



## General Sir John Kotelawala Defence University

Faculty of Computing  
Department of IT

Software Engineering Project undertaken in partial fulfillment of the requirements for the Bachelor of IT Degree program

Student Details		
Group Number	Student Number	Student Name
01	ICT/12/0121	WMIL Wasalage
Project Details		
Project Theme	Tablet PC based Motor Vehicle Claim Assessment Solution	
Start Date	22/12/2014	
Submission Date	28/12/2015	
Supervisor	Mr.Nandana Pathirage	

## **Abstract**

In motor vehicle claim settlement procedure, the preliminary assessment depends solely on the discretion of the insurance agent. He uses his experience and knowledge to assess the damage. Typically, customers are not satisfied with the assessment of the damage and there are no prescribed criteria or template on which the assessment is based. In most instances assessments are inadequate and customers are left with no choice but to bear the loss. In order to overcome of above mentioned problems this paper discuss a software solution to develop a smartphone based motor vehicle insurance claiming solution (SBMVCS) in order to enhance the information communication between the office and the field agents to minimize the loopholes in the current procedure. The solution consists of 2 main components as the online web application for the use of ground office staff and the smartphone application for the use of field agents to assist with the assessment process. The proposed software solution is stored in a web application server providing access to the system through a standard web browser. As for the mobile application, it access the system and the database via mobile networks through a web service. Overall system is divided in to three layers names as application layer, data layer and presentation layer. Thus the solution will allow the agents to provide real time updates and maintain the end to end connectivity with the company database.

Keywords— Smart Devices, insurance claiming, web services

## **Plagiarism Declaration**

I confirm that the enclosed written work, entitled "Tablet PC based Motor Vehicle Claim Assessment Solution" is entirely my ideas except where explicitly stated otherwise. I declare that wherever I used copying, paraphrasing, summarizations or other appropriate mechanism related to the use of another author's work, it has been properly acknowledged in accordance with normal scholarly conventions. I understand that wherever 6 or more consecutive words are extracted from a source they must be acknowledged and 10 consecutive words that are unacknowledged will be regarded as of plagiarism.

.....  
Signed: WMIL Wasalage  
ICT/12/121  
Dated: 2015-11-30

## **Acknowledgement**

I would like to express my deepest appreciation to all those who provided us the possibility to complete this project successfully.

I would like to express gratitude towards all manager, claim agents, operational staff, without their valuable input this would have been possible. I thank them heartily for their valuable time squandered.

In addition, a special thanks goes to former Head of Department Lt. Col SPP Pakshaweera and all the staff members of IT & Mathematics who provided their valuable advises during the project.

I would also like to thank our families and colleagues, who provided unwavering support through the entire project.

Finally and most importantly, a very big “thank you” is due to my project supervisor Mr.Nandana Pathirage for his continues support and inspiring guidance throughout the research project.

## Table of Contents

<b>Abstract.....</b>	2
<b>Plagiarism Declaration.....</b>	3
<b>Acknowledgement.....</b>	4
<b>Table of Figures.....</b>	10
<b>Table of Tables.....</b>	12
<b>Chapter 1 : Introduction.....</b>	12
1.1 Prolegomena.....	13
1.2 Objectives.....	13
1.3 Background and Motivation .....	14
1.4 Presenting Problem .....	14
1.5 Hypothesis.....	15
1.6 Software solution for Motor Vehicle Claim Settlement.....	15
1.7 Resource Requirement .....	16
1.8 Structure of Thesis .....	16
1.9 Summary .....	16
<b>Chapter 2 : Analysis of Existing Literature .....</b>	17
2.1 Introduction .....	17
2.2 Motor Vehicle Insurance Industry: An overview.....	17
2.3 Smartphones in the Age of Mobility and Smart Devices: An Overview .....	18
2.4 Mobile Network Penetration and Trends of Usage of Smartphones in SL ..	20
2.5 An Overview on Web Services .....	23
2.6 Examination of Past Research.....	24
2.7 Summary .....	26
<b>Chapter 3 : Technology Adoption .....</b>	27
3.1 Introduction .....	27
3.2 Software Development-Technology Considerations .....	27

3.3 Language and Tools Selection.....	27
3.4 Database Selection.....	29
3.5 Technical Requirements .....	29
3.6 Usability Requirements .....	29
3.7 Summary .....	30
<b>Chapter 4 : Approach to TBMVCS.....</b>	<b>31</b>
4.1 Introduction .....	31
4.2 Hypothesis .....	31
4.3 Research Methodology .....	31
4.4 Research Design .....	32
4.5 Users .....	33
4.6 Functional Requirements .....	33
4.7 Non-Functional and Performance Requirements .....	34
4.8 Input.....	35
4.9 Output.....	35
4.10 Process .....	35
4.11 Features .....	36
4.12 Summary.....	36
<b>Chapter 5 : Analysis &amp; Design.....</b>	<b>37</b>
5.1 Introduction .....	37
5.2 Data Gathering & Analysis.....	37
5.3 UML Designs for the New Solution .....	38
5.3.1 Use Cases for the New Solution .....	38
5.3.2 Class Diagram for the New Solution .....	42
5.3.3 Sequence Diagrams for New Solution .....	42
5.4 Overall System Architecture.....	45
5.4.1Application Layer.....	46
5.4.2 Data Layer.....	47
5.4.4 Presentation Layer .....	47
5.5 Software Architecture.....	47

---

5.6 Module Architecture.....	48
5.7 Database Design .....	52
5.7.1 EER Diagram .....	53
5.7.2 Database Relationship diagram.....	54
5.7.3 Data Type Design for the Database .....	55
5.8 Interface Design.....	58
5.8.1 Tab application .....	58
5.8.2 Smartphone Application .....	65
5.8.3 Web application.....	66
5.9 Summary .....	70
 <b>Chapter 6 : Development.....</b>	 71
6.1 Introduction .....	71
6.2 Software Development-Technology Considerations .....	71
6.2.1 Development Language Selection.....	71
6.2.2 Database Selection.....	72
6.2.3 Development Tool Selection.....	73
6.3 Development Approach.....	76
6. 4 Database Development.....	76
6. 5 Web Application Development Architecture .....	79
6.5.1 View Layer .....	80
6.5.2 Model Layer.....	82
6.5.3 Business Layer .....	83
6.5.4 Data Layer.....	84
6.5.5 WCF Web service .....	85
6.5.6 Communication between Layers .....	86
6.6 Tablet PC and Smartphone Application Development Architecture .....	89
6.7 Summary .....	89
 <b>Chapter 7 : Testing and Implementation.....</b>	 90
7.1 Introduction .....	90
7.2 System Testing Objectives.....	90

7.3 Testing Strategy.....	91
7.3.1 Unit testing .....	91
7.3.2 Integration testing.....	91
7.3.3 System Testing.....	91
7.3.4 Acceptance Testing.....	92
7.4 Test Plan.....	92
7.5 Test Results .....	95
7.6 Implementation Plan .....	119
7.7 System overview .....	119
7.8 Major Tasks on System Implementation .....	119
7.9 Installation.....	120
7.10 System Changeover .....	120
7.11 Data Conversion.....	120
7.12 User Training .....	121
7.13 Summary.....	121
 <b>Chapter 8 : Evaluation</b> .....	122
8.1 Introduction .....	122
8.2 Types of Evaluation.....	122
8.3 Evaluation of Project Outcome .....	122
8.4 Significance of the Solution .....	124
8.5 Evaluation of Project Practices.....	126
8.6 Summary .....	126
 <b>Chapter 9 : Conclusion</b> .....	127
9.1 Introduction .....	127
9.2 System Summary .....	127
9.3 Future Enhancement.....	128
9.4 Authors Conclusion .....	129
9.5 Summary .....	129

<b>References.....</b>	130
Appendix A .....	132
Appendix B .....	135
Appendix C.....	137
Appendix D .....	147

## Table of Figures

Figure 1: Loss Ratio - Non Life .....	18
Figure 2: Smartphones Becoming More Powerful.....	19
Figure 3: Increase of mobile subscription in Sri Lanka .....	21
Figure 4: How Internet Usually Accessed.....	21
Figure 5: Dialog GSM Coverage .....	22
Figure 6: Mobitel GSM Coverage .....	22
Figure 7: Web Services Architecture.....	23
Figure 8: Research Design.....	32
Figure 9 : Interaction of Users with Components .....	39
Figure 10: Use case Web Application .....	40
Figure 11: Use case Tab Application.....	41
Figure 12: Use Case for Mobile Application.....	41
Figure 13: Class Diagram for New Solution.....	42
Figure 14: SQ Diagram for Accessing a Module in Web Application.....	43
Figure 15: SQ Diagram for Accessing a Module in Mobile Application .....	44
Figure 16: SQ Diagram for Accessing a Module in Tab Application.....	45
Figure 17: Overall Solution Architecture.....	46
Figure 18: Login Module.....	49
Figure 19: Web Application Modules .....	50
Figure 20: Tab Application Modules.....	51
Figure 21: Mobile Application Module .....	52
Figure 22: EER Diagram.....	53
Figure 23: Database Relationship Diagram .....	54
Figure 24: Login Interface of Tab Application .....	58
Figure 25: Main Menu of Tab Application .....	59
Figure 26: Policy Information Interface .....	60
Figure 27: Claim Form.....	61
Figure 28: Claim Form.....	61
Figure 29: Claim Form.....	62
Figure 30: Claim Form .....	63
Figure 31: Garages Interface .....	64
Figure 32: Tow Truck Services Interface.....	65
Figure 33: Mobile Application Interface .....	66
Figure 34: Login Interface of Web Application .....	67
Figure 35: Home Page Interface .....	68
Figure 36: Spare parts Interface.....	69
Figure 37: Add Spare part Form.....	70
Figure 38: SQL Server Tables.....	77
Figure 39: SQL Server Stored Procedures.....	78
Figure 40: Sample Stored Procedure.....	79
Figure 41: Project Structure.....	80
Figure 42: CDN Library Links.....	80
Figure 43: Sample from Front End .....	81

Figure 44: Sample from Code behind.....	82
Figure 45: Sample Class from Model Layer .....	83
Figure 46: Sample Class from Business Layer .....	84
Figure 47: Connection Strings.....	84
Figure 48: Sample Code from Data Layer Class .....	85
Figure 49: Sample Code from Web Service.....	85

## **Table of Tables**

Table 1: Customer Table .....	55
Table 2: Claims Table.....	56
Table 3: Customer-Vehicle-Policy Table .....	57
Table 4: System Test Plan-Web Application.....	94
Table 5: System Test Plan-Tab Application .....	95
Table 6: WCF Service Test Plan .....	95
Table 7: System Test Plan-Mobile Application.....	95
Table 8: Evaluation of Tab Application .....	123
Table 9: Evaluation of Mobile Application .....	123
Table 10: User Interface Evaluation .....	124
Table 11: Solutions to Problems Summary.....	125

# Chapter 1

## Introduction

### 1.1 Prolegomena

In the modern world people have introduced software solutions as a major way of handling real world problems. Software solutions can be used to enhance the accuracy, efficiency, cost effectiveness, reusability and many more feature of a manual process. With the emergence concepts such as Internet of Things and advancements in networking people are connected with each other more than ever. Among other software technologies programming, databases, web and multimedia have been the most widely used for development of software applications in order to present solutions to various industries. Nowadays numerous software engineering technologies, tools, platforms are available for development of software applications for enterprises. This thesis presents a project to develop a software solution for the domain of motor vehicle insurance claiming procedure. In doing so, this chapter outlines background and motivation for the project, problem in brief, hypothesis, proposed solution, resource requirement and overview of the structure of the rest of the thesis.

### 1.2 Objectives

With the aim to develop a smartphone based software solution to streamline the extant motor vehicle claim assessment and settlement procedure in order to benefit both the insurer and the client followings are the key milestones which needed to be achieved successfully during the project.

- To identify the weaknesses in phases of existing motor vehicle claim settlement procedures.
- To investigate similar researches and projects steered to mitigate identified weaknesses with the aid of technological advancements.
- To develop the web application linked with a central database and host in a server such that employees can access the system through a standard web browser.
- To develop and host a web service along with web system in order to allow any platform smartphones to connect to the system.
- To develop smartphone application to access the main system as well as the database through the hosted web service.
- To provide means to make assessments of damages more effectively and efficiently through access to an online central database.
- To improve customer satisfaction by the reduction of misjudged claims.

## **1.3 Background and Motivation**

Over past years, the use of automobiles have become more common with the increasing complexity of human lives. In Sir Lanka for any vehicle, it is must to have a motor vehicle insurance before using the vehicle. The growth in the automobiles usage in the country has led to the establishment of several insurance companies and it has become a competitive market segment in the industry where there is constant cash flows. In any motor vehicle insurance company, one of the key process is the claim settlement of their customers in case of an accident. It can be seen that cost savings and efficient process improvements in claims management business processes have a great impact on the key performance indicators of insurers.

When a motor vehicle meets with an accident, in the past the traditional way was to wait till the police produce a report of the accident which may take usually more than 2 weeks. Therefore sometimes vehicle owners have to bear the cost of repair on his own before getting the insurance money. As a solution the 'on the spot' policy was introduced. In the on the spot claim settlement process, after an accident has happened a field agent from the insurance company has to go to the accident location and do the preliminary assessment depends solely on the discretion of the insurance agent to complete the claim settlement. Whether it is a on the spot or not an assessment should be done on the accident location. Agents should use his experience and knowledge to assess the damage. There are problems arisen with this scenario which will be discussed in the next section.

## **1.4 Presenting Problem**

Insurance companies has become a major revenue generator in the business sector over the past years. Their main business is handling life and general insurance policies and under the general insurance solutions insurers usually provide motor, fire, marine, personal, engineering, medical, title, and miscellaneous insurance solutions. The growth in the automobiles usage in the country has led to the establishment of several motor insurance departments in these companies and it has become a competitive market segment in the industry where there is constant cash flows.

Even though there is several companies competing in the industry, they are confronted with a gradual decrease of motor policy renewals which has caused a decline in revenue. This has caused due to the inherent loopholes in the vehicle claiming procedure. In the current procedure of claim settlement, on the scene of an accident, an insurance agent does a preliminary assessment of the damage to the vehicle. This preliminary assessment depends solely on the discretion of the insurance agent. He uses his experience and knowledge to assess the damage. Typically, customers are not satisfied with the assessment of the damage and there are no prescribed criteria or template on which the assessment is based. In most

instances assessments are inadequate and customers are left with no choice but to bear the loss.

Furthermore, the process of doing a preliminary assessment, verifying the documents and photographs in the scene of accident and the process of approving the claim and reprocessing the documents in the branch/head office to settle the claim causes hefty delays. These loopholes caused clients to leave the company effecting a decline of motor policy renewals.

To insurance companies, issues regarding the vehicle claim procedure is of utmost importance as motor policies are their top income earner and responsible for their considerable market share. As a solution for the delays 'On the spot' motor vehicle claiming procedure was introduced. In this methodology also several loopholes were seen similar to the earlier process due to the fact, the agent had limited access to the company database or any other resources. Hence, the need to re-engineer the existing motor vehicle claim management process using newer technologies is made apparent.

## **1.5 Hypothesis**

Technology is integrating with our lives more and more with the advancements in the technology. Today's world, even hand held mobile devices has dominant processing power cable of running applications smoothly. With these powerful hardware capabilities and software solutions combined using networks to convert traditional processes to automate processes. Therefore it is suggested that

**H<sub>1</sub>:** Web based solution with an integrated tab application and mobile application can address the issues arisen in the claim settlement of a motor vehicle accident

**H<sub>0</sub>:** Web based solution with an integrated tab application and mobile application will not address the issues arisen in the claim settlement of a motor vehicle accident

**H<sub>1</sub>:** By implementing the solution it will benefit both the customers and insurance company while improving customer satisfaction also.

**H<sub>0</sub>:** By implementing the solution it will not benefit both the customers and insurance company and will not improve customer satisfaction.

## **1.6 Software solution for Motor Vehicle Claim Settlement**

In view of above mentioned problems the proposed solution is to develop a Tablet PC based motor vehicle insurance claiming solution (TBMVCS) in order to enhance the information communication between the office and the field agents to minimize the loopholes in the

current procedure. The solution will consist of 3 main components as the online web application for the use of ground office staff, the tablet pc application for the use of field agents to assist with the assessment process and a mobile application for client to request claim. Thus the solution will allow the agents to provide real time updates and maintain the end to end connectivity with the company database.

## **1.7 Resource Requirement**

In order to execute the application in a successful manner the resources are required in technical and human perspectives. Therefore as for the technical perspectives it requires hardware and software in order to establish the application. It requires a tablet pc to run the application while central web host location to host the system and database. Apart from that for ground staff will need computers and an internet connection to connect to the system. In human perspective, it requires the people who are aware of working with the smart devices and the people who are having basic knowledge of working with a computer will be able to handle the application with the internet connection.

## **1.8 Structure of Thesis**

In context to the thesis this is an effort to represent the view of the entire research. In here these information related to the research is presented with separate chapters in order to get an easier understanding regarding the process. It includes the chapters as Introduction, Literature Review, Technology Adoption, Approach, Analysis & Design, Development, Testing & Implementation, Evaluation and Conclusion.

## **1.9 Summary**

As deliberated so far vehicle has become a necessity in our daily life making the vehicle important too in our live. Every vehicle need a motor vehicle insurance thus, insurance companies has become a major revenue generator in the business sector over the past years. In the traditional insurance claiming it was discovered that there were many loopholes. But with the help modern of powerful hardware and software, there is a need for a computerized claiming solution for motor vehicle insurance industry which will improve the efficiency, accuracy, cost effectiveness of the traditional procedure. In the next chapter it will be discussed about similar researches done on this subject.

# Chapter 2

## Analysis of Existing Literature

### 2.1 Introduction

This chapter is dedicated to scrutinizing literatures which investigate similar problem domains in the hope of identifying and analyzing basic concepts embedded within that body of research. Hence, first it is discussed about the motor vehicle insurance niche in the insurance field. It is followed by a discussion on smartphone technologies, usage of smartphones in Sri Lanka and an overall discussion about the emerging web technologies in the technical world.

In the next section, the feasibility of mobile technologies to the insurance industry is examined through studying past researches conducted on the similar subject by various researchers.

### 2.2 Motor Vehicle Insurance Industry: An overview

Over past years, the use of automobiles have become more common with the increasing complexity of human lives. In Sir Lanka for any vehicle, it is must to have a motor vehicle insurance before using the vehicle. The growth in the automobiles usage in the country has led to the establishment of insurance companies and it has become a competitive market segment in the industry. In the Fitch ratings it is stated that “Intense pricing competition in the motor segment is likely to hold the combined ratios in non-life above 100%”. Due to fall in imports with increased customs duty and unfavorable exchange rate movements, growth in the motor sector has dropped to 16.49 % in 2012 (2011: 29.56%). Yet, the influence of the motor segment in non-life remained high at 63.64% and will remain the major GWP contributor in non-life (Fitch ratings, 2014).

