filtering and decision support tools aimed at addressing these problems, providing product and service recommendations personalized to the user's needs and preferences at each particular request [3]. Meanwhile, context-aware computing is a mobile and ubiquitous computing paradigm in which application can discover and take advantage of contextual information such as user location, time of day, computing and communication characteristics, nearby people, objects and devices, user activities and goals, etc. [4]. Such applications are able to adapt their behavior, which is their functionality, content and interface according to user's current situation with minimal intrusion.

CARS is the most suitable research area to be implemented in developing the application to reduce the Stop-and-Go scenario when driver are stuck in the middle of highway congestion. It will function to suggest a suitable speed that driver should maintain in the highway after considering major contextual information that related to the scenes.

1.2 Problem Statement

Malaysia's road transport has increased dramatically over the last two decades. In 1989, the Malaysian population stood at 17.4 million, but has since increased to 28.31 million in 2009. For the same period, the number of registered vehicles has also increased from 4.15 million to 19.02 million [5]. It shows that the total number of vehicles registered in Malaysia increased by 78% for the past two decade.

These huge percentage shows that the total numbers of road transport had boost over the last few years. It will continue to grow especially when the quality of car manufactured also increase which will eventually improve the lifespan of a vehicle. However, the road infrastructure size are not having enough improvement in term of capacity holder, in order to cope with the increasing number of vehicles and that had resulted in traffic congestion in major cities in Malaysia.

The major growth in total number of vehicle in Malaysia shows that there is a need for new alternative to overcome the problem of traffic congestion. Common available application that available to avoid traffic congestion was by using Global Positioning System (GPS) and also real time traffic updates from websites and radio. The majority of existing options focuses on to create early precaution for drivers regarding the congested area which actually does not help directly to solve the congestion, instead it only create a reminder for the user to avoid certain congested route.

Current available application did not focus on how to control the root cause of the congestion to occur, which actually resulted from the Stop-and-Go scenario. Despite of that, public must consider that traffic congestion also increasing pressure on the environment for the emissions of manmade CO2 and NOX. Besides, traffic congestion also lower fuel consumption and cleaner fuels.

1.3 Objectives

The objectives of this project are:

i. To perform a small scale of study on the effect of Stop-and-Go scenario and its effect to highway congestion.
ii. To model, design and develop an application that will be used in avoiding driver to be trapped in highway congestion using context-aware recommender system approach.

1.4 Scope of Study

This research covers on the traffic congestion that was caused by a large volume of vehicles on the Malaysia's highway. Highway has been considered as the main key location for the research on the behavior of traffic congestion because the emergence of traffic congestion in this place occurs even with no traffic light, higher speed limit and also less junction. This will relates to the experiment performed by a group of scientist from University of Nagoya, Japan on the formation of congestion with no bottleneck [1].

The project focuses on determining the suitable vehicle speed to reduce the Stop-and-Go situation during highway congestion. The main input that will be considered in suggesting the appropriate speed to cope with the congestion is the distance between driver's vehicle* and the preceding vehicle. This research will analyze on a few approach to capture the input from the driver's vehicle.

Apart from that, critical review will be conducted to determine the other types of contextual information to be considered in determining output. Such factor as weather does affect the result in recommending the suitable vehicle speed to be taken by drivers. Lastly, the research will also focus on the development of suitable graphical user interface to ease the drivers in maintaining the suitable speed.

CHAPTER 2

LITERATURE REVIEW

2.1 Anatomy of Traffic Congestion

An experiment conducted in 2008 by Japanese scientist from University of Nagoya, Japan has shown the behavior of car movements when they are large numbers of vehicle on the traffic, which will eventually, causes the traffic congestion. The experiment was performed on a circuit to show the emergence of a jam with no bottleneck. In the initial condition, all the vehicles are moving, homogeneously distributed on the circular road, with the same velocity [1].

Figure 1: *A snapshot of the experiment on a circular road. The circumference is 230 m, and the number of vehicles is 22. A 360-degree video camera is situated at the center of the circle for measurement [1].*

The researchers had used a circular track with a circumference of 230m. They put 22 cars on the road and asked the drivers to go steadily at 30km/h around the track. While the flow was initially free, the effect of a driver altering his speed reverberated around the track and led to brief standstills. The problem is that when the distance between cars varies just a fraction, the smallest responses by drivers create a domino reaction with tiny changes in speed having a cumulative effect.

So when the in front driver slows to maintain his speed, the drivers behind will brakes a little, as does the next driver behind him. This scenario continues until the other drivers few meters back finds that their vehicles are at a standstill.

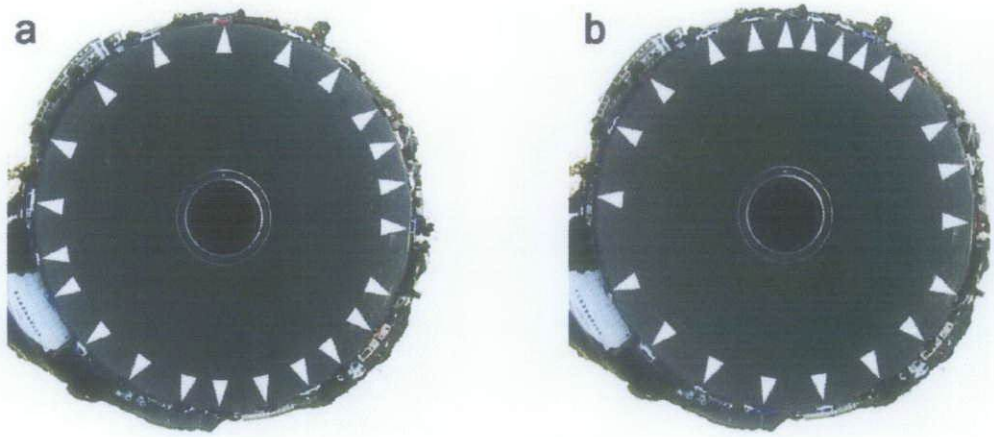


Figure 2: *The vehicles move along the circle. The pictures are taken by the 360-degree video camera. The position of each vehicle is shown by a triangle: (a) The snapshot at the initial stage shows that the vehicles move as the free flow. (b) The snapshot 3 min later shows that a jam has been formed, consists of five vehicles, which is seen at the top of the circle at this moment. This picture corresponds to Figure 1. (A jam is observed on the upper right corner of the circle).*

The scientists refer to the phenomenon as a shockwave. As soon as distance between vehicles begins to vary, each driver in turn must adapt his speed or distance from the other cars, and the necessary adjustment increases as it goes back from car to car. Yuki Sugiyama, physicist from Nagoya University, said, "Although the emerging jam in our experiment is small, its behavior is not different from large ones on highways. When a large number of vehicles, beyond the road capacity, are successively injected into the road, the density exceeds the critical value and the free flow state becomes unstable." [1].

According to Traffic Flow Theory, shock waves that occur in traffic flow are very similar to the waves produced by dropping stones in water [6]. A shock wave propagates along a line of vehicles in response to changing conditions at the front of the line. Shock waves can be generated by collisions, sudden increases in speed caused by entering free flow conditions, or by a number of other means. Basically, a shock wave exists whenever the traffic conditions change.

2.2 Current Technology Tools to Cope With Traffic Congestion

There are basically several applications or software that has been developed in order to help drivers to cope with the traffic congestion. The most popular option that was being used by drivers is Global Positioning Systems (GPS). GPS navigator functions to inform the driver whenever there is traffic congestion on the way to their destination. It also helps the drivers by suggesting another route direction to bypass the congestion.

Another available option that available for the driver to cope with traffic congestion is the real time traffic update from website. In Malaysia, there are several websites that provide the real time update regarding the traffic flow condition. Among of them are *Lembaga Lebuhraya Malaysia* (LLM) [7] and Integrated Transport Information System (ITIS) by *Dewan Bandaraya Kuala Lumpur* (DBKL) [8]. However, these websites only provides the information for certain major route in *Klang* Valley area. Likewise, major local radios also provide update on the traffic flow on major cities across the country. User will need to keep tune to the radio to get the latest updates.

Adaptive Cruise Control (ACC) is an optional cruise control system appearing on some more upscale vehicles. These systems use either a radar or laser setup allowing the vehicle to slow when approaching another vehicle and accelerate again to the preset speed when traffic allows. It detects the presence of a preceding vehicle and measures the distance as well as relative speed using a forward-looking sensor, and it automatically adjusts the vehicle speed to keep a proper range [9]. ACC technology is widely regarded as a key component of any future generations of intelligent cars.