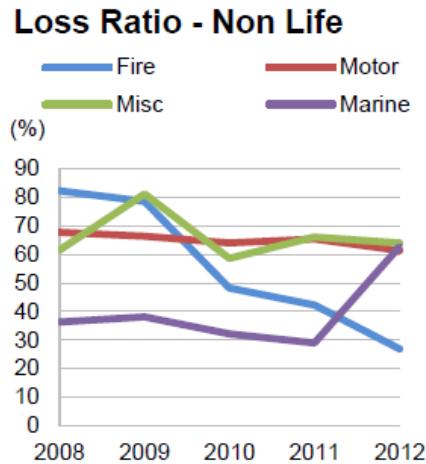


Figure 1: Loss Ratio - Non Life  
Source: FitchRatings (2014)

Motor vehicle industry can be considered as an industry where there is a constant financial flow in-and-out with respect to a company. It is argued that cost savings and efficient process improvements in claims management business processes have a great impact on the key performance indicators of insurers(Baecker and Bereuter, 2010). The main reasons why cost savings are hard to achieve include the late involvement of insurers after car accidents, the limited integration of business partners such as repair shops, and improper information about the case circumstances (Kaeslin and El Hage 2008). As indicated by Capgemini 2013, Mobility has been acknowledged as a growing phenomenon in the insurance industry.

Since motor vehicle insurances is a fast growing area, it can be seen that with the help of latest technologies average claim life-cycle from set-up to close can be shortened as well as decrease the average costs per case since in contrast current motor vehicle insurance claiming procedure drags on for weeks to complete.

### **2.3 Smartphones in the Age of Mobility and Smart Devices: An Overview**

A smartphone can be simply considered as a mobile phone with highly advanced features. Beal (n.d.) has described smartphones as a handheld device that integrates mobile phone capabilities with the more common features of a handheld computer or PDA. A smartphone is expected to have a more powerful CPU, more storage space, more RAM, greater connectivity options and larger screen. Smartphones are more powerful than regular cellular phones and PDAs in almost every aspects such as efficiency, reliability, user friendly and etc. A typical smartphone now a days generally has a high-resolution touch screen display, Wi-Fi connectivity, Web browsing capabilities, and the ability to accept sophisticated applications. The most popular mobile operating systems at the time being are Android, iOS and Windows Mobile.

The fine integration of these powerful hardware and software has paved a way to introduce sophisticated mobile devices with number of different services all in just one device. Those provided features along with the rapid developments in the internet technologies have amplified the usage of smartphone in the society. According to Gartner, a leading information technology research and advisory company, smartphones accounted for 53.6 percent of the overall mobile phone sales in 2013, which exceeded the annual sales of feature phones. Gartner revealed the global sales of smartphones to end users reached 968 million units during 2013, which is an increase of 42.3 percentage points from 2012's figure of 680 million.

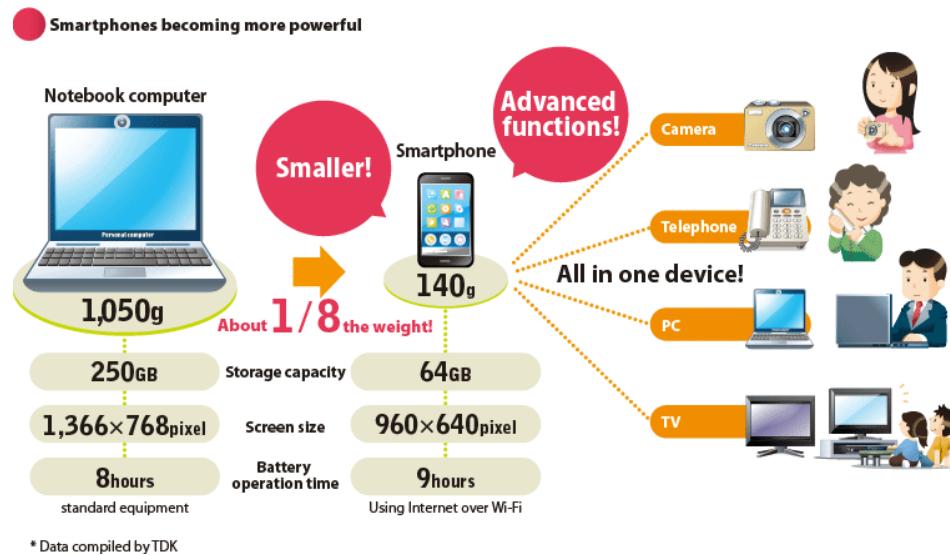


Figure 2: Smartphones Becoming More Powerful

Source: [www.global.tdk.com](http://www.global.tdk.com) (2012)

The above figure illustrate how smartphones are becoming more powerful day by day with the improvements in the technology field. Some of the common uses of smartphones are GPS to find your way, access your business data at any time, you will be able to conduct last minute researches, schedule keeping, managing the social collaborations and maintaining your schedules.

These trends make clear that these handheld devices and their technologies have become common currency in popular usage. Furthermore, they have penetrated every aspect of human existence. The widespread use of handheld devices in almost all major industries and sectors including financial services is a testimony for the ubiquitous applicability and popularity of mobile technologies.

Susan, a VP in Business Development at Scisbo Business Solutions states some applications of smartphone technologies which can be seen in the society. She mentions that medical doctors are able to monitor patients' health with heartbeat monitors, sleep monitors, and pedometers that connect to the cloud via a smartphone and send an alert when there's a

problem. People with chronic conditions such as diabetes can use a smartphone to monitor their blood sugar, making fewer trips to the clinic.

Furthermore she describes that smartphone docks in cars are valuable for more than just phone calls and entertainment. Now, they can be used for monitoring and diagnostics for the car itself or to track driving patterns for pay-as-you-drive insurance programs. Apart from that she also states that small size business reduce the energy use in commercial buildings by installing intelligent thermostats and heating controllers that can be programmed and monitored with apps and also smartphones can control door locks remotely or even control and monitor a whole security camera network.

It is the latest trend to develop smartphone applications which will communicate with your main server with the aid of a web service. Smartphone apps have been created to solve a problem, increase productivity, offer an intrinsic benefit, or for pure entertainment. Apple's iPhone store alone has more than 150,000 apps available for download and users have downloaded more than 3 billion apps. A custom application for your business can be developed to suit your need with minimum cost.

These results further strengthen the notion that use of handheld devices can greatly enhance the productivity of the business process and the productivity of employees. With the real time access to crucial business statistics, employees are given the opportunity to make informed decisions that greatly contribute to the overall productivity of business operations.

Considering breath of its features, it is clear to the researcher that smartphones have the capacity to be utilized in many areas of work as it is frequently seen that some government institutions and business organizations extensively use web services bases smartphone applications to enhance their customer interaction, productivity of the business and provide a strategic value to their businesses.

## **2.4 Mobile Network Penetration and Trends of Usage of Smartphones in Sri Lanka**

Accidents can happen anywhere inside this Small Island, it can be a remote location or an urban location. Because of that it is very important study the mobile network penetration inside Sri Lanka. Not only that it is of pivotal importance to study the level of mobile literacy of Sri Lankans to determine whether the insurance filed agents will be comfortable in using such smartphone based solution.

There has being a significant increase in the mobile subscriptions in Sri Lanka over the last decade. It is stated in tradingeconomics (2014) that in the year 2008 up to 95% of population is covered by mobile cellular network in Sri Lanka. The following chart extracted from

Trading Economics website illustrate the increase of mobile subscription in Sri Lanka from 2001 to 2013.



Figure 3: Increase of mobile subscription in Sri Lanka  
Source: Trading Economics (2014)

It is clear that Sri Lanka enjoys a high mobile literacy and considering the statistics of last ten years it is evident that emerging market trends in the mobile usage are seen in Sri Lanka. Apart from that, in Sri Lanka there are approximately over 4 million internet users. According to Nielsen's "2013 Sri Lanka review" report further emphasizes this trend which highlights 57% of the internet users in Sri Lanka do access internet through a mobile device.



Figure 4: How Internet Usually Accessed  
Source: Nielsen (2014)

In the present scenario there are five mobile network operators in Sri Lanka namely Dialog, Mobitel, Etisalat, Hutch and Airtel which provides mobile connections to consumers. Dialog offers a wide coverage and undisturbed service whereas Mobitel also have good coverage and specially a good 3G coverage. However if you are not traveling frequently Etisalat or Airtel would be good options. They offer lower call rates. Some youths opt to Hutch as they can easily obtain a connection and easily switch for a new one.

When considering the mobile network coverage available in Sri Lanka, it was obvious to researcher that Sri Lanka enjoys an Island wide mobile coverage encompassing almost 4/5th of the Island. This coverage is facilitated through different network types such as, CDMA, WCDMA, HSDPA, 3G, GSM, and GPRS mostly. Along with that 4G networks are also starting to emerge near major cities. Hence, this tendency to move ahead with latest next generation networks and the breadth of coverage testifies to the existence of positive trends in terms of mobile technological usage. The following figures shows the network coverage of two major service providers.

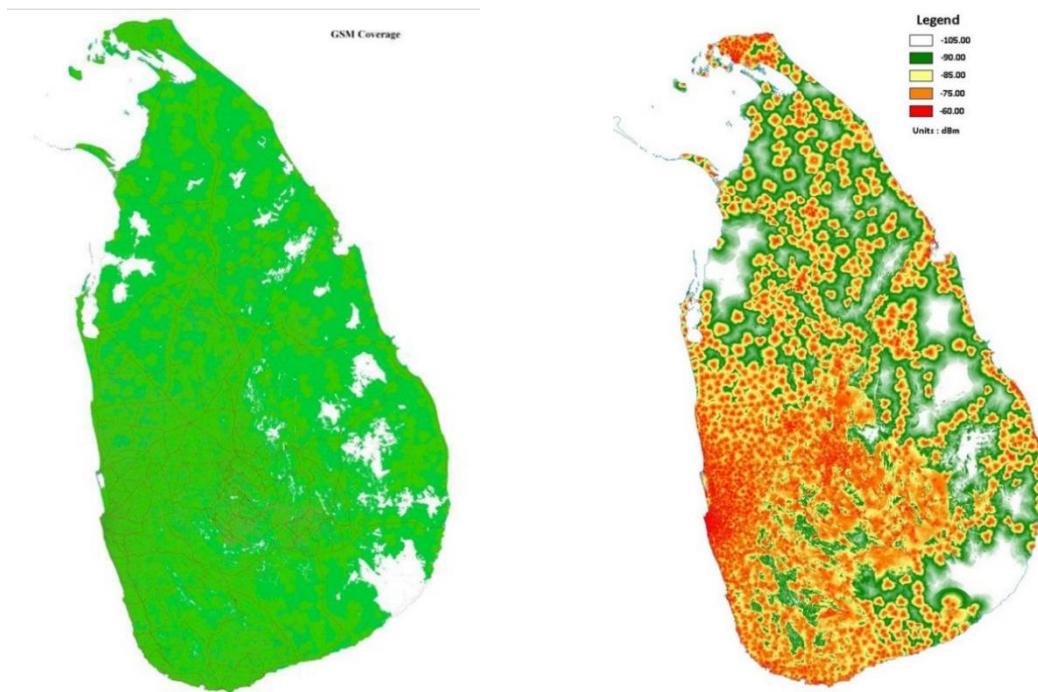


Figure 6: Mobitel GSM Coverage

Source: Mobitel (2014)

Figure 5: Dialog GSM Coverage

Source: Dialog (2014)

Although mobile and internet usage is widespread in the country it is also important to take a look at the usage of smartphones also. Couple of years back smartphones were not much popular on the market since they were perceived as high end devices which are considered to be expensive also. But now it's not the case it has become widely affordable and available across the island, you can buy a decent smart phone for a very reasonable price. This results in more people buying smartphones and more people using internet through their device.

By taking into consideration the above facts it is evident that mobile literacy in Sri Lankans is in a healthy condition and all the indications are towards positive growth. Moreover, expanded and expanding nature of mobile network coverage in Sri Lanka shows the emerging trends in mobile technologies and services. Hence, it can be said that Sri Lanka is in the threshold of an adorning mobile era which would facilitates life styles and cooperate

habits shaped by mobile technologies which indicates for the researcher that use of mobile technologies in the solution is a timely approach.

## 2.5 An Overview on Web Services

In the development of the proposed solution web services technologies will be used to connect the main system with the smartphone application. Web services can be identified as application components which communicate using open protocols (XML, SOAP, WSDL and UDDI) over an Internet protocol backbone. Web services are self-contained and self-describing. They can be discovered using UDDI and can be used by other applications (w3schools, 2014). It runs totally without any intelligence or input by a human being. One of the main ideas in developing such systems is that the human involvement is to a minimum and the human error is dealt to a nullity.

When the claiming agent conduct the assessment, the data has to be retrieved and also synchronized with the main database. This happens through the web server. Extensible Markup Language (XML) which is derived from Standard Generalized Markup Language (SGML) is a simple, flexible text format which is used for the development of the web server. There are no pre-defined tags in XML, user has to design their own tags. Document type definitions or XML Schema files are used by XML in order to describe data. XML works as a message carrier in web services.

Following figure demonstrate the overall architecture of a web service.

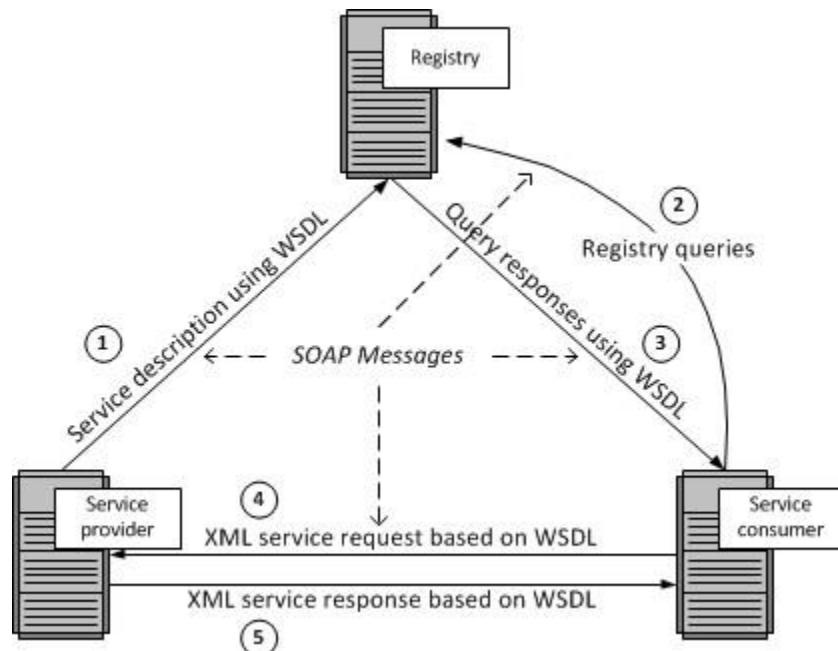


Figure 7: Web Services Architecture

Author: Barry DK (n.d)

Service provider host a web service and upload the service description using Web Service Description Language (WSDL) to a registry containing a list of WDL files. WSDL is an XML based language used to describe capabilities and locations of services or in simply how to work with the web service. WSDL uses Simple Object Access Protocol (SOAP) as its communication protocol. SOAP is a framework which support exchange XML based information in a network environment. SOAP is a wrapper class which encapsulates XML based information within HTTP packets in order to send through internet.

Service consumers will search the registry for available web services. Registry is a list of web services from different service providers. It uses the UDDI technology for maintaining the registry. UDDI stands for Universal Description, Discovery and Integration. This provides a platform independent, XML- based registry mechanism between the server and client in order to situate each other. SOAP is used as the communication protocol used by the UDDI.

AS the diagram indicates the once the relevant web service is found, registry will respond with the necessary WSDL files. Using the information in those files service consumer will be able to use the services provided by the provider.

The main advantage of using web services is it allow different applications from different sources to communicate with each other without time-consuming custom coding. Web services do not depend on any one operating system or programming language. They are platform independent because all communication is done using XML. Apart from that using web services help to tighten the security of your data and system since when accessing through a web service, external parties do not have the direct access to companies system or database.

By using the web services in the proposed solution it will enhance the security on cooperate data while providing real time access for the field agents. Apart from that, smartphone solution can be developed on any platform such as Android, Windows or Apple IOS because the smartphone application connect through the web service.

## **2.6 Examination of Past Research**

This section presents past research conducted by academia and industry with regard to use of mobile technologies in claim settlement procedure of motor insurance industry.

In a research conducted by Baecker and Bereuter (2010), they have presented an analysis of claims management processes in motor insurance, and identified several areas for technology-based process improvements. They have categorized the areas such as Mobile loss report, Partner management, Status notification and Customer feedback. In order to establish the technical foundation for identifies areas, they have designed and implemented a service-oriented architecture and has used it to develop a demonstrator, which illustrates

the identified process improvements. The demonstrator they have developed incorporates three major components namely Mobile phone application, Service-oriented integration architecture and Claims management enterprise system.

Furthermore they have discussed in the paper about the benefits of a mobile claims assistance application for customers involved in a car accident as well as for insurance companies. In addition, with the aid of demonstrator they have demonstrated that mobile phones are technically capable of reporting claims into commercial claims management enterprise systems.

In 2008, Kaeslin and El Hage conducted a research with the aim of identifying the possible cost reductions for claims management in motor insurance sector. As a result it was observed that the delays in claim settlement can largely be averted by automating the process. In addition to that it states that, it is vital to have proactive contacts between all the relevant parties throughout the claim settlement process. Kaeslin and El Hage also argues that one of the major challenges that insurance companies have to address in order to improve control over the claims management processes is the lack of adequate IT solutions in the process.

In another research conducted by Lee and colleagues (2007), they have studied about possible strategies and applications for M-commerce in the insurance industry. After conducting a survey among insurance agents in order to explore mobile solutions which are applicable in the business process and to identify the possible benefits for the agents' tasks, researchers concluded that personal digital assistant (PDA) technology is appropriate for the insurance industry and highlighted the potential for improvements of customer care and claims management.

Although in that era PDA was the popular mobile device which could fit into business solutions, in today's world smartphone are much more powerful than PDAs and can be much more helpful than PDAs. Since, the entire business world moving towards a virtual market place, the integration of M-strategies with insurance tasks will definitely hold the key for insurance industry in future.

Moreover, IBM (2009) conducted a similar study on process automation in claiming sector. The research investigated the potential effects of automating some steps in a claiming procedure on the quality of customer service. The research concluded that 60% of claiming time can be reduced by the automation which eliminated "low- to no-touch management" in several phases.

It is stated by Want (2009) that while the applications such as eCall are currently deployed in motor vehicles, similar applications will become available for mobile phones as their computational power increases, additional sensor technology becomes available, and the necessity to integrate mobile phones with enterprise wide systems will rise.

As the above researches indicates the interest of academics of the possibilities of integrating smartphone technologies to enhance claim management processes and shows the necessity for further research in this area considering the competitive nature of the motor insurance industry which demands the integration of such mobile technologies.

## **2.7 Summary**

This chapter was dedicated to conduct a survey regarding to problem domain, technologies available for the solution and also to examine past research conducted on the similar area of research. As the first step it is discussed about the motor vehicle insurance industry niche of insurance sector. It was pointed out that better coordination and management of claiming process will lead to gaining more profits.

Next it is reflected an overview about the smartphones in this era of mobility and smart devices. It is briefly explained about what is a smartphone device and about the powerful hardware and software combination that smartphones are made up of. Along with that as the next step mobile network penetration and smartphone usage in Sri Lanka is examined. Since the smartphone prices are gradually decreasing but with day by day advancements in the devices, people have showed positive responses to the use of smartphones. Mobile networks in Sri Lanka also covers large areas around all most all the major cities and day by day it is growing with new technologies such as 4G. By considering networks and smartphone technologies it can be deduced that in the present smartphone based approach for the solution is the most suitable option.

As the proposed solution includes web technologies, in the chapter it is also discussed about web services and other related components such as WSDL, UDDI and SOAP. By using the web services in the proposed solution it will enhance the security on cooperate data while providing real time access for the field agents. Apart from that, smartphone solution can be developed on any platform such as Android, Windows or Apple IOS because the smartphone application connect through the web service.

Finally this chapter presents an evaluation on past researches conducted by professional on the subject of use of mobile technologies in the motor vehicle insurance sector. It shows that there is a great potential towards research for mobile integrated solution for insurance companies as well as for the clients also.

# Chapter 3

## Technology Adoption

### 3.1 Introduction

This chapter is dedicated to discuss the software development of tablet pc based motor vehicle claiming solution from the technological perspective. It is discussed about the technologies considered when developing the solution and the factors considered when choosing the technologies. Furthermore technical and usability requirements are also mentioned in this chapter.

### 3.2 Software Development-Technology Considerations

In order to develop a successful system which meets the end user expectations, it is very important to use appropriate tools in the development process. Use of any inappropriate tools will only leads to develop a system with unnecessary errors and faults and it can affect the user expectations in efficiency and reliability of the solution. It is very vital to use appropriate computer language and any other necessary tools in order to develop a successful system. So these technologies and tools will help to develop the system within a minimum development time. The main objective of TBMVCS is to do a fast, easy, accurate claiming process via tablet pc. And this is done by connecting to the company database via web service with the use of mobile networks. So it is very important to consider some factors such as platform supportiveness and the efficiency of the system. And in order to meet above mention factors we must use the most appropriate tools available in the market to develop the system.

Technological considerations - followed during the development of the system

- Efficiency and Performance
- Re-usability and flexibility
- Object oriented development support

### 3.3 Language and Tools Selection

In modern programming field there are many programming languages and many more are introducing day by day. Core function of the system is vehicle claiming through the tablet pc, researchers have been done on many software development languages in order to recognize

the appropriate language that helps efficient data retrieval and updating from the company database via web service. Since the entire solution has 3 main components namely the ground office system, web service and tablet application it was vital to select a language that will support all these three features so that it will reduce the learning time of different languages. Several aspects such as build powerful web based applications, powerful, flexible, Simplified Data Access, platform dependency, build fast mobile applications were considered.

For the development of ground office system main technology considerations were ASP.net platform, java platform and PHP platform. Java and asp.net provide greater object oriented programming feasibility while PHP work as a fast backend server scripting language. Visual Studio IDE provide ASP.NET development environment with rich inbuilt functionalities for faster programming. As for the web service it was considered about coding a Visual studio web service, PHP web service and also WCF web service. WCF was considered to be a far better option than other web service types Microsoft has provided.

Another key part of the solution was the development of tablet applications. Main platform considerations were android, windows and IOS. In order to develop an IOS app it requires to learn Objective C language and it also need a Mac computer which are far more expensive than the others. Android is an open source platform with free available tools. There are lot of resources available for android development as well. Windows tablet pc was feasible to build on the same environment which was used to build the ground office system and the time taken for the learning curve was fairly less than android. Visual studio provides an in built emulator for testing purposes as well.

After considering several options ASP.Net was selected as the main programming platform to develop the ground office system since ASP.Net provides an easier and stable environment to create great web applications and it supports object oriented development greatly. Although it has to be purchased for commercial use, it can be purchased for free for university students from Dream spark. Bootstrap, a free CSS library was used to design the interfaces. C# programming language was used to implement the business logics in the backend of the system.

As for the initial step tablet application was developed for windows surface tab. It is using XAML for interface development and C# for backend coding. To connect the Tab application and the web system, WCF web service technology was selected since it can be developed using the same tools which are used to develop the system. Visual Studio 2013 was selected as the main tool for developing the solution while using open source text editors such as brackets were used wherever necessary.

### **3.4 Database Selection**

Database which is most important aspect of the overall system which handles all the relevant details related to insurance as well as claims. In order to fulfill the need of database operation this system has been used SQL Server Management Studio which enables to access and manage the database engine. Another reason for using this was it was free to purchase using Dreamspark account. Management Studio brings graphical tools for database management together with a rich development environment. Database is maintained at a central cloud server such that it can be accessed from wherever necessary.

### **3.5 Technical Requirements**

Technical requirements for the development of system are as follows.

- The ground office system was developed using Visual Studio 2012 using ASP.net platform. Bootstrap, an open source CSS library was used for interface designing while backend coding was done using c# language.
- Web service was implemented as a WCF web service which is available in C# language platform. JSON parser has being used for transferring data.
- Tab application was developed for Windows platform using Visual Studio 2013 Express version. It uses XAML for interfaces and C# as backend coding.
- The application package has been developed as it is compatible for any computer which runs on windows platform such as Windows Vista, Windows 7, Windows 8 and Windows 8.1.
- Tab application is available for devices which runs Windows 8 and windows 8.1.

### **3.6 Usability Requirements**

As the usability requirements it was expected the following standards from the developed system. Effectiveness, Efficiency, Safety Utility, Learnability and Memorability are factors which will be taken in discussion. Consistency and Standard Visibility of system, status Flexibility and efficiency of use, User control and freedom Match between system and the real world will be taken into consideration too.

Requirements that ensure that there is a good match between the system and its users. In most cases usability is expressed in terms of measurable objectives. The usability of the TBMVCS is to be considered with the education of the claiming staff members, how well they are conversant with technology and a tablet pc and such conditions. If the users are not up to the standards to use the system in an effective way, system developers should conduct training sessions to develop knowledge to use the system without having any trouble.

The tab application is going to be on a small screen compared to a desktop. Also since the claiming agent would be using it under direct sunlight the visibility of the screen would be an important aspect that needs to look at. The wisest option is to have the interface with dark black letters for more clarity. The use of colors might lessen the visibility of the claiming staff member to input data on to the device. Anyhow windows tabs come with option to choose dark and light themes which user can easily swap between them according to the situation.

Interfaces of the system components shall be designed with appropriate fonts, font sizes, colors and menus in way such that users are more comfortable to work with. Solutions software components shall be designed with simplicity but covering all necessary aspects ideally with the principle “Recognition rather than recall in mind.”

### **3.7 Summary**

Technology is one of the essential criteria when developing a software project. Through this chapter it has been described about the various technology options for the proposed software solution. Furthermore this chapter discussed the technical aspects that need to be considered for the developed solution as well as the usability requirements that need to be considered. Next chapter is dedicated for describing the approach which was used to develop the tablet pc based motor vehicle claiming solution.

# Chapter 4

## Approach to TBMVCS

### 4.1 Introduction

This chapter is dedicated for describing the approach for the development of TBMVCS. With the hypothesis is stated this chapter will give a brief idea about the users of the system. And most importantly this chapter will describe the functional and non-functional requirements for the solution. It is vital to identify the input, process and the output of any solution. It is described in this section and at the end some of the key features are stated in this chapter.

### 4.2 Hypothesis

Technology is integrating with our lives more and more with the advancements in the technology. Today's world, even hand held mobile devices has dominant processing power cable of running applications smoothly. With these powerful hardware capabilities and software solutions combined using networks to convert traditional processes to automate processes. Therefore it is suggested that

**H<sub>1</sub>:** Web based solution with an integrated tab application and mobile application can address the issues arisen in the claim settlement of a motor vehicle accident

**H<sub>0</sub>:** Web based solution with an integrated tab application and mobile application will not address the issues arisen in the claim settlement of a motor vehicle accident

**H<sub>1</sub>:** By implementing the solution it will benefit both the customers and insurance company while improving customer satisfaction also.

**H<sub>0</sub>:** By implementing the solution it will not benefit both the customers and insurance company and will not improve customer satisfaction.

### 4.3 Research Methodology

As inductive approach was used as the main approach according to which the research is conducted, research techniques such as, questionnaires, interviews, case studies, observations and document reviews are appropriately collaborated. To make the research more in depth and productive a combination of quantitative and qualitative methodologies were employed as the research style. Moreover the emphasis of quantitative research is on

collecting and analyzing numerical data. But with the difficulties arisen in the domain gathering of quantitative data was not successful.

On the other hand qualitative research is more subjective in nature than quantitative research and involves examining and reflecting on the less tangible aspects of a research subject. Interviews observation of workflow in sites is examples for qualitative methods which will be used in the project. Such a combination would allow the researcher to discern the techno-economic implications of a tablet pc based solution for motor vehicle claim settlement procedure.

#### 4.4 Research Design

The above mention research approach will be executed according to the dynamic system development life cycle on agile development methods as depicted in figure 11.

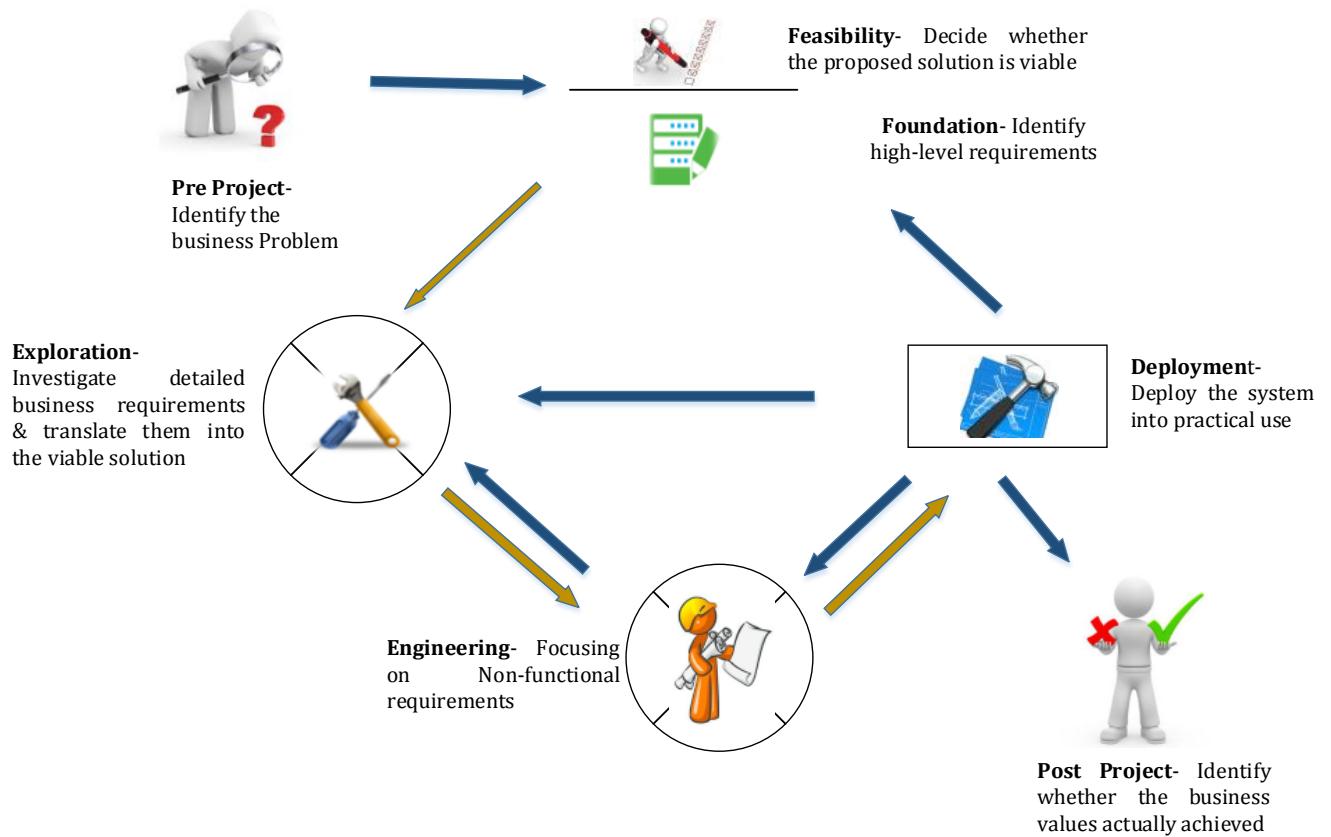


Figure 8: Research Design

Source: Author

## **4.5 Users**

The main users of the web system will be the ground office staff. They are able to add new records, edit records, and obtain reports through the web interface. Adding new records will include records such as new policies, garage details, tow truck services and vehicle spare part details as well. All details are kept updated in the database by the ground office users. Other users' type is the claiming field agent who will access the customer details through the tab application. They can conduct the whole claiming procedure on the spot of the accident and submit it to company database. They will be able to access all the updated, necessary customer, vehicle and spare part data through the tab application easily and in a fast manner.

## **4.6 Functional Requirements**

This section contains the functional requirements required for the tablet pc based solution for vehicle insurance claim settlement procedure. In the solution there are 3 main components namely the online web application, tablet pc application and mobile phone application. The requirements in this section specify the functions that each component must be capable of performing.

Online web application component is designed for the use of operating staff at the ground office. When a client need to buy an insurance, operating agents needs to register new customer and create account acquiring the relevant information. It will involve the following key functions.

- The users shall be able to create, view and update customer profiles by filling customer details such as first name, last name, date of birth, phone number, company details etc.
- The users shall be able to add, view and update insurance vehicle details such as model, manufacturer, engine number etc. as per the customers' vehicle.
- The users shall be able to add, edit and view insurance policies in to the system database.
- The users shall be able to add, view and update spare parts details relative to various manufactures and vehicle models.
- The system shall track client claim requests and display the locations in a map.
- The users shall be able to assess the submitted claims and approve them.
- The users shall be able to add details such as manufacturers, vehicle types, employees, and spare part categories.

The other component is the tablet pc application which will aid the field agent with the assessment process. When an accident happens, once the agents goes to the location, the person can access the following key functions using the tablet pc application.

- The user shall be accessed to the insurance policy details of the customer from the database.

- The user shall be able to access the insurance vehicle details and the details of the clients on the system database.
- The user shall be able to access and assess the client's insurance history.
- The user shall be able to complete the assessment of the accident by entering the details such as drivers' details of the caused accident, accident location details, cause of accident and damages to the vehicle.
- The users shall be able to attach photos and videos to the record using tablet pc camera.
- The system shall be able to complete the assessment by calculating the damage by involving the database and the web server.

From the mobile phone applications perspective

- User shall be able to report an accident for claim along with their GPS location.

## **4.7 Non-Functional and Performance Requirements**

In the below discussed are the non-functional and performance requirements to be considered when implementing the motor vehicle claim settlement solution.

When considering the performance, this would depend on factors such as the network connection involved. As the field agents should move to the accident location, the network data transfer speeds will depend on the signal strength received to the particular location with the network provider.

In today's smart devices market, there are several major OS platform which the devices work on. Since initial mobile application is developed for the Windows platform, Mobile application will require a smart device running on Windows OS platform. In considering the smart devices, another important factor which should be considered is the hardiness of the device. It is important to remember that this will be used outdoors and taken in the Sri Lankan tropical climate necessary precaution should be taken to protect the device.

System will be tested for identifying and reducing bugs to the minimum. If any bug may appear they will be debugged using demo test. Finally reliable software system will be introduced to provide real time information. The hours need for the maintenance of the system will be kept at a minimum by conducting testing thoroughly. The company database should also be updated with the current vehicle component values in the market and there should be constant updating done on the part of the head office.

The system will have its own security to prevent unauthorized write/ read/delete access along with its authentication module. Authentication module will include a module for encryption and decryption of authorization data. Apart from that necessary steps would be taken to back up the database periodically.

## **4.8 Input**

Ground office staff employees who will interact will be added to the system with valid authentication details. Then they will be able to log into the system with their credentials. Then they will be able to create new insurance records and enter customer details, vehicle details, policy information, vehicle spare part information updates and so on. These data input will be very vital for the assessment of the accidents when conducting a claiming procedure.

## **4.9 Output**

Accidents can happen anywhere in the country. Once an accident happened claim agent have to go to particular location to do the assessment. Agent will be equipped with a tablet pc installed with claim assistant application. He will be able to get all the necessary details such as customer information, vehicle information, policy information, spare part prices and so on. He will be able to conduct the assessment at that moment itself and attach photos to the claim record whenever necessary and finally submit them to insurance database. Apart from that he will be able to access the nearest company approved garages and tow truck service within a matter of minutes.

## **4.10 Process**

When a customer needs to insure his vehicle, ground office staff can register the user on the system adding necessary details such as customer profile, vehicle information which is going to be insured and the policy information such as amount of the insurance, what covers are needed by the customers and so on. Once the registration is complete company can issue a document with the insurance record id which is auto generated from the system. Once they are in the system claim assessment can be done via the tab application.

When an accident happens client can inform it via mobile app. Ground staff can respond to it and send an agent. Claim agent will go to the location and do the assessment via tab application. First agent needs to log into the tab application using valid credentials. Once he is logged on to the application, he can request the policy id from the customer and enter it in the tab application. Application will retrieve all the necessary data from the company database through the web system via a web system. If all the details are in order agent can proceed with the claim assessment by filling the necessary data forms, attaching photos if necessary, add the necessary costs and get the total cost and submit the data into the company data base through the web system via web service. It can be seen that the manual process is minimized to a process with only few click using this approach.

## **4.11 Features**

Features of the complete solution include maintaining all the insurance records in a central cloud database and access them from a standard browser from any location in the country. The main feature is to conduct the claim assessment through a tab application minimizing the loopholes present in the traditional claiming process. Claiming agent have the accesses to all the necessary data in his tablet pc which will help him to do the assessment in a speedy but accurate manner. Since the data is passed as JSON string format even with a slow network connection tab application will be able to retrieve data through the web service. As an added feature security will be ensured using https when hosting and enabling encryption and decryption when comes to authorization aspects.

## **4.12 Summary**

In this chapter, it is described about the one of the key area which is functional and non-functional requirements for the solution. Apart from that there is a brief description about the users of the solution and a description of the system from an input, process output perspective. It also includes a brief description of features included in the system such as central database security wise and also access to the system. In the next chapter it will be discussed about the design of the system in depth.

# Chapter 5

## Analysis & Design

### 5.1 Introduction

Since the approach to the development project is discussed this chapter focuses on explain the data gathering and analysis of that data to produce a set of user stories. User stories were converted into the functional requirements and UML techniques have been used to represent the system in different technical and non-technical perspectives. At the end of this chapter some of the important user interfaces are also shown.

### 5.2 Data Gathering & Analysis

Data gathering and analyzing the data in order to identify the exact requirement is a key factor in making a project success. Limited number of data collection methodologies were used to keep it simple. Main data source was from interviews and document reviews and field visits also help to gain rich knowledge in the domain.

The easiest method to get close to the staff and their information was the interview method. The staff was more conversant in the local language of Sinhala. They were much more comfortable with the interviews. The interviews were conducted amongst some senior managers as well. They came up with good suggestions.

Observation is another method which was used to gather data in the initial stages. This is a method which performs an informal assessment technique which involves specifying, counting, and recording MVCS procedure. The observation is an interesting part of the research as to the practical experience. The agonizing part was to wait until an accident or breakdown occurred. The biggest advantage of an observation tour is that you get a picture of the whole spectrum of the system. We see for ourselves that the claiming staff members are actually quite competent and pretty quit in their work. It just seems so that they want to finish off the evaluation as soon as possible. The important factor to consider here is that it shows us on what other areas do they might be vulnerable to make an error.