Based from the available tool options for driver to cope with the congested traffic, GPS and real time traffic updates from websites did not really solve the root cause of the congestion to occur, instead it only provide a few ways for the driver to avoid the congested area. Meanwhile, the ACC system is only applicable for car produced by certain car manufacturer. The research on this project will emphasize on the mobility and ease of use of the system to benefits larger scale of mobile phone users.

2.3 Context Aware Recommender System

The importance of contextual information has been recognized by researchers and practitioners in many disciplines, including e-commerce personalization, information retrieval, ubiquitous and mobile computing, data mining, marketing, and management. While a substantial amount of research has already been performed in the area of recommender systems, the vast majority of existing approaches focuses on recommending the most relevant items to users and does not take into account any additional contextual information, such as time, location, weather, or the company of other people.

The context definition from the mobile and ubiquitous computing point of view is any information that can be used to characterize the situation of an entity. An entity is a person, place or object that is considered the relevant to the interaction between a user and application [10]. ReduxGO, a Context-Aware Recommender System, will functions to suggest suitable speed for drivers when they are trapped in the middle of traffic congestion in the highways. The types of contextual information that will be consider into accounts, which includes the user location, time of the day, season of the year, traffic conditions, and temperature.

2.4 Possible Methods to Capture Input

Important key research item for this project lies on the needs to capture the appropriate input in order to come out with correct speed to be maintained while driving. ReduxGO will require the correct measurement of distance between the owner's vehicle and the preceding vehicle. There are various methods that can be used in order to measure a distance between user's vehicle and the vehicle in front, which includes camera applications in mobile phone, Global Positioning System (GPS), forward-looking sensor, vision-based car distance measuring system and others.

2.4.1 Global Positioning System (GPS)

The Global Positioning System (GPS) is a worldwide radio-navigation system formed from a constellation of 24 satellites and their ground stations. It has been widely used to determine position and calculate distance from one location to another. GPS determines the location by indentifying how long it takes a satellite signal to reach a receiver, which generates its own signal.

Assuming that the signals are synchronous, GPS compares the satellite signal's pseudorandom number code—a digital signature unique to each satellite—with the receiver's PNC to determine the signal's travel time. The system multiplies this value by the speed of light to compute the satellite's distance from the receiver [11]. GPS way of distance estimation depends on the longitudinal between two different location. It requires both locations/objects to have GPS signal.

2.4.2 Distance Measurement Using License Plate Character Height

Researchers from Faculty of Engineering, Kingston University, London, had made a research to develop a system that can capture distance between two vehicle using license plate character heights. With the help of an image preprocessing stage, the region of interest (ROI) in the acquired images is identified.

The ROI is then examined by a rule-based algorithm that identifies the characters in the plate and computes the corresponding height of the plate characters and thus the distance between the cars [12]. However, this system involved complex image identification system where it requires the use of special camera which has the capability of CCD/CMOS Image Sensor.

2.4.3 Smart Measure

Meanwhile, another useful application that was available for android smart phone is Smart Measure. Smart Measure is an application developed for android mobile phone that can be used to measure object distance, height and direction [13]. Using this application, user can simply measure the distance by specifying the height and

aim the camera cursor at the specified object. User need to tilt the camera in order to get proper angle. The application will then calculate the distance by using simple trigonometry, as shown in the **Figure 3** below:

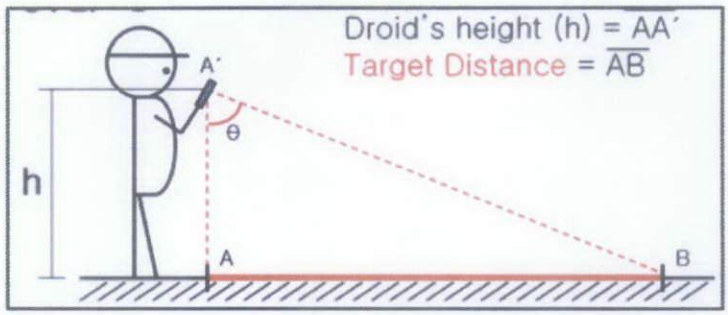


Figure 3: *By using the height of the camera above the ground, and the angle which user need to tilt the unit so that a crosshair touches the base of the object, Smart Measure can calculate the distance to the object [13].*

Figure 4: *User will need to aim the cursor on the bottom of the object (as shown in the picture) and tilt the camera to get proper angle before capturing the image, in order for the application to calculate the distance.*

Smart Measure is practical enough to help user measure the distance because it only require user to specify the height of the target object and tilt the phone to get the appropriate angle. However, if taking into consideration the same way of system implementation to ReduxGO, it is impossible for the driver to use the application while driving because it require user to tilt the phone to get proper angle for calculations. Some modifications on the use phones angle need to be done in order to implement the same concept of measuring distance in ReduxGO.

2.4.4 Smart Distance

Another useful tool that available in android market to measure distance is the Smart Distance. This range-finder application measures the distance to a target by a camera perspective [14]. The effective distance is 10m-1km. To measure the distance, user will need to know the height of the target. Then, the application works with the help of touch on the screen, which require user to specify the target object's range between two (2) horizontal lines. The system will then calculate the distance between user and target object using image-height percentage comparison.

Figure 5: *By using touch screen help, user will need to specify the target's range between two (2) horizontal lines, as shown in the Figure.*

The concept of Smart Distance application can be implemented in ReduxGO, which will refer to the way of how the application processed the input to generate the distance. Some modification will need to be made on the capturing input process, in context of ReduxGO's user that will involve car drivers, where user cannot manually specify the specified range of the two (2) horizontal lines while driving. ReduxGO will maintain a fixed range of two (2) horizontal lines, which will refer to the average height of the medium-sized car from the camera perspective. The camera will capture the image automatically, whenever there are object identified in the specified region. User will just need to key in the height of the phone position inside the car once, and ReduxGO will recommend the suitable speed, whenever applicable.

2.4.5 VIO – Motion Detection Application for Windows Mobile

VIO is a new motion detection application which uses the primary camera of any phone running on Windows Mobile 5.0 or later to detect movement in a should-be stationary location. It can be set to run while the owner is away from home, sending an image with file type of .jpg of the moment in which movement is detected in an email to any given address, and sending an SMS alert, in the case of the sensor being triggered [15]. This motion detection was applied by comparing the actual picture with one reference picture that will be automatically refreshed within specified period. There are two (2) ways of comparison which are Squares and Edges. Using Squares method, the picture will be cut to squares with specified square's width. The developer will then specify the squares sensibility and minimum squares needed to trigger an alert.

Meanwhile, the second method which is Edges, is more accurate but slower. This method will create some zones based on difference between actual and reference picture. Then, these zones will be extracted one by one. The application will then identify the number of pixels in each zone. If the zone contains the minimum pixels needed (as specified by developer), the application will then trigger an alert.

This motion detection application is suitable to be implemented in ReduxGO as it will help the application to detect the presence of the preceding car in order to capture the distance and calculate the recommended speed. Motion detection will help the system to detect the exact position of the car from the camera view, which is crucial, in order to get the correct angle for distance calculations.

2.5 Mobile Recommender System Architecture

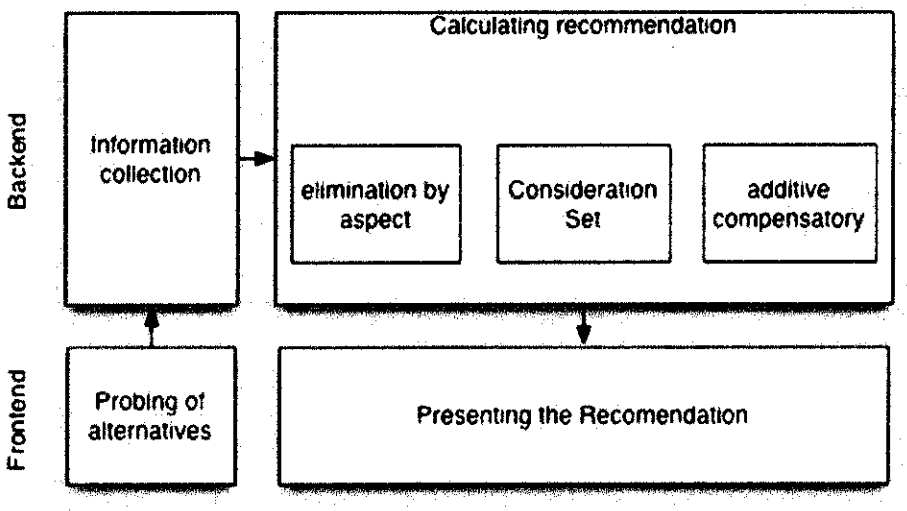


Figure 6: *A generic architecture for Mobile Recommender System* [16]

2.5.1 Probing of alternatives in the environment

Scanning the alternatives in the new environment can be done in many different ways. There are four basic approaches according to directed versus undirected "scanning" of objects and the exchange of an identification tag (ID) by an (active) object or by the mobile device.

The identification of the object is triggered by a scanning device or by the object itself. One can imagine the products being scanned by a user pointing the mobile device at them. Alternatively, the products can be probed as the device comes into a certain vicinity of the physical location of that product, without the user actually pointing to anything [17].

2.5.2 Information collection

Information about the scanned product alternatives may not be available in the current environment. To complete the information required for alternative evaluation, access to additional product information, typically available at central databases, may be necessary. The codification of product information in a particular

identification tag does not seem feasible given limited data. The scanned tag will be transmitted to a product database, which in turn transmits detailed product information to the recommender system.

2.5.3 Calculating the recommendation

In combination with the personalization of the recommender system (user preferences for alternative evaluation) the scanned products are evaluated and compared according to their detailed product information. This element of the recommender system represents the underlying decision capabilities. The composition of the consideration set and the subsequent evaluation and ranking of alternatives are the result of the configuration of the recommender system by user preferences.

2.5.4 Presenting the recommendation

The presentation of the recommendation can be done in at least three ways: visual, acoustic, and tactile [18]. Fortunately, many mobile devices have the option to provide information on a small screen, to produce sound, and to tremble. The last option is usually implemented in mobile phones as a way of providing a silent phone alert. The visual way of presenting information is the richest of the three. Information can be provided by color codes, numerical codes, or text. Color codes can be made lighter, and darker, providing even more ways of communicating recommendations. Acoustic ways of communicating recommendations are also possible. One way to do it would be to create a low pitch for unattractive alternatives, and a high pitch for attractive alternatives.

Another way is to produce intermittent signals, or beeps, and to increase the frequency of the signals as the products become more attractive. This is similar to the way that acoustic signals are implemented in radars. Yet another way is to provide speech based on the recommendation value [16].

CHAPTER 3

METHODOLOGY

The project adopts Extreme Programming Methodology, which is an example of agile development methodologies. An Agile Development Methodologies focus on streamlining the System Development Life Cycle (SDLC) by eliminating much of the modeling and documentation overhead and the time spent on those tasks. Instead, projects emphasize simple, iterative application development. The agile development approach typically is used in conjunction with object-oriented programming [19].

The project's methodology is based on iterative development, prevalent in the development stage. This stage covers feasibility study, planning, analysis, design and implementation. Agile development is chosen to accommodate requirements changes and frequent adaptation to alternative designs and revised models. The developed modules from the iterations will be evaluated, inspected and further enhancements will be made, if needed.

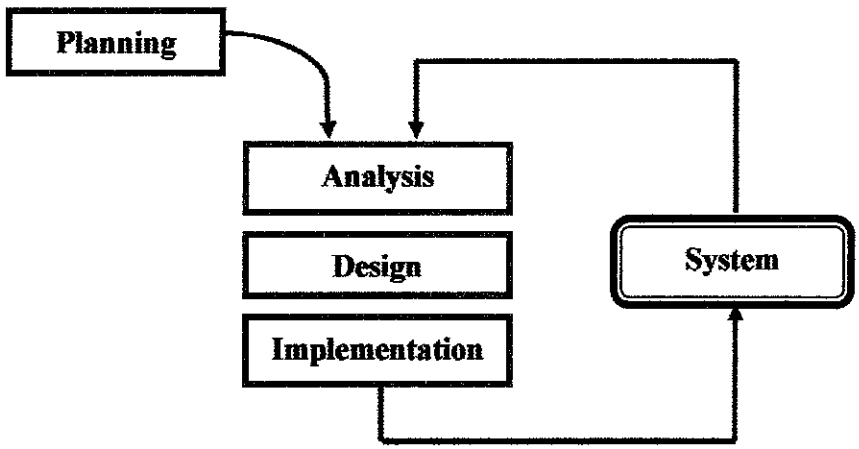


Figure 7: *The Extreme Programming Methodology* [19].

3.1 Phase 1: Project Initiation and Planning

This is the first phase of the system development life cycle where the initial scope of the entire project is identified. The project is initiated with a detailed background study on the Anatomy of Traffic Congestion. A brief study has been conducted to identify the main cause of traffic congestion with no bottleneck. Based from the research, author has come out with the solution to create an application to recommend suitable speed to the user in order to maintain appropriate gaps distance.

Research on the related works was done to come out with proper idea on identifying the required input and related process to generate the output. The observation also being made on the available Android Application in order to get the idea of the common functionalities and how the application capture the input in order to measure the distance.

A rough timeline for the project was developed as well as a draft of the scope. It has been estimated about 1 month for data gathering and research, 3 months for system development and 1 month for system testing and modification. Deliverables for this phase are the problem statement, the project scope, the objectives of this project and the project work plan.

3.2 Phase 2: System Analysis

In this phase, an analysis has been made to identify the requirements that the application need to meet in order to address and resolve the problems identified in Phase 1. The techniques used in defining the requirements are through analysis on the current application's capabilities that available in the market and also by conducting interviews with the potential users on their expected outcome. The outputs for this phase are the requirements analysis and system's activity diagram.

3.3 Phase 3: Technical Design

3.3.1 Architecture Design

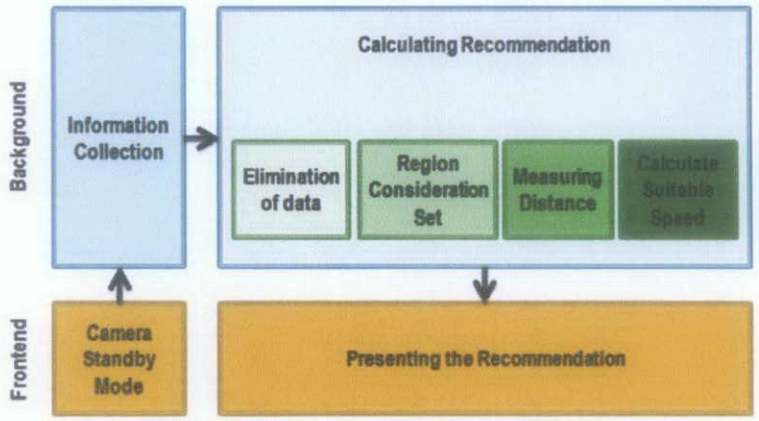


Figure 8: *ReduxGO System Architecture*

Architecture design deals with the system flow of ReduxGO mobile application. This is very important since all of the conceptualize data gathered before this were put into actual physical model. The architecture is divided into two (2) main block which is Frontend Block – which refer to the applications procedure that involve interactions with user. Meanwhile, Background block refer to the back-end application procedure that will function to process the input in order to get the required output.

3.3.2 Interface Design

User interface design is the design of computers, appliances, machines, mobile communication devices, software applications, and websites with the focus on the user's experience and interaction. Unlike traditional design where the goal is to make the object or application physically attractive, the goal of user interface design is to make the user's interaction experience as simple and intuitive as possible.

Safety factors have been taken into consideration in designing the ReduxGO interface to ensure that users, which are car driver, can use it well without any distractions and give them perfect assistance in managing their car speed. Details of interface design will be discussed in Chapter 4: Result & Discussion.

3.4 Phase 4: System Implementation & Testing

The coding activities took place in this phase after all of the requirements needed had been gathered during the execution of the previous phases. The coding activities were based on the diagrams which were the deliverables from each of the earlier phases. The system will be put on test in order to ensure the system performance conforms to the requirement of the system. System testing took place after the system had been fully developed. Any adjustments were made to the system if necessary.