Documents reviews were very much helpful in designing the database for the new solution. There were several forms to be studies and finally design a compatible database design to hold all the necessary data. There were issues with some of the form kind since they were not able to provide it for studying purposes since they are confidential.

Since the agile methodology was used there was no pre planning like in traditional software development cycle. After few initial meetings few user stories were created using the knowledge and confirmed them with the users. After that the development was started and as they were completed feedback was taken and new requirements and improvements were identified. Similar cycle was maintained until the end of the project to keep the users also updated.

More details on analysis are attached in **Appendix A**.

## **5.3 UML Designs for the New Solution**

After the initial data gathering and analysis user stories are created and according to them system is designed according to various technical perspectives. The MVCS systems design is presented gradually by stating the overall system architecture. Then the rest of the architectural designs would be described independently. The software design, the module architecture, the database, web application, tab application and the interface design would all be dealt with in detail. This chapter would be the place where all the analysis and research comes in to usage. The theory will be tested in to practice and the limitations and ideas would be spelt out at each level of the design

### **5.3.1 Use Cases for the New Solution**

This diagram show how the users interact with the solution in respect the components of the solution which are web application, tab application and mobile application.

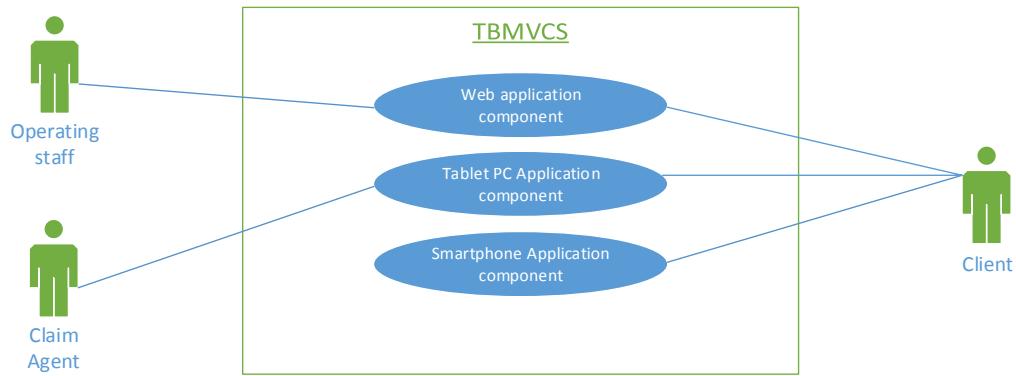


Figure 9 : Interaction of Users with Components

Source: Author

The following figure depicts the use case diagram for web application component of the solution.

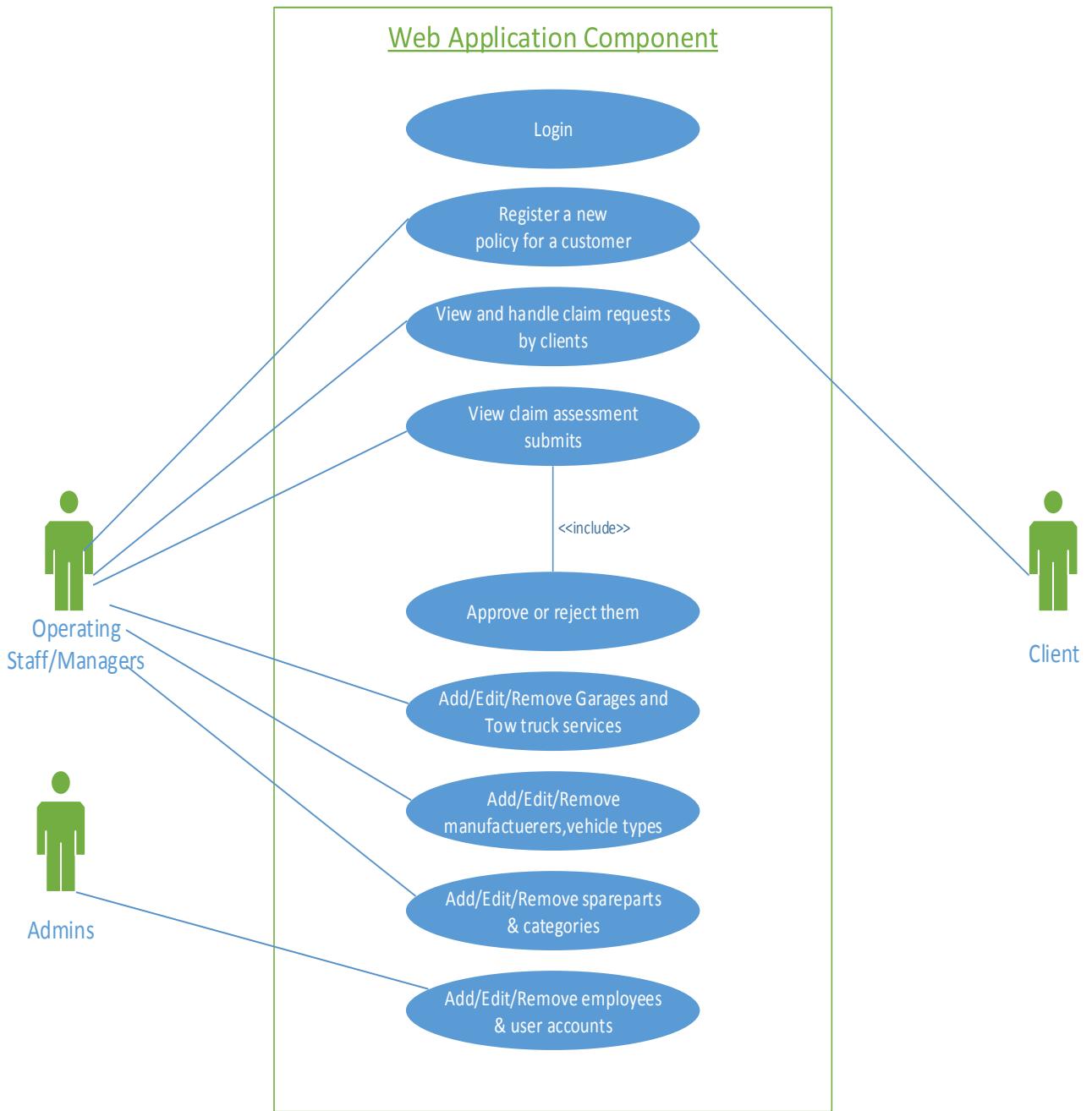


Figure 10: Use case Web Application

Source: Author

**Note:** Admins also have to login to the system in order to interact with other modules.

Next use case diagrams illustrate the tab application component of the solution and the smartphone application which clients can use.

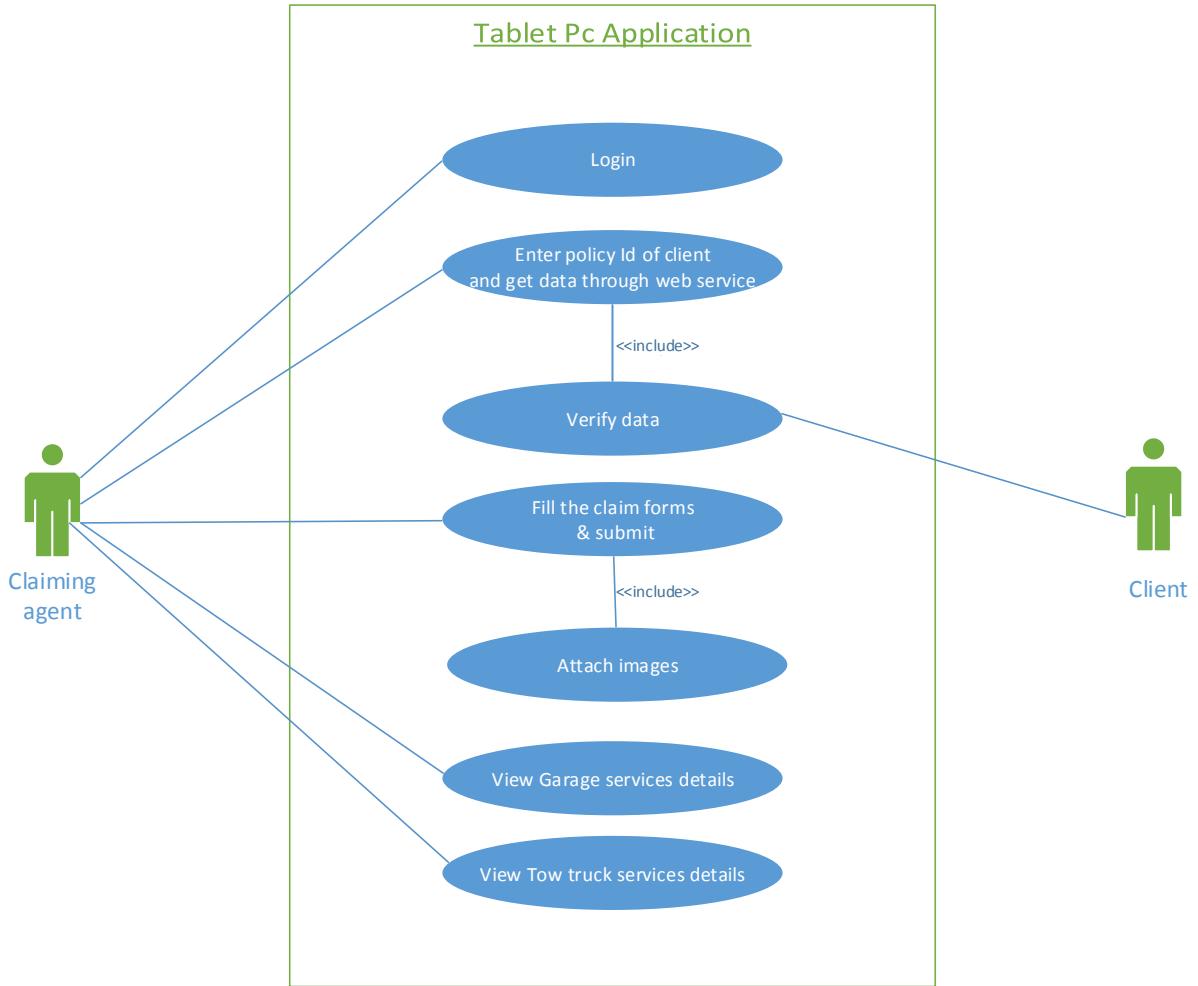


Figure 11: Use case Tab Application

Source: Author



Figure 12: Use Case for Mobile Application

Source: Author

### 5.3.2 Class Diagram for the New Solution

Class diagram expresses the various kinds of system objects and different types of relationships exist among those system objects. System objects and relationships that are involved in the proposed system are represented in the following diagram

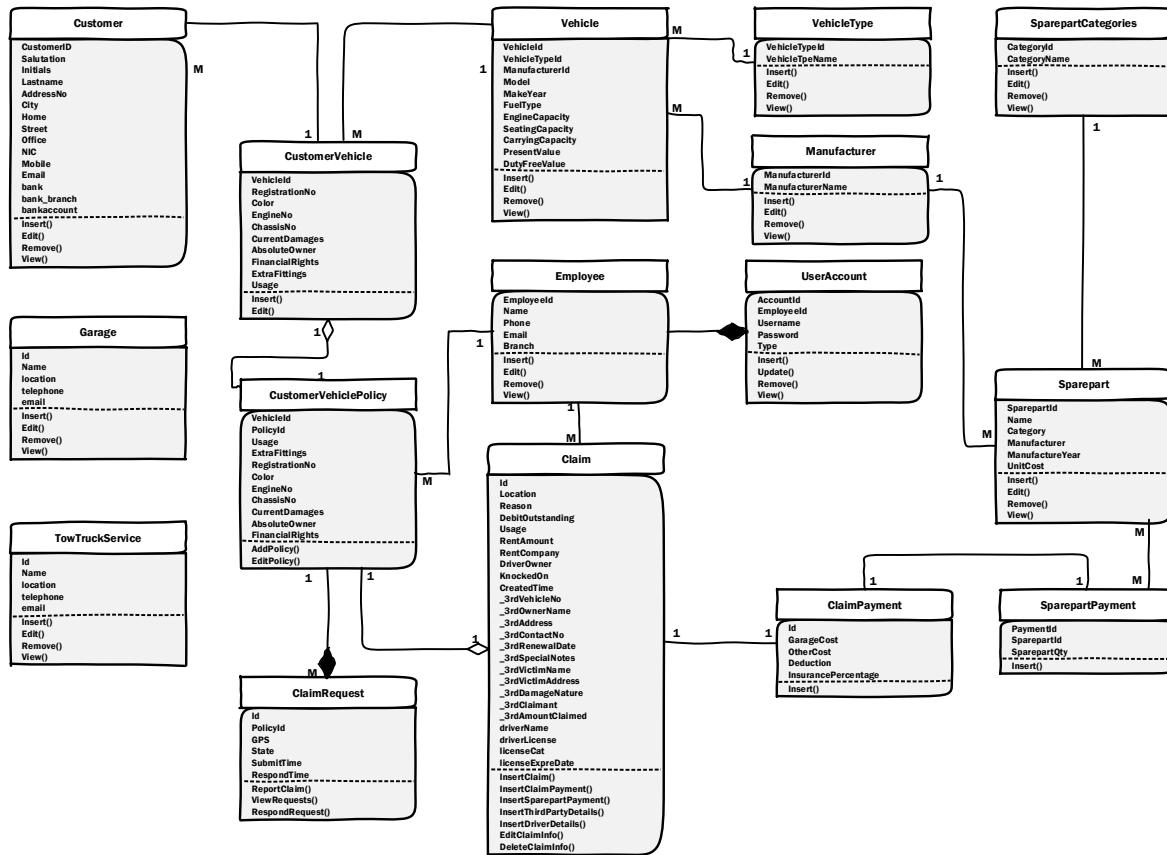


Figure 13: Class Diagram for New Solution

Source: Author

### 5.3.3 Sequence Diagrams for New Solution

Following diagrams represents how the processes in the new system operate with one another and in what order.

Following diagram illustrates sequence diagram for accessing a module in web application.

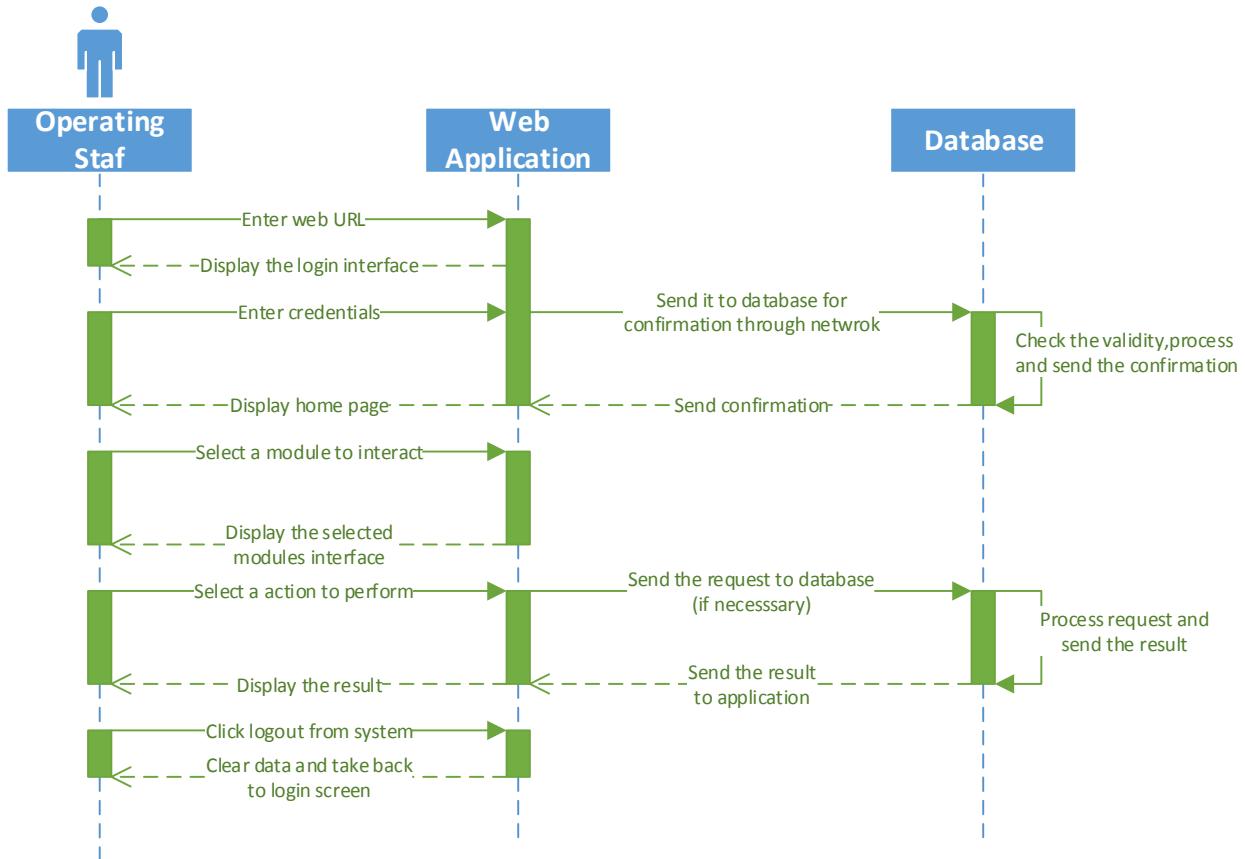


Figure 14: SQ Diagram for Accessing a Module in Web Application

Source: Author

Following diagram illustrates sequence diagram for accessing a module in smartphone application.

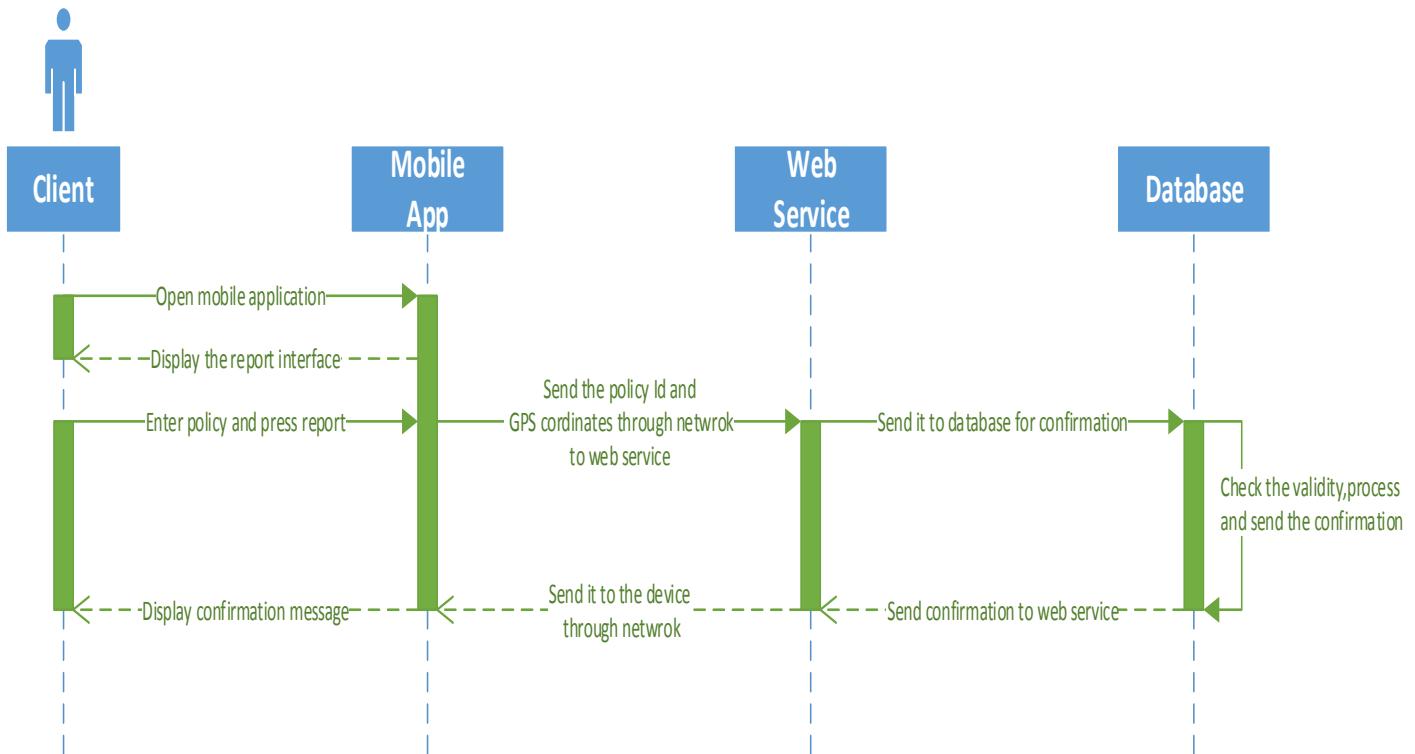


Figure 15: SQ Diagram for Accessing a Module in Mobile Application

Source: Author

Following diagram illustrates sequence diagram for accessing a module in tab application.

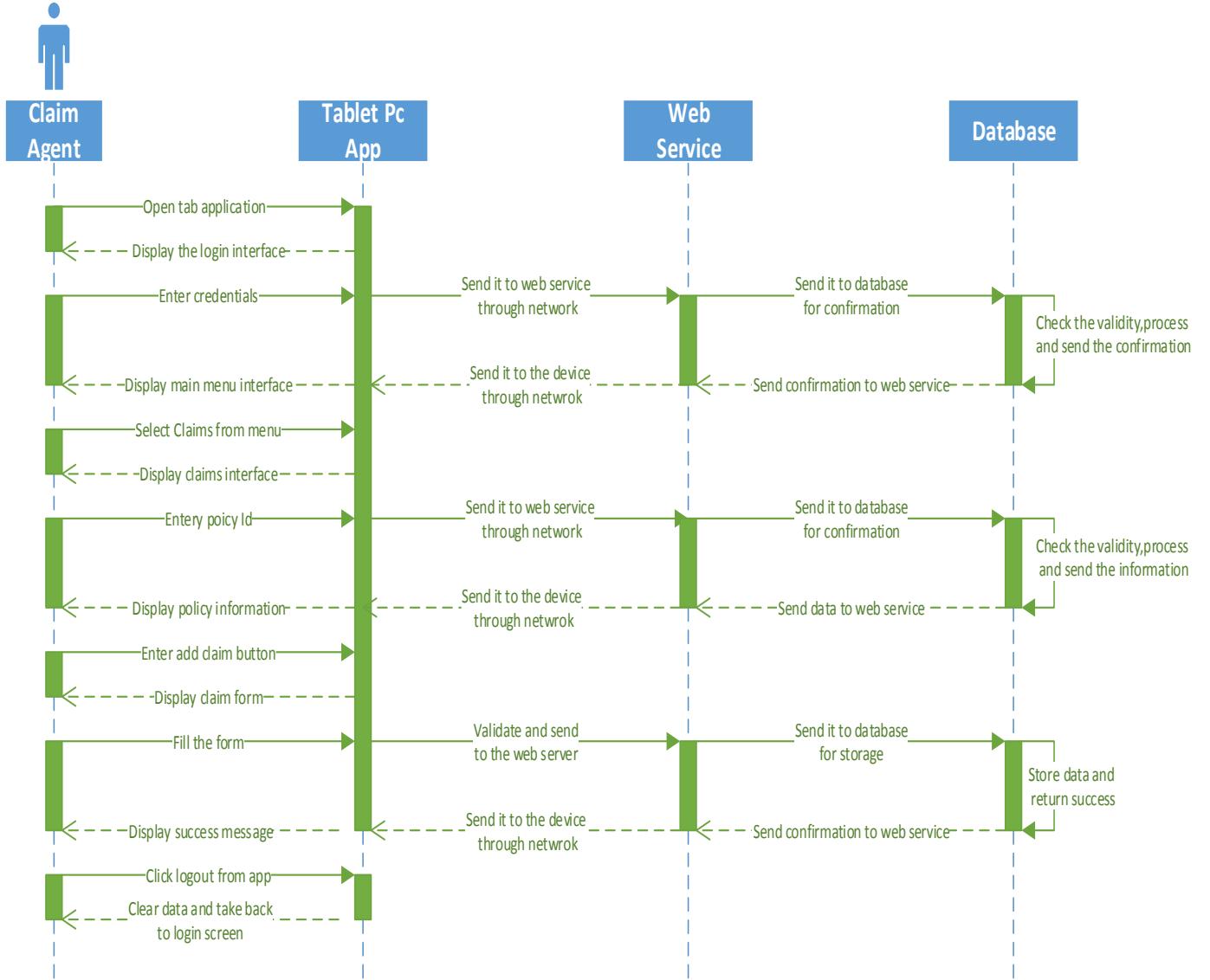


Figure 16: SQ Diagram for Accessing a Module in Tab Application

Source: Author

## 5.4 Overall System Architecture

Architectural design defines the overall structure and the connections in between components of the solution before moving on to the detail design or the low level design which includes the design of specific components details. The architectural design is given according to the three-tier-architecture where overall design is split in to three layers of Client Tier, Application Tier and Data Tier. The overall system architectural design for the proposed system is as follows.

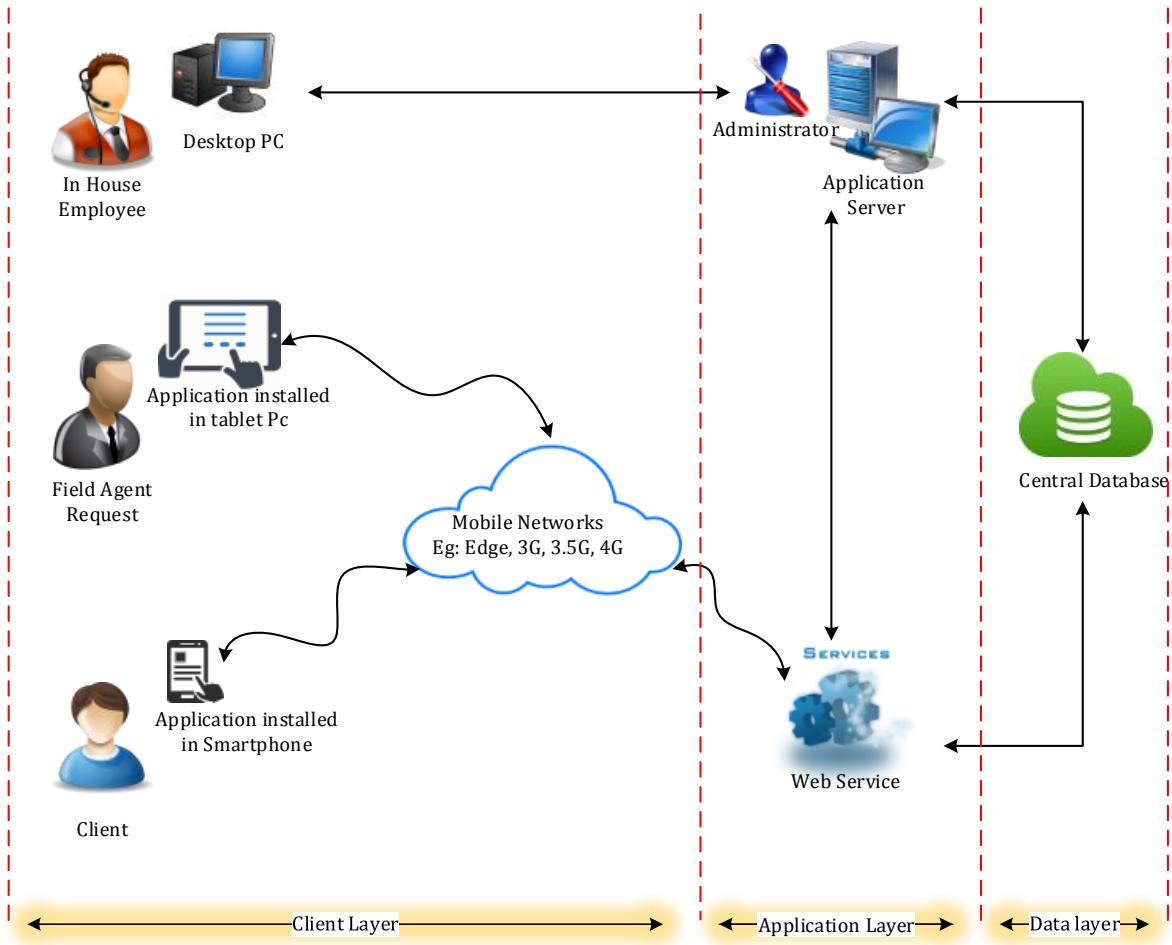


Figure 17: Overall Solution Architecture

Source: Author

As the above given design illustrates company office web application is build and hosted in a web server along with a central database. A web service is attached to the web application in order for agent tab application and client mobile app to connect with the company web application. Web service is accessible through internet or data communication media.

#### 5.4.1 Application Layer

The company logics and processes of the system will be executed at this layer in order to achieve the expected objectives of the system and can be named as the heart of the overall system. This layer will interact between application layer where the interfaces are running and data layer where the information is stored. Data gathered by user inputs or by other processes will be manipulated according to the predefined operational instructions at this layer. The client, incident and vehicle information will be applied in the application for the user to access and input data. The main components of this layer will be the web application

for ground staff and the web service hosted for the communication of tab application and phone application.

#### **5.4.2 Data Layer**

Data layer manage the data storage operations of the overall system where the database management applications are running. A central database containing the information about policies, claims, vehicles, spare parts and employees will be stored in several tables in the database in order to improve the efficiency. Tab application and the mobile application will communicate with the database through web service while web application will directly communicate with the database.

#### **5.4.4 Presentation Layer**

Presentation layer is responsible of control interactions with users by monitoring interfaces to current requested information and retrieve the inputs delivered by the user. Information gathered by this layer will be provided to the application layer in order to manipulate according to the given instructions. Main system components in this layer are tab application and the mobile application. Data added through the apps will be communicated to the company web application through the web service.

### **5.5 Software Architecture**

Software architecture was based on modularized approach where the software is divided into parts. Each module is assigned to execute one or more tasks of the overall system in order to achieve the ultimate objectives expected. The software architecture is details in the following illustration and we see the different layers of the total system.

The solution has three major user interacting components namely web application, tab application and the mobile application. Operating staffs web application will be the major data source for the database. They will be in charge with registering policies, updating vehicle details and process and also administrative purposes also. Tab application will be used to issue a claim to the client and claiming agent's needs to log into tab application to conduct the claiming process. Client mobile application is used to inform the office about the accident via web to the operating staff with its GPS coordinates. These 3 components are connected by the web service to the web application and to the database.

## 5.6 Module Architecture

Main modules and sub-modules

- Online claim management system (Web application)
  - Insurance policy module
  - Customer details module
  - Vehicle detail management module
  - Spare parts module
  - Garage services module
  - Tow truck services module
  - Client requests module
  - Employees and user account management module
  - Claim review module
  - Vehicle categories module
  - Spare part categories module
  - Manufacturers module
- Web service module
  - Connect tab application and mobile application to the web application component.
- Authentication module
  - Authentication to the web application
  - Authentication to the Tab application
- Tab application component
  - Policy information module
  - Claim history information
  - Spare part details module
  - Claim assessment module
  - Image upload module
  - Garage services module
  - Tow truck services module
- Mobile application component
  - Inform accident with GPS coordinates

In below some of the most important modules are explained at a high level using graphical representations.

## Login Module

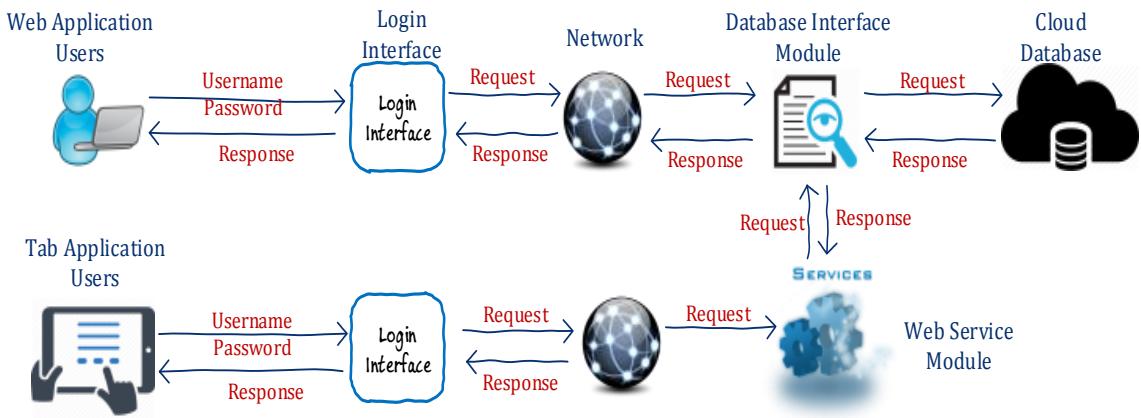


Figure 18: Login Module

Source: Author

Every type of user needs a valid account to login to the respective application. Operating staff will be given access to login to the web application while the claim agents will be given access to the tab application. Since the web application is hosted in the cloud, once the user enter their credentials data has to pass through the network and confirm it from the database and provide the confirmation back to the user through the network. Web application will communicate with database directly via database module interface while tab application will communicate through the web service and database interface module.

## Online Claim Management System (Web Application)

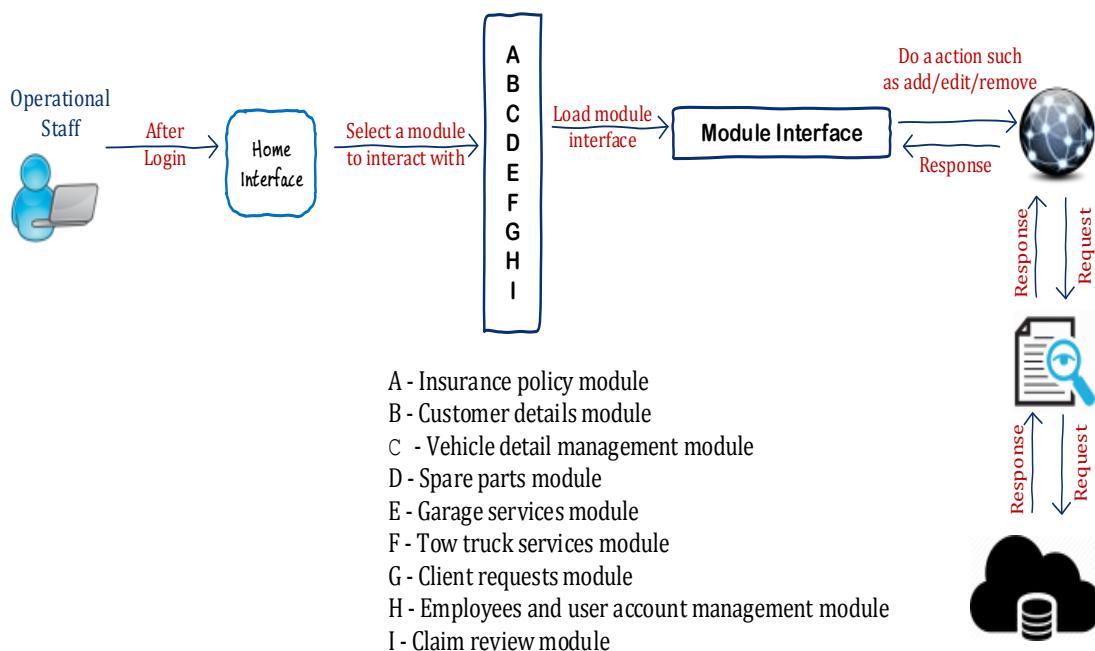


Figure 19: Web Application Modules

Source: Author

Once the user login to the web application user can choose a module to interact with, which also will be filtered according to user type. As shown in the diagram user can chose a module and o to that modules particular interface. From that interface user can coordinate any possible action such as add a record. Then via the network data will be sent to the cloud database. Similarly user can go back and choose a task to perform and deal with it accordingly.

## Tab Application Component

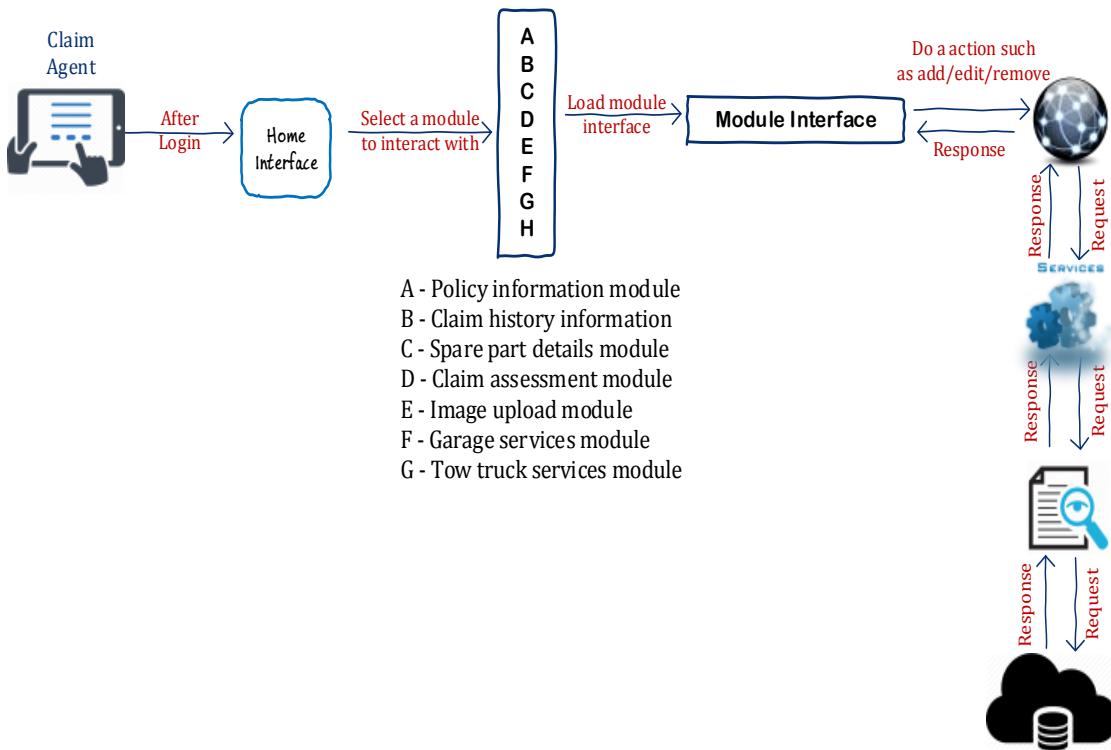


Figure 20: Tab Application Modules

Source: Author

Similarly as the web application once the user login to the web application user can choose a module to interact with. As shown in the diagram user can chose a module and o to that modules particular interface. From that interface user can coordinate any possible action such as calming procedure. When user submit data from the tab device they will be communicated to the web service through the network and web service will handle the communication with the company database.