3.4.1 System Implementation

The fourth phase consists of construction phase where the development of the working application being made on a regular, incremental basis. Details on the tools required for development phase are as follows:

- Android Software Development Kit (SDK)

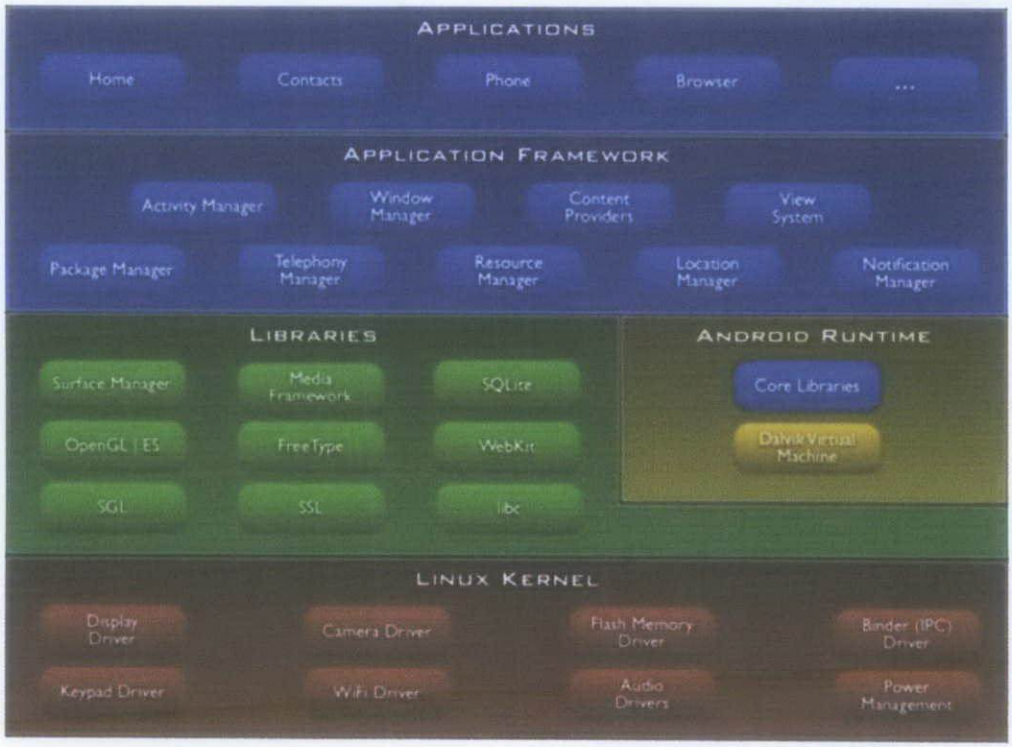


Figure 9: *Android Software Development Kit (SDK) Content Overview*

The Android SDK provides the tools and libraries necessary to begin developing applications that run on Android-powered device [20]. It enables developers to create applications for the Android platform. The Android SDK includes projects with source code, development tools, an emulator, and required libraries to build Android applications [21]. Applications are written using the Java programming language and run on a custom virtual machine designed for embedded use.

- Eclipse Integrated Development Environment (IDE)

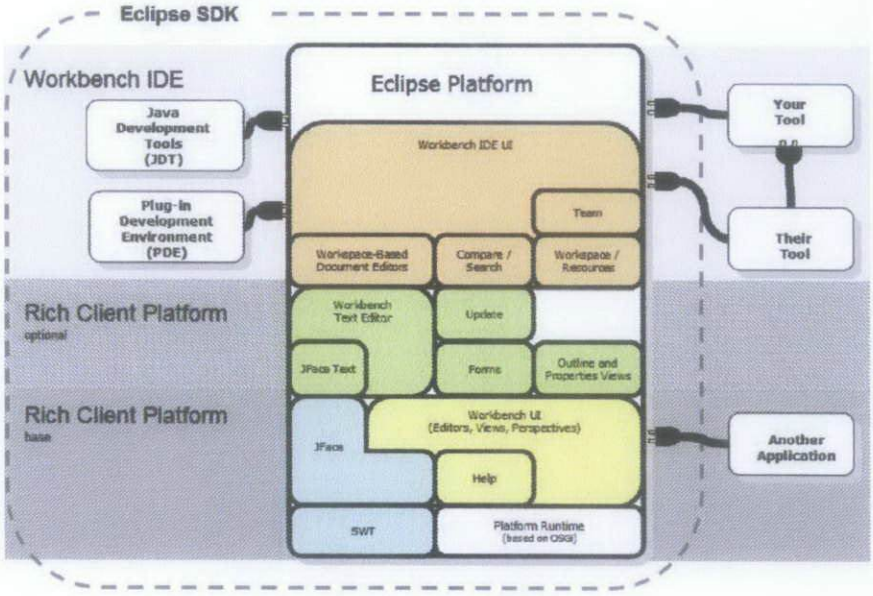


Figure 10: *Eclipse IDE Architecture*

Eclipse is an open source community, whose projects are focused on building an open development platform comprised of extensible frameworks, tools and runtimes for building, deploying and managing software across the lifecycle [22]. Eclipse has been chosen as the suitable IDE to develop an android-based application because Google has created an Android Development Tools (ADT) for Eclipse plug-in.

ADT will extends the capabilities of Eclipse to help user quickly setup a new Android projects, create an application UI, add components based on the Android Framework API, debug the applications using the Android SDK tools, and even export signed .apk files in order to distribute the application [23].

- Open Source Computer Vision (OpenCV)

Figure 11: *OpenCV Logo*

Open Source Computer Vision is a library of programming functions for real time computer vision. There are various features provided by OpenCV that is useful for android application development. Among the features are image data manipulation, basic image processing, camera calibration, motion analysis, object recognition, image labeling and many more [24].

OpenCV capabilities to perform real-time image processing is important for ReduxGO development especially during the Region Consideration Set (*refer Figure 9*), where the application needs to identify the location of the preceding car by using object recognition feature, in order for the application to measure the distance and finally calculate the recommended speed.

3.4.2 System Testing

This last phase includes validation for quality assurance. The system will later be tested to ensure it performs as per designed. Testing is one of the most critical stages. Iterations of Phase 2 and Phase 3 may be necessary to introduce fixes in the optimization mechanism before the final solution can be released. Continuous integration and enhancements are applied in Phase 4.

The main purpose of prototype testing is to check whether this application meets the requirements and achieve the objective of the project. The testing phase will be conducted using simulation approach on the prototype circuit. Two (2) mini control cars will be used as car prototype. An android smart phone will be placed on the first car. The diagram is as follows:

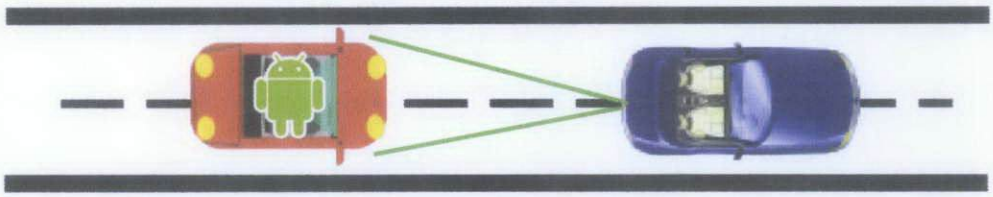


Figure 12: *Simulation Circuit from the Top view*

CHAPTER 4
RESULTS & DISCUSSION

This chapter will focus relatively with methodology chapter, where the result of all the diagrams and interface been mentioned earlier become outputs here. Most of the content in this chapter will comprise of the research findings to overcome the problem statement and solve the project objectives.

4.1 ReduxGO – System Analysis

Based on the literature review research, it is concluded that the Stop-and-Go scenario during highway congestion can be reduced by maintaining a safe distance between the driver's car and preceding vehicles. The application must be able to determine the suitable distance and recommend it to the car driver since they cannot estimate the exact distance using manual approach.

4.1.1 Requirements Analysis

Findings in requirements analysis part will specify the user's expectation of the application. This phase involves interview with expected users to determine specific feature expectations, resolution of conflict in requirements as demanded by the various users in order to avoid requirement creep. It is important to ensure that the final system conforms to user needs rather than attempting to mold developer expectations to fit the requirements. Details of requirement analysis for ReduxGO:

- System boundaries - ReduxGO application must be able integrate with driver situation in the car, which will provide a mobile help on the road to avoid Stop-and-Go scenario during highway congestion.
- The application must be able to measure the distance between owner's vehicle and one preceding vehicle.
- The application must deliver the speed output in most appropriate way to suit with driver's condition.