## Mobile Application Component

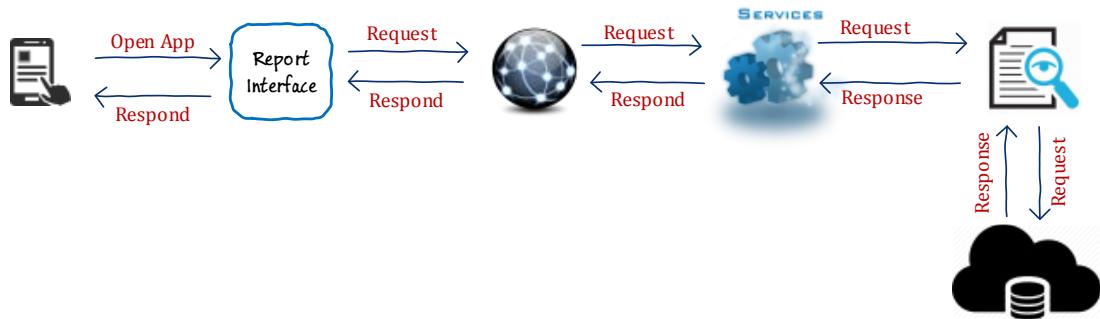


Figure 21: Mobile Application Module

Source: Author

Mobile application component is a very simple component which requires no login from the customer. Customer can open the app and enter policy number and request for a claim agent. Data will be communicated to the web service via the network and web service will handle the communication with the company database.

## 5.7 Database Design

The database was a very important in developing the solution of TBMVCS. Since we are giving a lot of assurance of data and sorting and calculation of data the database should also have a good foolproof design. As the database technologies, SQL server 2012 was used because it is easy and quick to respond for the solution and provide easy interfaces for creating tables and testing purposes. Stored procedure were used for querying in order to make the code reusable.

## 5.7.1 EER Diagram

Following figure conceptual database diagram for the solution

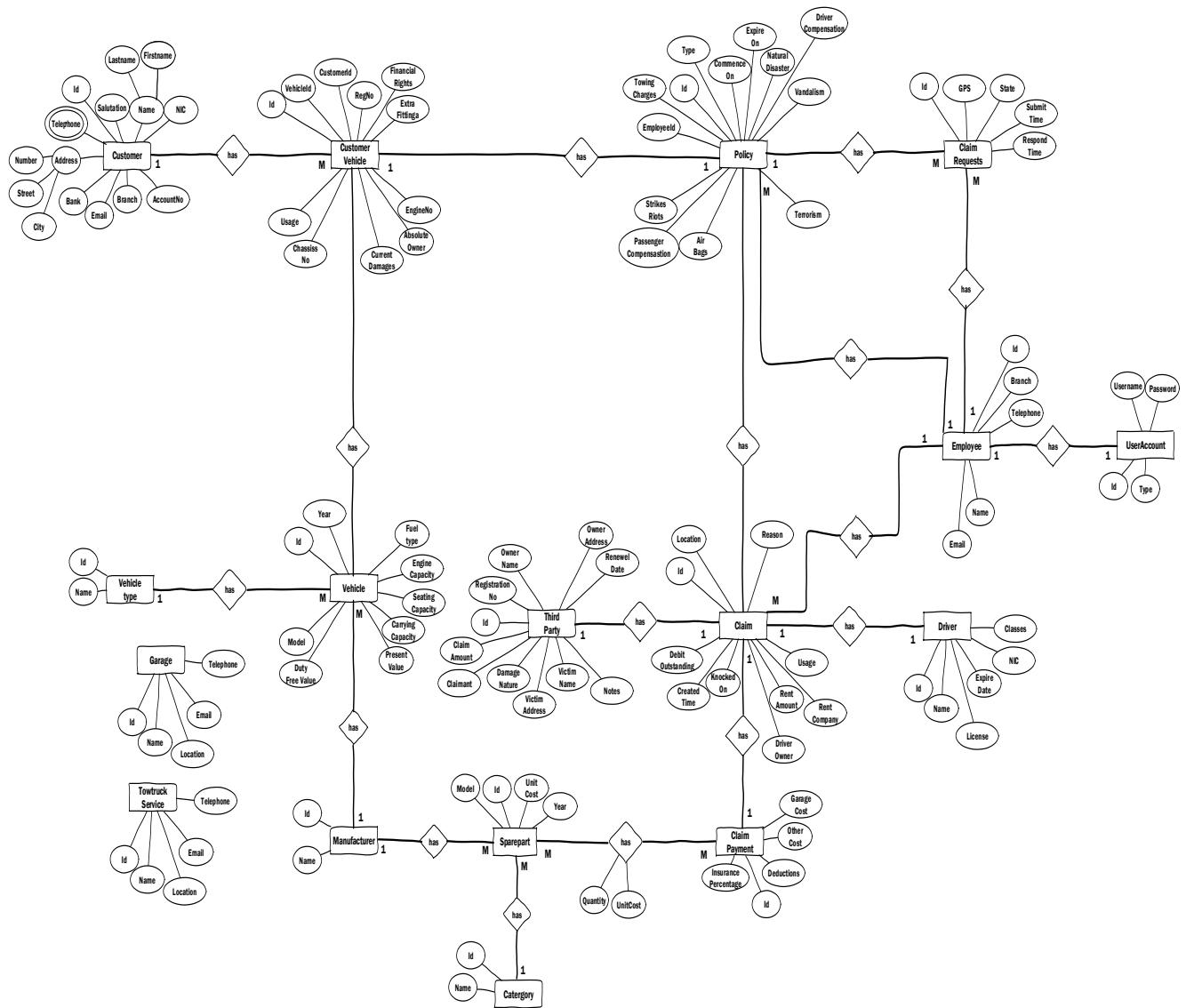


Figure 22: EER Diagram

Source: Author

## 5.7.2 Database Relationship diagram

The diagram shown in the next page indicated the relationship among tables in the database.

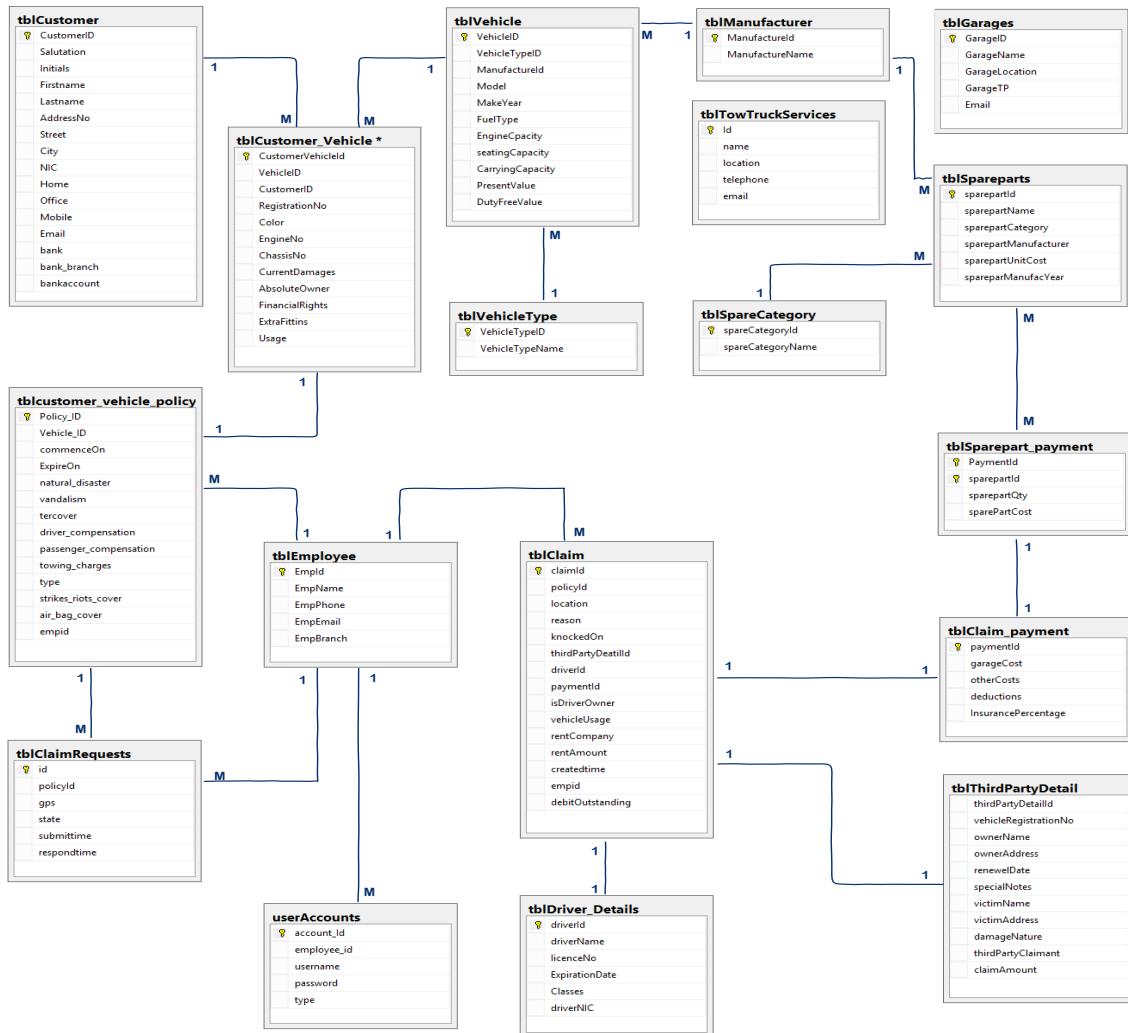


Figure 23: Database Relationship Diagram

Source: Author

### 5.7.3 Data Type Design for the Database

Given below are some of the tables in the database describing attributes and data types

Customer Table		
Attribute	Data Type	Length
CustomerId	Int	-
Salutation	nvarchar	10
Initials	nvarchar	20
Firstname	nvarchar	50
Lastname	nvarchar	50
AddressNo	nvarchar	10
Street	nvarchar	50
City	nvarchar	50
NIC	nvarchar	25
Home	nvarchar	20
Office	nvarchar	20
Mobile	nvarchar	20
Email	nvarchar	100
Bank	nvarchar	50
Branch	nvarchar	50
AccountNo	nvarchar	25

Table 1: Customer Table

Source: Author

It is important that the customer is identified by the database and that the agent does not have to depend on the third party at the scene to give the necessary customer details. The customer information is entered in this relation. The attributes can be the customer ID, the customer name, his personal details, his contact information etc. the data type and the length of such data will also be entered in the customer relation

Claim Table		
Attribute	Data Type	Length
Claimid	Int	-
Location	nvarchar	50
Reason	nvarchar	50
Knockedon	nvarchar	50
Thirdpartydeatilid	Int	-
Driverid	Int	-
Paymentid	Int	-
Isdriverowner	nvarchar	10
Vehicleusage	nvarchar	50
Rentcompany	nvarchar	50
Images	varbinary	Max
Rentamount	Float	-
Createdtime	DateTime	-
Empid	Int	-

Table 2: Claims Table

Source: Author

One of the key feature is to assess the claim form the tab application. Therefore this is one of the most important table in the database where we store all the claim details.

Customer Vehicle Policy Table		
Attribute	Data Type	Length
Policy_ID	Int	-
Vehicle_ID	Int	-
Type	nvarchar	20
Commenceon	nvarchar	15
Natural_disaster	nvarchar	20
Vandalism	nvarchar	50
Tercover	nvarchar	50

Currentdamages	nvarchar	Max
Driver_compensation	nvarchar	10
Passenger_compensation	nvarchar	50
Towing_charges	nvarchar	10
Air_bag_cover	nvarchar	10
Strikes_riots_cover	nvarchar	10

Table 3: Customer-Vehicle-Policy Table

Source: Author

Another important table which is used to store policy details of each client. This will have relationship with customer and vehicle table mainly. Full set of tables are available in the **appendix C** for your reference. Next sub section will illustrate the interface designs.

## 5.8 Interface Design

The user friendly interface designs for the developing system are shown below.

### 5.8.1 Tab application

Interfaces of the tab application are shown in this sub section.

#### Login Screen

Claim agent need to be logged on to the application first. In that case, user needs to have a valid username and password given by the admin of the system. When submitting the username and password database checks the validity of entered username and password. Then it checks the user levels and provides the window according to the user type.

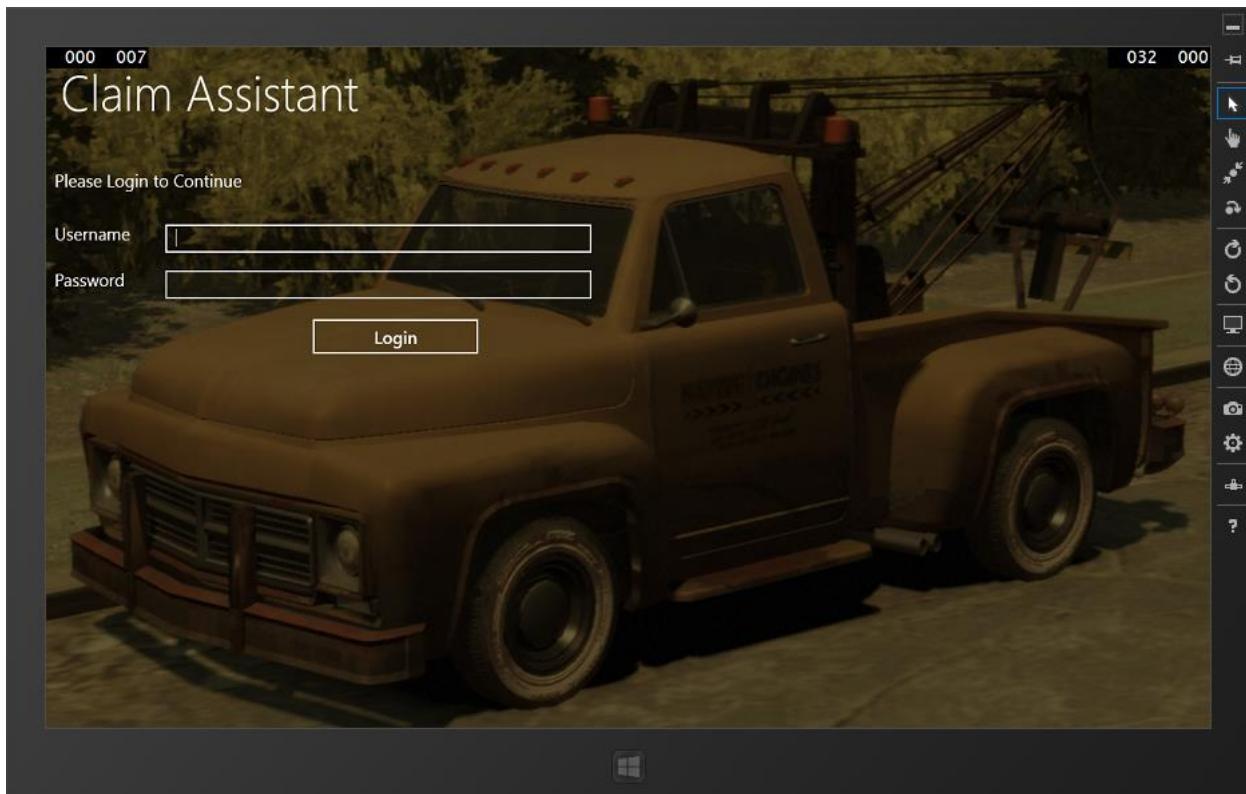


Figure 24: Login Interface of Tab Application

Source: Author

#### Main Menu

After a successful login this will be the land on interface. From here user can select the module which he may want to interact with.

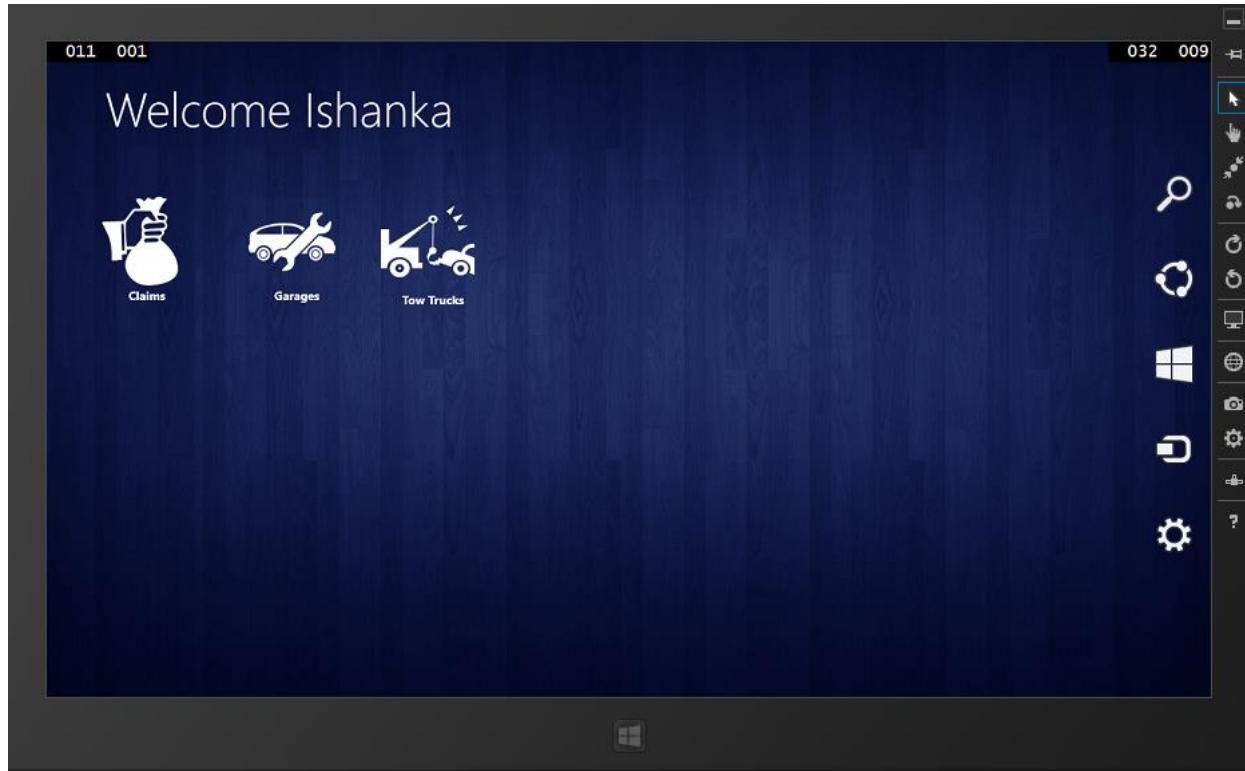


Figure 25: Main Menu of Tab Application

Source: Author

## Policy Information

This is the landing page for selecting claims from the main menu. In here claim agent will enter the policy Id of a client to get his policy details. After getting information agent have to click claim button to start the assessment.

The screenshot shows a software application window titled "Policy Information". The top status bar displays "000 001" on the left and "032 011" on the right. On the far right, there is a vertical toolbar with various icons. The main content area is divided into four sections: "Client", "Vehicle", "Policy", and "Claim History".

Section	Details
<b>Client</b>	Insured Person: Mr WML Wasalage Address: 8b Kadhwala Ratmalana NIC: 912701395v Contact No: 0716405220 Email: isankalakshan@gmail.com
<b>Vehicle</b>	Reg No: 253-3402 Color: White Engine No: 548654aa5454 Chassis No: 654987646541ad Manufacturer: Toyota Model: Allion Year: 2012 Engine Capacity: 1400cc Absolute Owner: Loan Financial Rights: Yes Usage: private Current Damages: no Extra Fitting: Spoiler Present Value: 2300000
<b>Policy</b>	Policy ID: 4 Expire On: 12/5/2018 12:00:00 AM Commenced From: 12/5/2015 12:00:00 AM Natural Disaster: No Passenger Compensation: Rs. 0 Driver Compensation: Rs. 0 Towing Charges: Rs. no Terrorism Cover: Rs. no Vandalism Cover: Rs. Yes
<b>Claim History</b>	10/31/2015 7:37:32 PM

At the bottom right, there are two buttons: "Home" and "Claim".

Figure 26: Policy Information Interface

Source: Author

## Claim Forms

After getting the policy information claim agent can start the claiming procedure by filling the series of forms in the tab application.

The screenshot shows a Windows application window titled "New Claim". The top status bar displays "006 002" on the left and "032 015" on the right. On the right side of the window is a vertical toolbar with various icons. The main form is divided into two main sections: "Driver Details" on the left and "3rd Party Damages" on the right.

**Driver Details**

- Is the driver owner?  
Yes  No
- Driver Name: [Text Box]
- License: [Text Box]
- Categories: [Text Box]
- Expire Date: [Select Box] November 22 2015
- NIC: [Text Box]
- Date of Purchase: [Select Box] November 22 2015
- Vehicle used for?: [Text Box]

**3rd Party Damages**

- Victim Name: [Text Box]
- Address: [Text Box]
- Damage Nature: [Text Box]
- Third party Claimant: [Text Box]
- Amount Claimed Rs.: [Text Box]

**Buttons:**

- Next

Figure 27: Claim Form

Source: Author

This screenshot shows the same "New Claim" application window as Figure 27, but with different data entered or visible.

**Cause of Damage**

- \*Location: [Text Box]
- \*Reason: [Text Box]
- Custom: [Text Box]
- \*Knocked On: [Text Box]
- Custom: [Text Box]

**3rd party vehicle details**

- \*Vehicle No: [Text Box]
- \*Owner Name: [Text Box]
- Address: [Text Box]
- Contact No: [Text Box]
- \*Renewal Date: [Select Box] November 22 2015
- Special Notes: [Text Box]

**Buttons:**

- Next

Figure 28: Claim Form

Source: Author

The screenshot shows a Windows application window titled "New Claim". At the top left are status bars with "005 002" and "032 015". On the right is a vertical toolbar with icons for zoom, search, refresh, and other functions. The main area has a dark blue header with the title "New Claim" and a back arrow icon. Below the header is a section labeled "Attach Photos" with a "Choose Image" button. To the right is a "Payments" section with several input fields:

Sparepart Category	Manufacturer
<input type="text"/>	<input type="text"/>
Sparepart	Quantity
<input type="text"/>	<input type="text"/>
<input type="button" value="Remove"/> <input type="button" value="Add"/>	
Spare parts total cost	<input type="text" value="34 000"/>
Garage Costs	<input type="text"/>
Any Other Costs	<input type="text"/>
Amount Payable	<input type="text" value="30 000"/>

At the bottom right is a "Submit" button.

Figure 29: Claim Form

Source: Author

## Success Screen

On the successful submit of a claim following screen will be displayed,

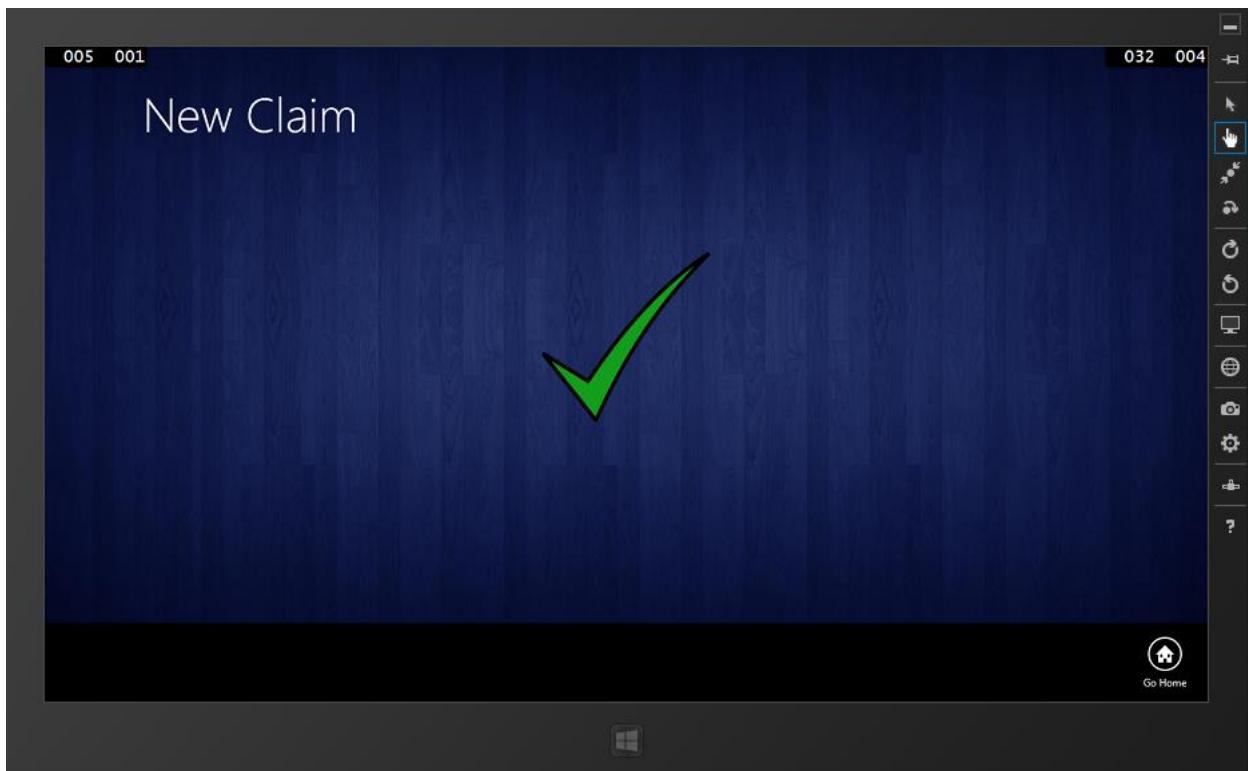


Figure 30: Claim Form

Source: Author

## Garage Services

Following interface is used to display garage services available to the claim agent. Agent can search garages using the location.

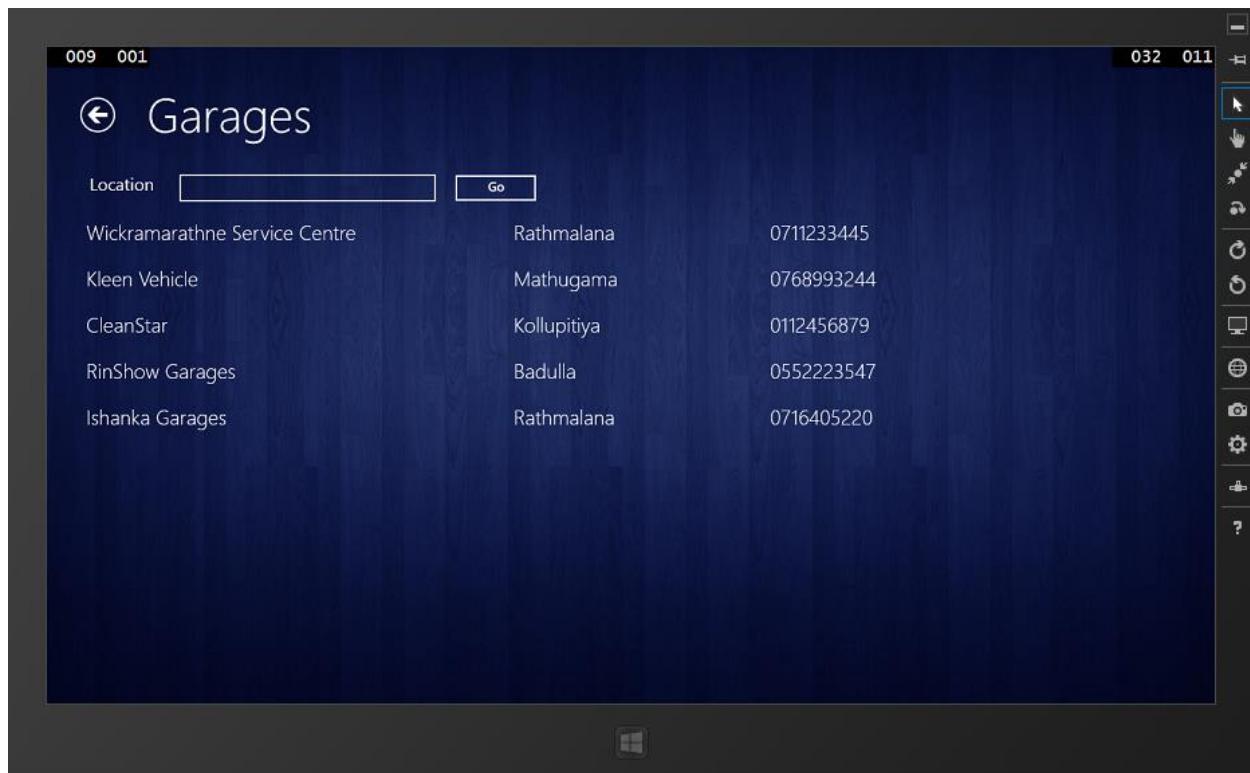


Figure 31: Garages Interface

Source: Author

## Tow Truck Services

Following interface is used to display tow truck services available to the claim agent. Agent can search garages using the location.

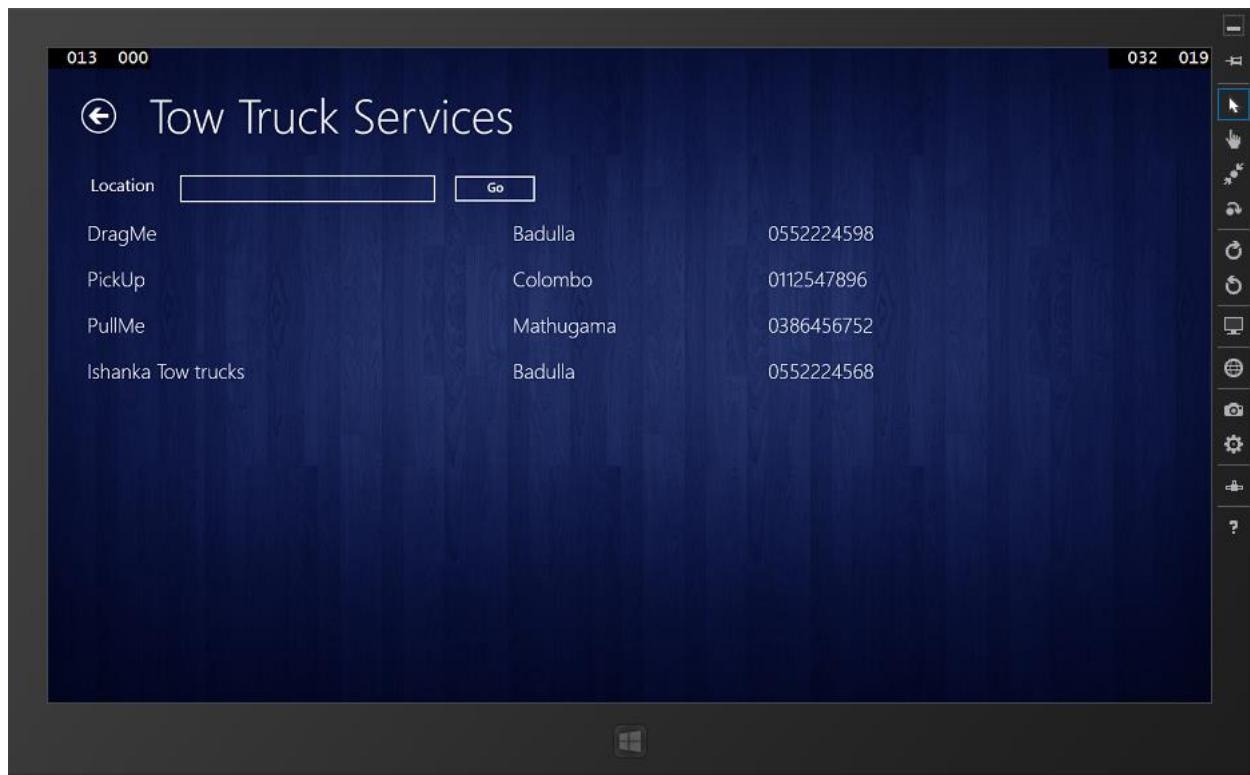


Figure 32: Tow Truck Services Interface

Source: Author

### 5.8.2 Smartphone Application

This will be the landing page when application is launched. Client will be able to enter the policy id and report it to the central system.



Figure 33: Mobile Application Interface

Source: Author

### 5.8.3 Web application

Some interfaces of the web application component are shown in this sub section.

#### Login Interface

Users need to be logged on to the application first. In that case, user needs to have a valid username and password given by the admin of the system. When submitting the username

and password database checks the validity of entered username and password. Then it checks the user levels and provides the window according to the user type.

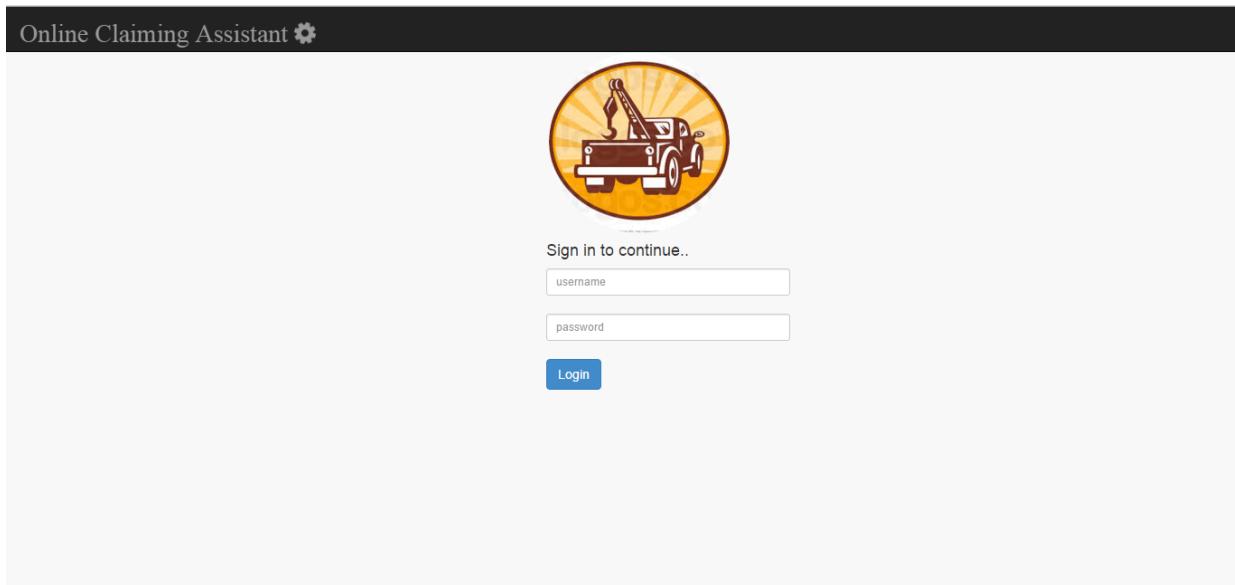


Figure 34: Login Interface of Web Application

Source: Author

## Home Interface

This will be the landing page upon a successful login to the online claim assistant system.

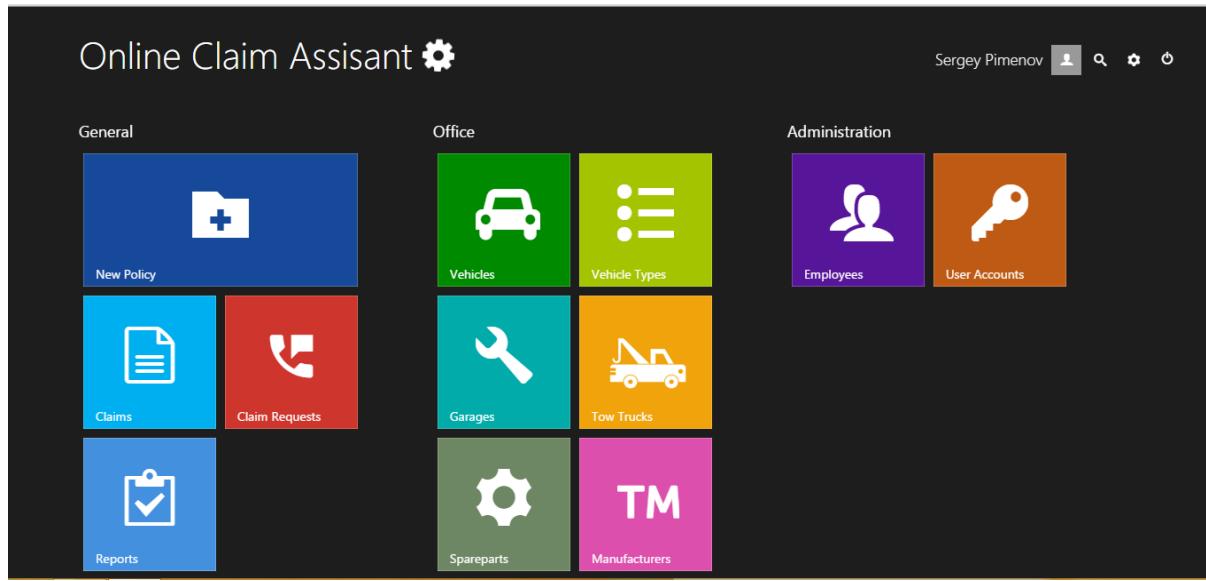


Figure 35: Home Page Interface

Source: Author

### Spare Part Module Interface

Following figure is a sample screen for a module screen. Shown screen is the spare parts modules interface. Other interfaces are shown in the appendix.