4.1.2 ReduxGO Activity Diagram

Activity diagram has been designed as the result of the requirement gathering to illustrate the process flow and functions of ReduxGO application.

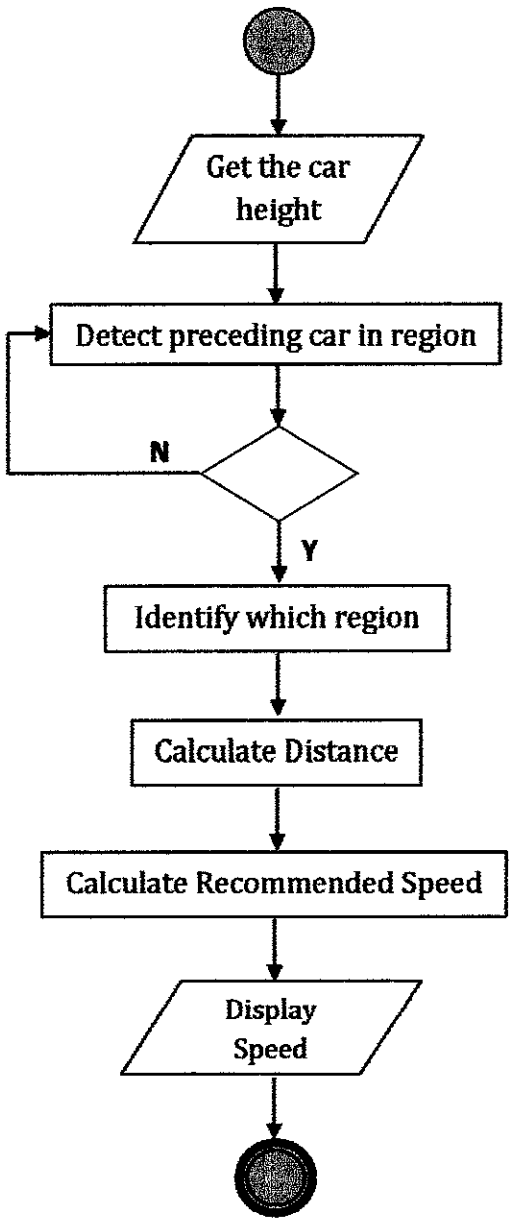


Figure 13: *ReduxGO Activity Diagram*

4.2 Technical Design

4.2.1 ReduxGO Flowchart

ReduxGO will take Location as the main Context to be considered in developing the application. It will require the application to measure distance between the locations of owner's car and preceding vehicle in order to recommend the suitable speed. The procedure to capture and process the input in order to come out with the speed output will be explained by the diagram above.

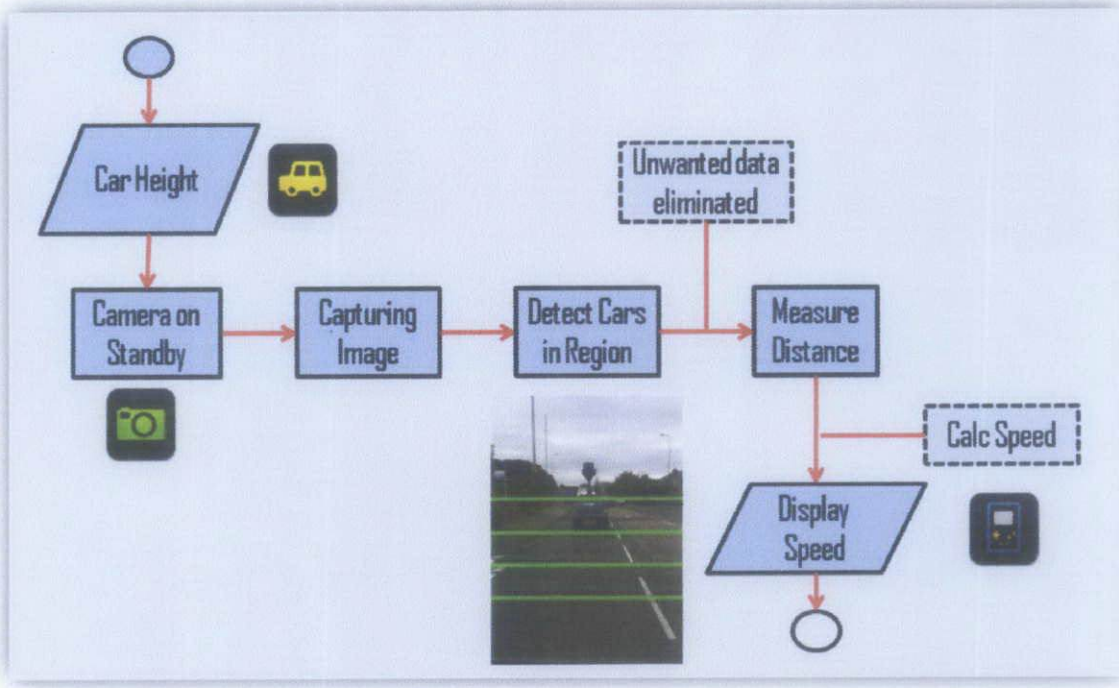
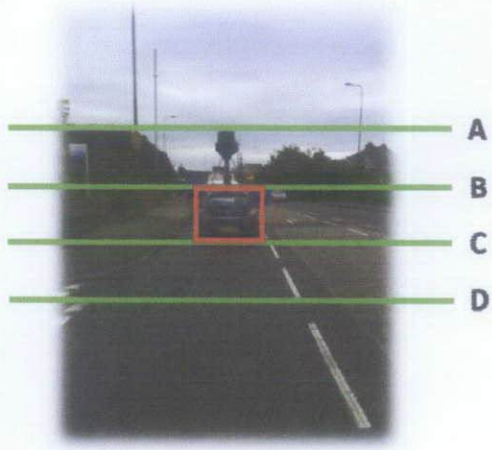


Figure 14: *ReduxGO Process Flow*

4.2.2 ReduxGO Prototype Interface Design

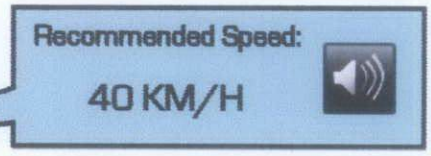
ReduxGO will have a simple interface where the development of the application will focus more on the background functions. The interface was designed to support the Frontend Block of the system architecture, which is the interaction between user and the application and maximize the usage of the prototype functions. There is only minimal involvement of user in this application where user just needs to activate the application and wait to receive the recommended speed.

i. Detecting Preceding Car in the Region Using Camera Input



ii. Display the Recommended Speed

The recommended speed will be delivered to the user by displaying the figure on the bottom level of the user interface followed by verbal instruction using computer voice.



4.3 System Implementation

System implementation will focus on how the application development to support the Background Block functions of system architecture. Details of the main functions in the background block are as follows:

4.3.1 Detect Preceding Car in the Region

This is the most important functions of ReduxGO where the application needs to detect the image of preceding car from the camera input. By using Open Source Computer Vision (OpenCV) Library as main tool to detect object through real-time image, there are several useful features. Those features in the library are able to perform object detection and matching for different type of image such as affine transformed images, scale changed images, rotation images, blurred images and illumination changed images.

For ReduxGO application development, it will require a suitable feature detection that can serve a scale changed images – which refer to the different size of cars when there are at different distance from the user's vehicle. The more distance between user's car and preceding vehicle, the smaller is the size of the car image. Speeded Up Robust Features (SURF) has been considered the most suitable for real-time image processing for object detection because of it robustness and speed compared to others features.

A set of SURF key points and descriptors can be extracted from an image and then used later to detect the same image. SURF uses an intermediate image representation called Integral Image, which is computed from the input image and is used to speed up the calculations in any rectangular area [25]. It is formed by summing up the pixel values of the x, y co-ordinates from origin to the end of the image. This makes computation time invariant to change in size and is particularly useful while encountering large images. The SURF detector is based on the determinant of the Hessian matrix.

The SURF descriptor describes how pixel intensities are distributed within a scale dependent neighborhood of each interest point detected by Fast Hessian Object detection using SURF is scale and rotation invariant which makes it very powerful. Also it doesn't require long and tedious training as in case of using cascaded Haar classifier based detection.

```
#include <stdio.h>
#include <opencv2/features2d/features2d.hpp>
#include <opencv2/highgui/highgui.hpp>
#include <opencv2/imgproc/imgproc_c.h>

using namespace std;
int main(int argc, char** argv)
{
CvMemStorage* storage = cvCreateMemStorage(0);
cvNamedWindow("Image", 1);
int key = 0;
static CvScalar red_color[] ={0,0,255};
CvCapture* capture = cvCreateCameraCapture(0);
CvMat* prevgray = 0, *image = 0, *gray =0;
while( key != 'q' )
{
int firstFrame = gray == 0;
IplImage* frame = cvQueryFrame(capture);
if(!frame)
break;
if(!gray)
{
image = cvCreateMat(frame->height, frame->width, CV_8UC1);
}
//Convert the RGB image obtained from camera into Grayscale
cvCvtColor(frame, image, CV_BGR2GRAY);
//Define sequence for storing surf keypoints and descriptors
CvSeq *imageKeypoints = 0, *imageDescriptors = 0;
int i;

//Extract SURF points by initializing parameters
CvSURFParams params = cvSURFParams(500, 1);
cvExtractSURF( image, 0, &imageKeypoints, &imageDescriptors, storage,
params );
printf("Image Descriptors: %d\n", imageDescriptors->total);

//draw the keypoints on the captured frame
for( i = 0; i < imageKeypoints->total; i++ )
{
CvSURFPoint* r = (CvSURFPoint*)cvGetSeqElem( imageKeypoints, i );
CvPoint center;
int radius;
center.x = cvRound(r->pt.x);
center.y = cvRound(r->pt.y);
radius = cvRound(r->size*1.2/9.*2);
cvCircle( frame, center, radius, red_color[0], 1, 8, 0 );
}
cvShowImage( "Image", frame );

cvWaitKey(30);
}
cvDestroyWindow("Image");
return 0;
```

The explanation of the code is straightforward. It captures a frame from the camera, and then converts it into grayscale because OpenCV SURF implementation works on grayscale images. The function cvSURFParams sets the various algorithm parameters. The function cvExtractSURF extracts the key points and descriptors into the corresponding sequences. Now circles are drawn with these key points as center and the strength of descriptors as the radius.

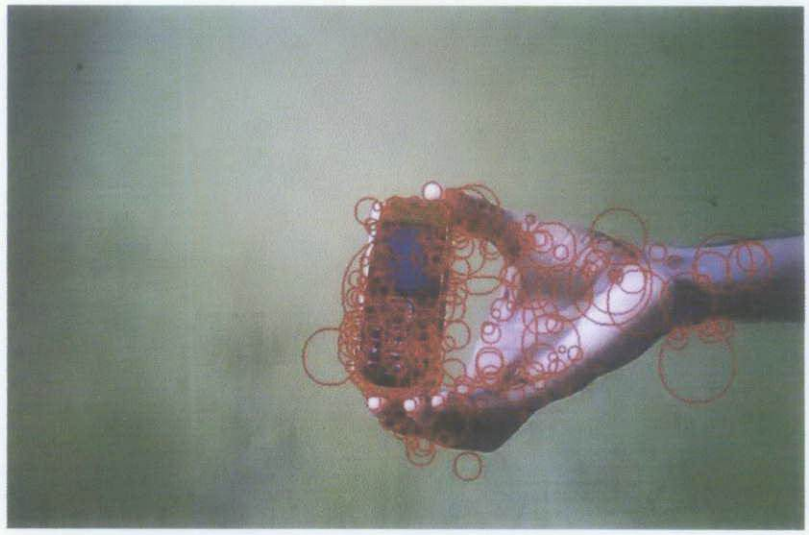


Figure 15: *SURF Key Points Detection*

The above image shows the SURF key points being identified while a person is holding a mobile phone. The background wall has no strong intensity variations and hence no key points exist. The same scenario applies to the image taken from inside the user's car where the background is basically a road and the main object is the preceding vehicles.

From the image above, it can be clearly seen that SURF key points are those pixels whose average intensity values differ much greatly from the immediate neighbors, and by using the descriptor, a relation between the reliable key points and the neighboring pixels are obtained. Once the SURF descriptors and key points of two images are calculated, then they can be compared using one of the many algorithms like nearest neighbor, k-means clustering etc.

4.3.2 Calculate Distance

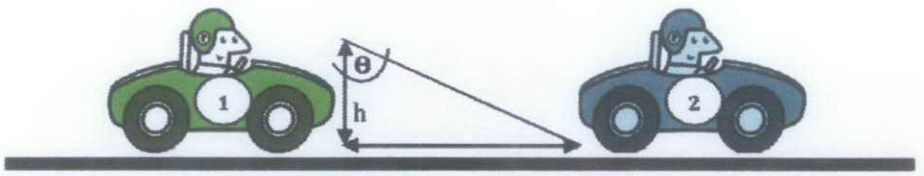


Figure 16: *Apply Trigonometry function to calculate distance*

The second important function in the development phase of ReduxGO is to measure the distance between user's car and preceding vehicle. Based from the car image detection from the camera input, it will belong to different region which represent different distance. The distance calculation will be done using simple trigonometry calculations. The height of the camera position in the car will be inserted by user during installation. The degree, which is θ, will be measured using accelerometer's function that available on the smart phone (refer 2.4.3). Figure 11 shows how the simple trigonometry being illustrated to calculate and get the distance between Car 1 and Car 2.

$$\text{Tan } \theta = \frac{\text{Distance}}{\text{Height}}$$

4.3.3 Calculate Recommended Speed

In order to calculate the recommended speed, ReduxGO would take into account the suitable distance that need to be maintained by driver and compared it with the current distance between the vehicles (as being calculated earlier). If the current gaps distance is smaller than the required distance, the driver needs to reduce the speed. However if the current gaps distance is larger than the required distance, this application will not issue any recommendations. The recommended speed will differ according to the region in which the preceding car belongs.

The recommended speed was presented using two approaches which are:

i. Display the speed figure on the user interface
ii. Verbal instruction using computer voice

4.3.4 Text-to-Speech Function

In other to maximize the effectiveness of ReduxGO application, there is special function created to give verbal command to the user. This is important to comply with the user condition, where they are driving while using the application. The verbal instruction will give a verbal command to the user on the recommended speed. It will function to create extra alert and at the same time didn't want to distract the driver's attention by keep looking at the phone interface in order for them to get the correct speed for their vehicles. The portion of text-to-speech function that has been implemented is as follows:

```
protected void onActivityResult(int requestCode, int resultCode, Intent
data) {
        if (requestCode == MY_DATA_CHECK_CODE) {
        if (resultCode == TextToSpeech.Engine.CHECK_VOICE_DATA_PASS) {
            // success, create the TTS instance
            tts = new TextToSpeech(this, this);}
        else {
        // missing data, install it
        Intent installIntent = new Intent();
        installIntent.setAction(TextToSpeech.Engine.ACTION_INSTALL_TTS_DATA);
         startActivity(installIntent);}
        }
}

public void onInit(int status) {
        if (status == TextToSpeech.SUCCESS) {
            Toast.makeText(ReduxGOActivity.this, "Text-To-Speech engine is
             initialized", Toast.LENGTH_LONG).show();}

        else if (status == TextToSpeech.ERROR) {
            Toast.makeText(ReduxGOActivity.this, "Error occurred while
             initializing Text-To-Speech engine", Toast.LENGTH_LONG).show();}
        }
}
```

4.4 Technology Acceptance Model (TAM) Survey

In order to measure the level of user acceptance of the new application – ReduxGO, a survey has been conducted to the prospective users, which are the car drivers. The survey questions were developed based on the Technology Acceptance Model. This model has been chosen as the main reference to develop the survey questions because it contains key indication in determining the level of user acceptance to a new technology application.

4.4.1 Technology Acceptance Model (TAM)

Technology acceptance model was first introduced by Davis [26]. TAM was an adaptation of theory of reasoned action (TRA) and described the user attitude toward a new application technology. It was determined by two particular beliefs; perceived usefulness (PU) and perceived ease of use (PEOU). The attitude (ATT) is in turn leads to behavioral intention (BI) to use or accept the technology, and then generate the actual usage behavior.

A specific application system is perceived to be usefulness by prospective users if they believe the system will improve or facilitate their job performance within an organizational context. The system is perceived to be ease of use if the prospective user believes that the use of the system will be free of effort.

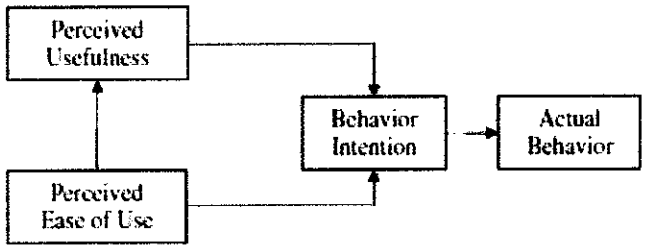


Figure 17: *Technology Acceptance Model*

4.4.2 Survey Results

The survey comprised of fourteen (14) questions where it belongs to four (4) main variables which are perceived usefulness, perceived ease of use, behavioral intention and attitude. The question items were measured by a 5-point scale anchored by "Strongly Disagree" (1) to "Strongly Agree" (5). The survey results are as follows:

Question Items	Question Average	Variable Average
Perceived Usefulness (PU)		
PU1: I believe the use of ReduxGO application would help to maintain safe distance between my car and preceding vehicle.	4.18	
PU2: I believe the use of ReduxGO application during my driving would help to increase the level of safety by controlling my car speed.	4.27	**4.0**
PU3: I believe the use of ReduxGO application would reduce the Stop-and-Go scenario during highway congestion.	3.73	
PU4: Overall, I believe ReduxGO application useful for my driving.	3.82	
Perceived Ease of Use (PEOU)		
PEOU1: Learning to use ReduxGO application is easy for me.	4.00	**3.9**
PEOU2: It is easy to use ReduxGO application while I am driving.	3.73	
PEOU3: Overall, I believe ReduxGO application is easy to use.	4.00	
Behavioral Intention (BI)		
BI1: I will use ReduxGO application on a regular basis in future.	3.82	
BI2: I will frequently use ReduxGO application in the future.	3.91	**3.9**
BI3: I will strongly recommend others to use ReduxGO application.	4.00	
BI4: Overall, I intend to use ReduxGO application during my driving.	3.82	
Attitude (ATT)		
ATT1: In my opinion, it is desirable to use ReduxGO application.	4.09	**4.1**
ATT2: I think it is good for me to use ReduxGO application.	4.09	
ATT3: Overall, my attitude towards ReduxGO is favorable.	4.18	

4.4.2 Survey Results Analysis

Perceived Usefulness (PU)

Based on the survey result, the average value for perceived usefulness variable is **4.0/5.0** which represent that the car users agree that ReduxGO application will be useful or facilitate their driving in order to maintain safe distance between preceding vehicles and at the same time reduce the Stop-and-Go scenario.

Perceived Ease of Use (PEOU)

According to the survey, it shows that prospective users had agreed, with variable average value of **3.9/5.0**, that the use of ReduxGO application will be free of effort.

Behavioral Intention (BI)

The behavioral intention will verify whether the user accept the usage of new ReduxGO application. Based on the survey, it is been proven that most of the user have rated the average value of **3.9/5.0**, which represents their actual usage behavior towards ReduxGO application.

Attitude (ATT)

Attitude of a user toward a system was a major determinant of whether the user will actually use or reject the system. Based on the survey result, the average variable value is **4.1/5.0**, which proven that most of the users agree with the usage of ReduxGO application in order to reduce Stop-and-Go scenario during highway congestion.

CHAPTER 5
CONCLUSION & RECOMMENDATION

5.1 Conclusion

The aim of this project was to study on Stop-and-Go scenario and its effect to the highway congestion. Based from the research, it is proven that Stop-and-Go scenario has become the root cause of the highway congestion. Traffic Flow Theory has indicate that shock waves, which refer to the Stop-and-Go scenario, that occur in traffic flow are very similar to the waves produced by dropping stones in water [6]. A shock wave propagates along a line of vehicles in response to changing conditions at the front of the line. Basically, a shock wave exists whenever the traffic conditions change.

According to the hypothesis, if we can reduce the Stop-and-Go scenario during traffic congestion, we can overcome the root cause of the congestion and create a smooth journey for the drivers. ReduxGO has come out with the solution by creating an alert using recommender system approach to recommend the suitable speed to the car driver in order to maintain safe distance between their preceding car.

It also has created an alternative to help car driver to be aware about their distance between the preceding car and educate them to be more conscious on the safety factor. At the same time, ReduxGO help to reduce the stress level faced by the car driver whenever they are trapped in the congestion area. In the meantime, ReduxGO can help the driver to maximize the usage of their vehicle's oil consumption by avoiding the Stop-and-Go scenario.

5.2 Recommendation and Future Enhancement

There are several improvements that can be made to the ReduxGO application by taking into consideration other context-aware factor such as user's car speed in order to recommend a more accurate speed to be taken. Further research can be made to study on the time taken by the car to reduce their speed. This will avoid a sudden deceleration by the user's car that might create inconvenience to other drivers.

Several external factors such as weather should also taken into account because different types of weather would allow different speed to be taken by the driver since the roadways might be wet during rainy days and it is dangerous for the driver to take a high speed.

APPENDIX I

Project Planning Gantt Chart FYP1

ReduxGO: Context-Aware Mobile Recommender System to Reduce Stop-And-Go Scenario during Highway Congestion

WBS	Tasks	Start	End	Duration (Days)	% Complete	Working Days	Days Complete	Days Remaining
1	**Stage 1: Proposal & Approval**	31-Jan-11	11-Feb-11	12	100%	10	12	0
1.1	Submit Project Proposal	2-Feb-11	2-Feb-11	1	100%	1	1	0
1.2	Get Approval on the Project Topic	11-Feb-11	11-Feb-11	1	100%	1	1	0
2	**Stage 2: Extended Proposal**	12-Feb-11	4-Mar-11	21	100%	15	21	0
2.1	Project Background	12-Feb-11	18-Feb-11	7	100%	5	7	0
2.2	Literature Review	18-Feb-11	24-Feb-11	7	100%	5	7	0
2.3	Methodology	24-Feb-11	2-Mar-11	7	100%	5	7	0
3	**Stage 3: Proposal Defence**	5-Mar-11	23-Mar-11	19	100%	13	19	0
3.1	Revised Extended Proposal	7-Mar-11	19-Mar-11	13	100%	10	13	0
3.2	Proposal Defence Preparation	20-Mar-11	22-Mar-11	3	100%	2	3	0
3.3	Proposal Defence Presentation	23-Mar-11	23-Mar-11	1	100%	1	1	0
4	**Stage 4: Interim Report**	24-Mar-11	22-Apr-11	30	100%	22	30	0
4.1	Post Proposal Defense	24-Mar-11	27-Mar-11	4	100%	2	4	0
4.2	Analysis	31-Mar-11	4-Apr-11	5	100%	3	5	0
4.3	Discussion/feedback from other lecturer	1-Apr-11	1-Apr-11	1	100%	1	1	0
4.4	Architecture Design & System Flow	4-Apr-11	15-Apr-11	12	100%	10	12	0
4.5	Prototype Design	15-Apr-11	22-Apr-11	8	100%	6	8	0

APPENDIX II

PROJECT GANTT CHART

No	Activities	Months									
		FYP I						FYP II			
		Jan	Feb	Mar	Apr	May – Aug		Sept	Oct	Nov	Dec
1	**Planning**										
	Choose topic										
	Preliminary research on topic										
	Specify scope										
	Feasibility analysis										
2	**Analysis**										
	Understanding concepts										
3	**Design**										
	Diagram Design										
	Flow Chart										
	Prototype Design										
4	**Development**										
	Data Extraction										
	Integration data										
	Functionalities										
5	**Testing**										
	Unit Testing										
	Integration Testing										
	Process Testing										
	Acceptance Testing										

APPENDIX III
TECHNOLOGY ACCEPTANCE MODEL SURVEY

USER ACCEPTANCE SURVEY OF REDUXGO APPLICATION

Welcome readers to the survey page of ReduxGO - a new application for car drivers!
Kindly go through the explanation below to get more understanding on how the application works. The survey will basically ask for your feedback on the usage of this new technology and identify the level of your intention to use it!

WHAT IS REDUXGO?

Before we go into detail about ReduxGO application, let we first understand the background of it.

The objective of ReduxGO application development is to create an application that will be used in avoiding driver to be trapped in Stop-and-Go scenario during highway congestion.ReduxGO also promote safety precaution on a proper and safe distance to be maintained by car driver.

Highway congestion can occur even there are no traffic lights, accidents, road works or police road blocks happen along the way. Have you ever wondered why congestion can still happen? ReduxGO was developed to counter the main reason why highway congestion occur, which is the occurrence of Stop-and-Go along the congestion. Stop-and-Go occurs when driver did not maintain a suitable distance between their car and preceding vehicle, thus require them to stop their car whenever the distance is not appropriate.

- STOP-AND-GO: Occur when driver will need to stop their car whenever they are struck in traffic congestion and somehow it will take time for them to move back their car from the speed 0.
- More info about Anatomy of Highway Traffic Congestion: http://www.youtube.com/watch?v=Suugn-p5C1M

HOW REDUXGO WORKS?

Image 1: ReduxGO detect car motion from the camera image. Based from the detection area, the application will recommend the appropriate speed for the driver in order to maintain safe distance between user's car and preceding vehicle.

Image 2: To accommodate the driver condition while driving, ReduxGO will provide a **Text-to-Speech** function that use verbal instruction on the recommended speed. The speed will be updated once in 5 seconds or if there any changes on the preceding vehicle's position.

HOW TO USE REDUXGO?

1. Activate ReduxGO application on your Android phone.
2. Insert the height of your car (one time only).
3. Put your phone at the front dock of your car, preferably in the middle part.
4. You can start driving, and wait for the recommended speed, whenever applicable (i.e if the distance between your car and preceding vehicle is quite close/inappropriate).
5. The recommended speed will be delivered to you through two (2) approach which are via screen display and verbal command from the phone.

Please rate the survey questions accordingly: Strongly Disagree (1) to Strongly Agree (5)

PERCEIVED USEFULNESS - A specific application system is perceived to be usefulness by prospective users if they believe the system will improve or facilitate their job performance within an organizational context.

PU1 I believe the use of ReduxGO application would help to maintain safe distance between my car and preceding vehicle.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

PU2 I believe the use of ReduxGO application during my driving would hep to increase the level of safety by controling my car speed.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

PU3 I believe the use of ReduxGO application would reduce the Stop-and-Go scenario during highway congestion.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

PU4 Overall, I found the ReduxGO application useful for my driving.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

PERCEIVED EASE OF USE - The system is perceived to be ease of use if the prospective user believes that the use of the system will be free of effort.

PEOU1 Learning to use ReduxGO application is easy for me.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

PEOU2 It is easy to use ReduxGO while I am driving.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

PEOU3 Overal, I believe ReduxGO application is easy to use.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

BEHAVIORAL INTENTION - To use (accept) the technology, and then generate the actual usage behavior.

BI1 I will use ReduxGO application on a regular basis in future.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

BI2 I will frequently use ReduxGO application in the future.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

BI3 I will strongly recommend others to use ReduxGO application.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

BI4 Overall, I intend to use ReduxGO application during my driving.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

ATTITUDE

ATT1 In my opinion, it is desirable to use ReduxGO application.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

ATT2 I think it is good for me to use ReduxGO application.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

ATT3 Overall, my attitude towards ReduxGO application is favorable.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Reset

REFERENCES

1. Yuki Su., Minoru F., Macoto K., Katsuya H., Akihiro N., Katsuhiro N., et al. (2008). "Traffic congestions without bottlenecks - experimental evidence for the physical mechanism of the formation of a jam", *New Journal of Physics.*
2. Adomavicius, G., & Tuzhilin, A., (2008). "Context-Aware Recommender Systems (CARS)" *Proceedings of the 2008 ACM conference on Recommender systems.*
3. Francesco R., (2010). "Mobile Recommender Systems" *2nd Workshop on Context-Aware Recommender Systems (CARS-2010).*
4. Stojanovic, D. (2009). *Context-Aware Mobile and Ubiquitous Computing for Enhanced Usability.* Hershey, USA: Information Science Reference.
5. Ministry of Transport, Malaysia (2010). *Media Statement By Yb. Dato' Sri Ong Tee Keat, Minister Of Transport, 5 Februari* 2010, Retrieved March 1, 2011, from http://www.mot.gov.my/index.php?option=com_content&task=view&id=680&Itemid=1
6. Traffic Flow Theory. In *Traffic Flow Theory: Theory and Concept* Retrieved February 20, 2011, from http://www.webs1.uidaho.edu/niatt_labmanual/Chapters/trafficflowtheory/theoryandconcepts/ShockWaves.htm
7. Lembaga Lebuhraya Malaysia (2011). *Traffic Updates*, Retrieved March 1, 2011, from http://www.llmtrafik.gov.my/traffic.aspx
8. Dewan Bandaraya Kuala Lumpur, Malaysia (2011). *Integrated Transport Information System*, Retrieved March 1, 2011, from http://www.itis.com.my
9. Bailey, L., (2004). *Smart cruise control eliminates traffic jams* Retrieved February 23,2011, from http://www.ur.umich.edu/0304/July19_04/19.html
10. Dey, A.K., & Abowd, G. D. (2000a). Towards a better understanding of context and context-awareness. *Workshop on the What, Who, Where, When and How of Context-Awareness, as part of the 2000 Conference on Human Factors in Computing Systems (CHI 2000)*, ACM Press, New York, (pp. 304 – 307).
11. Bajaj R., Ranaweera S. L., Dharma P. A. (2002) "GPS: Location Tracking Technology" *France Telecom R&D, San Francisco, CA.*

12. Chan K., Ordys A., Duran O. (2010) "A System to Measure Gap Distance between Two Vehicles Using License Plate Character Height", *ICCVG'10 Proceedings of the 2010 international conference on Computer vision and graphics: Part I.*
13. Android Market (2010), *Smart Measure*, Retrieved March 15, 2011, from https://market.android.com/details?id=kr.sira.measure
14. Android Market (2011), *Smart Distance*, Retrieved March 21, 2011, from https://market.android.com/details?id=kr.sira.distance
15. XDA-Developers (2010), VIO App uses your Phone's Camera for Motion Detection, Retrieved April 8, 2011, from http://www.xda-developers.com/windows-mobile/vio-app-uses-your-phones-camera-for-motion-detection/
16. Heidjen H. V. D., Kotsis G., Kronsteiner R., (2005) "Mobile Recommendation System for Decision Making 'On-the-Go'.", ICMB, pp.137-143, *International Conference on Mobile Business (ICMB'05).*
17. Ferscha L., Hechinger M., Mayrhofer R., Santos R. M., Franz M., Oberhauser R., (2004) "Digital Aura", *Advances in Pervasive Computing*, part of the Second International Conference on Pervasive Computing, Austrian Computer Society (OCG).
18. Donaldson R. W., (1967) "Communication via the auditory, tactile, and kinesthetic senses," *IEEE Transactions on Information Theory*, vol. 13, pp. 11-21.
19. Dennis A., Wixom B. H., Tegarden D. (2010) *System Analysis and Design with UML: An Object-Oriented Approach (3rd Edition)*, John Wiley & Sons (Asia) Pte Ltd, Asia, (pp. 14 - 15).
20. Google Code, *Google Project for Android*, Retrieved April 15, 2011, from http://code.google.com/android/
21. Webopedia Computer Dictionary, *Android SDK*, Retrieved November 25, 2011 from http://www.webopedia.com/TERM/A/Android_SDK.html
22. The Eclipse Foundation Open Source Community Website, *About The Eclipse Foundation*, Retrieved November 25, 2011 from http://www.eclipse.org/org/
23. Android Developer, *ADT Plugin for Eclipse*, Retrieved November 27, 2011 from http://developer.android.com/sdk/eclipse-adt.html

24. Cognitics – Resources for Cognitive Robotics, *Seeing with OpenCV: Introduction to OpenCV*, Retrieved November 27, 2011 from http://www.cognotics.com/opencv/servo_2007_series/part_1/index.html
25. Evans C., (2009), "Notes on the OpenSURF Library", *Msc Computer Science, University of Bristol.*
26. Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 13(3), 319–340.