The screenshot shows the 'Spareparts' section of the Online Claiming Assistant. On the left, a sidebar lists various menu items: Home, New Policy, Claims, Claim Requests, Vehicles, Vehicle Types, Garages, Tow Trucks, Spareparts (which is selected and highlighted in blue), Manufacturers, Categories, Employees, and User Accounts. The main content area has a title 'Spareparts' with a gear icon. Below it are three buttons: 'Add' (blue), 'Edit' (orange), and 'Remove' (red). A search bar labeled 'Enter Name to Search' is present. A table titled 'Spareparts' displays data with columns: Category, Manufacturer, Model, Model Year, and Unit Cost. The table contains three rows:

Category	Manufacturer	Model	Model Year	Unit Cost
Engine	Hero Honda	Caps	2010	2010
Internal	Honda	Radiator	2011	35000
Windscreen	Toyota	Windscreen	2015	13000

Figure 36: Spare parts Interface

Source: Author

## Data Adding Form

When user need to add new records to the database, the following shown pop up modal will be used.

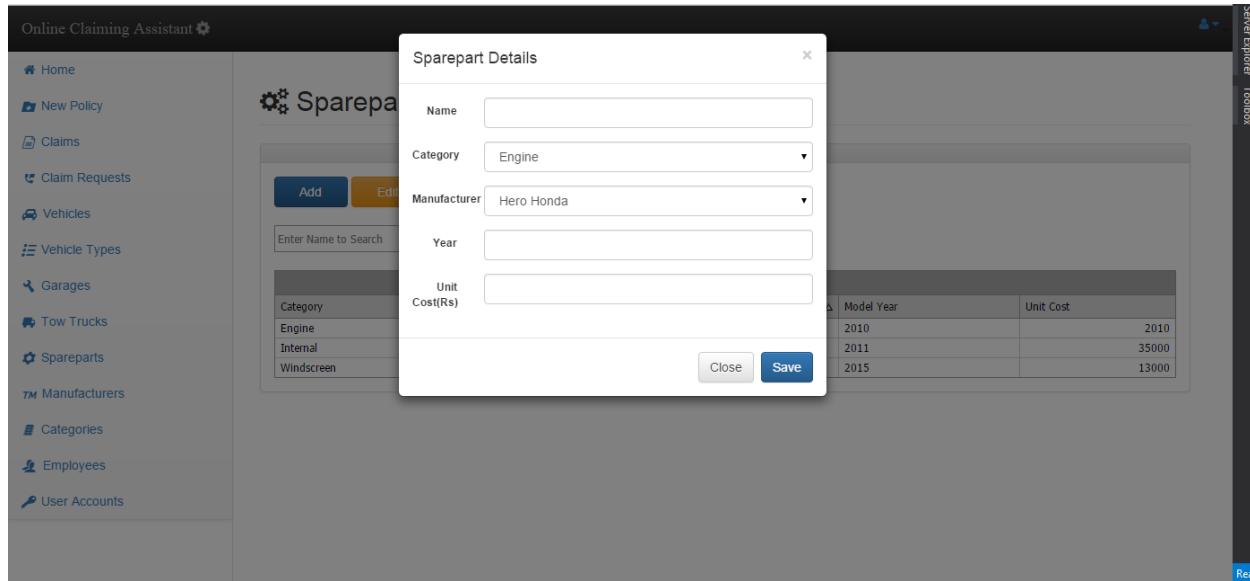


Figure 37: Add Spare part Form

Source: Author

Some of other interfaces are attached in the **Appendix C**.

## 5.9 Summary

In this chapter we started with data gathering and analysis then created an outlook to the design of the MVCS system. We went through the total system architecture and the software design to find out how this system runs. Then we have discusses how the system is broken down to modules and sub modules and what these module do. While discussing the modules we have briefly gone through all the actors participating in the implementation and the process of the system and what they have to do to keep the system running. Then we spoke of the database design on general terms and the interface screen shots were given later. With this chapter anyone could have a general feeling of what the system is or what the system is going to be.

# Chapter 6

## Development

### 6.1 Introduction

This chapter is dedicated to discuss the development process of the solution. Starting with the development methodology, areas such as language and tool selection. Later on architectures and coding standards also have been discussed. Several coding samples also provided from the real project.

### 6.2 Software Development-Technology Considerations

In order to develop a successful system which meets the end user expectations, it is very important to use appropriate tools in the development process. Use of any inappropriate tools will only lead to develop a system with unnecessary errors and faults and it can affect the user expectations in efficiency and reliability of the solution. It is very vital to use appropriate computer language and any other necessary tools in order to develop a successful system. So these technologies and tools will help to develop the system within a minimum development time. The main objective of TBMVCS is to do a fast, easy, accurate claiming process via tablet pc. And this is done by connecting to the company database via web service with the use of mobile networks. So it is very important to consider some factors such as platform supportiveness and the efficiency of the system. And in order to meet above mentioned factors we must use the most appropriate tools available in the market to develop the system.

Technological considerations - followed during the development of the system

- Efficiency and Performance
- Re-usability and flexibility
- Object oriented development support

#### 6.2.1 Development Language Selection

In modern programming field there are many programming languages and many more are introducing day by day. Core function of the system is vehicle claiming through the tablet pc, researchers have been done on many software development languages in order to recognize the appropriate language that helps efficient data retrieval and updating from the company

database via web service. Since the entire solution has a 3 main components namely the ground office system, web service and tablet application it was vital to select a language that will support all these three feature so that it will reduce the learning time of different languages. Several aspects such as build powerful web based applications, powerful, flexible, Simplified Data Access, platform dependency, build fast mobile applications were considered.

For the development of ground office system main technology considerations were ASP.net platform, java platform and PHP platform. Java and asp.net provide greater object oriented programming feasibility while PHP work as a fast backend server scripting language. Visual Studio IDE provide ASP.NET development environment with rich inbuilt functionalities for faster programming. As for the web service it was considered about coding a Visual studio web service, PHP web service and also WCF web service. WCF was considered to be a far better option than other web service types Microsoft has provided.

Another key part of the solution was the development of tablet applications. Main platform considerations were android, windows and IOS. In order to develop an IOS app it requires to learn Objective C language and it also need a Mac computer which are far more expensive than the others. Android is an open source platform with free available tools. There are lot of resources available for android development as well. Windows tablet pc was feasible to build on the same environment which was used to build the ground office system and the time taken for the learning curve was fairly less than android. Visual studio provides an in built emulator for testing purposes as well.

After considering several options ASP.Net was selected as the main programming platform to develop the ground office system since ASP.Net provides an easier and stable environment to create great web applications and it supports object oriented development greatly. Although it has to be purchased for commercial use, it can be purchased for free for university students from Dream spark. Bootstrap, a free CSS library was used to design the interfaces. C# programming language was used to implement the business logics in the backend of the system.

As for the initial step tablet application was developed for windows surface tab. It is using XAML for interface development and C# for backend coding. To connect the Tab application and the web system, WCF web service technology was selected since it can be developed using the same tools which are used to develop the system. Visual Studio 2013 was selected as the main tool for developing the solution while using open source text editors such as brackets were used wherever necessary.

## 6.2.2 Database Selection

Database which is most important aspect of the overall system which handles all the relevant details related to insurance as well as claims. In order to fulfill the need of database operation this system has been used SQL Server Management Studio which enables to access and

manage the database engine. Another reason for using this was it was free to purchase using Dreamspark account. Management Studio brings graphical tools for database management together with a rich development environment. Database is maintained at a central cloud server such that it can be accessed from wherever necessary.

### **6.2.3 Development Tool Selection**

After selecting the development language and database type necessary tools were selected for the development. This includes tools for build and debug environment, integrated development environments, database servers, plugins and frameworks and so on. Some of the key tools which were used are described below.

#### **Visual Studio 2013 Premium Version**



Visual Studio was used as the main development environment and IDE for the web application, tab application, web service and the smartphone application. In order to test and emulate the tab application and smartphone application separate emulate tool set provided by Microsoft was installed. This provide very rich tools for debugging which made the developer task less complex. This was downloaded for free from Microsoft Dreamspark.

#### **DevExpress Developer Framework**



DevExpress framework was integrated to the visual studio since it provided very powerful tools for representing and manipulating data grids. DevExpress is a software product company that providing the UI controls to the Borland Delphi, and Active X controls to the Visual Studio to make easy development and used for high quality user experience.

## **Gimp**



This was used for photo editing and icon creating works. It is the Free and Open Source, cross-platform image editor available for GNU/Linux, OS X, Windows and more. GIMP provides the tools needed for high quality image manipulation. From retouching to restoring to creative composites, the only limit is your imagination.

## **Bootstrap**



Bootstrap was used for front end development and styling purposes. With Bootstrap, you get extensive and beautiful documentation for common HTML elements, dozens of custom HTML and CSS components, and awesome jQuery plugins. It is an open source styling library.

## **Microsoft Visio 2013**



Microsoft Visio 2013 is used to data modeling, develop the diagrammatical designs and data analyzing designs to the project. Mostly used in the designing stage of the project as well as system analyzing stage.

## **Font Awesome Icon Font**

# Font Awesome

Font Awesome gives you scalable vector icons that can instantly be customized — size, color, drop shadow, and anything that can be done with the power of CSS. This is an iconic font designed for Bootstrap

## **Metro UI CSS & JavaScript Frame Work**



The front-end framework for developing projects on the web in Windows Metro Style. Metro UI CSS developed with the advice of Microsoft to build the user interface and include: general styles, grid, layouts, typography, components, and built-in icons. Metro UI CSS is open source and has MIT licensing model.

## **SQL Server 2012**



In the Enterprise industrial level with the .Net develop framework based development and most recommended database engine is Microsoft SQL Server. As the current requirement to develop the application, .Net framework has been used. According to that, as better database environment, SQL Server has been selected.

## **Visual Studio Code**



Visual Studio Code is a code editor redefined and optimized for building and debugging modern web and cloud applications. Visual Studio Code is free and have stylish UI design which offer various text editing options.

### **6.3 Development Approach**

A custom development has been selected as the development strategy in order to develop the system. This method has been selected after considering some vital factor such as complexity of the system, development time, etc. Main reason of using this development strategy is building the system interfaces at initially which will be used to develop system prototypes and develop the databases with accordance with interfaces in order to develop the web service in a fast manner. Then the tab application was developed while developing the some modules of web application as well.

Components that has been recognized as vital and to be built at first are,

- Database Development
- Interfaces of all components
- Policy information module and vehicle spare part module of web application
- Web service
- Tab application
- Claim approval module
- Rest of the web application components
- Mobile application

### **6.4 Database Development**

SQL server 2012 was used for database related activities. All the tables were created in the design view and a naming convention was used to identify the tables easily. Following figure shows the tables in the SQL server views.

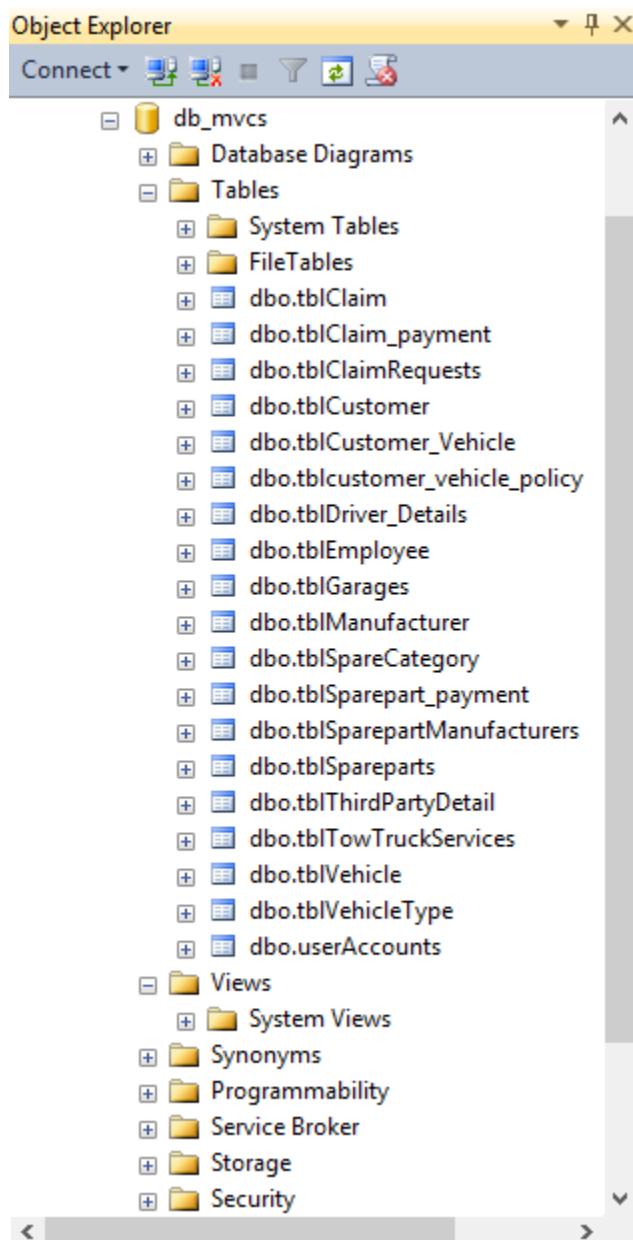


Figure 38: SQL Server Tables

Source: Author

In order to query to the database tables stored procedures has being used in the entire web application. A stored procedure is simply a way to put blocks of code inside the database. Stored procedures are compiled once and stored in executable form, so procedure calls are quick and efficient. Executable code is automatically cached and shared among users. This lowers memory requirements and invocation overhead. Stored procedures increase scalability by isolating application processing on the server. Following figure shows some of the stored procedures declared in the application.

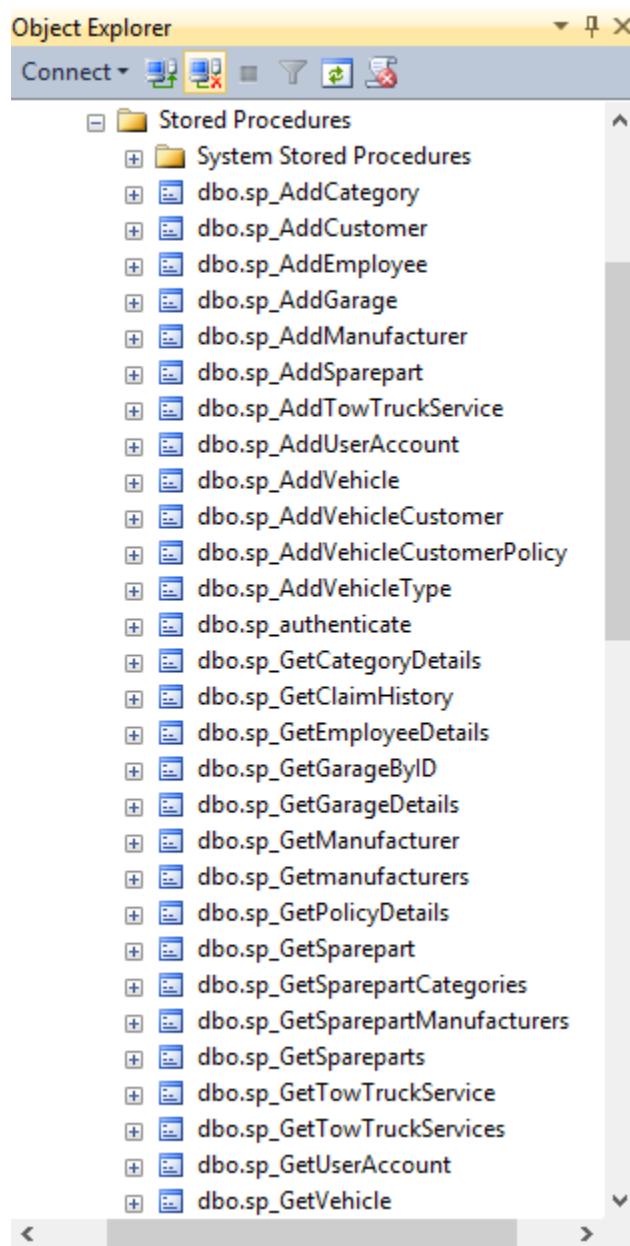


Figure 39: SQL Server Stored Procedures

Source: Author

Below shown is an example stored procedure taken from this solution.

```
USE [db_mvcs]
GO
***** Object:  StoredProcedure [dbo].[sp_AddGarage]      Script Date: 11/28/2015 5:07:30 PM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

ALTER PROCEDURE [dbo].[sp_AddGarage]
    -- Add the parameters for the stored procedure here
    @GarageName NVARCHAR(200),
    @GarageLocation NVARCHAR(50),
    @GarageTP NVARCHAR(50),
    @Email NVARCHAR(100)

AS
BEGIN
    SET NOCOUNT ON;

    -- Insert statements for procedure here
    INSERT INTO tblGarages(GarageName, GarageLocation, GarageTP, Email)
    VALUES (@GarageName, @GarageLocation, @GarageTP, @Email)
END
```

Figure 40: Sample Stored Procedure

Source: Author

## 6.5 Web Application Development Architecture

A layered architecture approach has been adopted when developing the web application. WCF web service also has been added as a layer in the project. Therefore main layers in the web application solution are data layer, business layer, model layer, view layer and the WCF service layer.

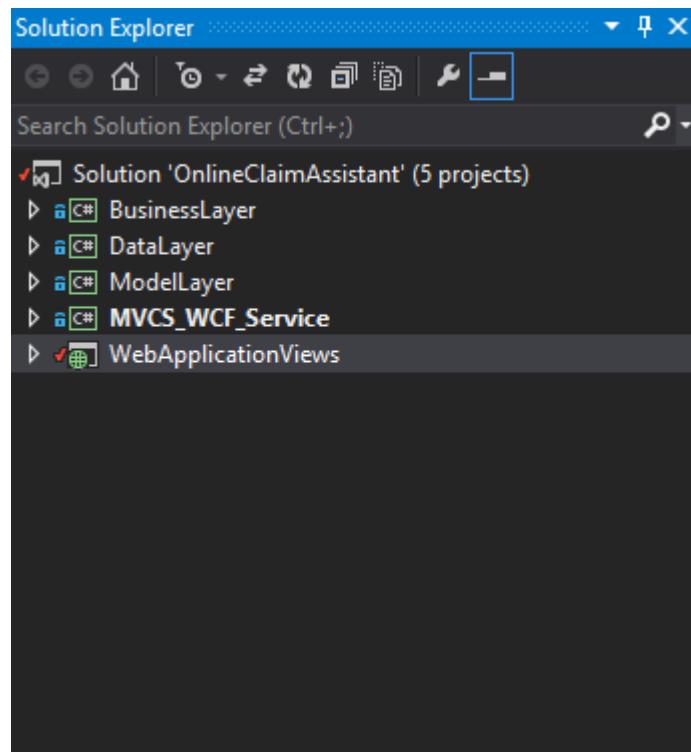


Figure 41: Project Structure

Source: Author

### 6.5.1 View Layer

This layer is the visible layer for end users. All the front end designs are coded in this layer. Views are created with aspx extension web forms connecting to a single master page. In the master page all the libraries such as bootstrap, font awesome are imported using their CDN libraries.

```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.4.0/css/font-awesome.min.css"/>
<link rel="stylesheet" href="css/style.css"/>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css"/>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap-theme.min.css"/>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.min.js"></script>
```

Figure 42: CDN Library Links

Source: Author

Basic html and asp.net tags have been used to develop the front end. In the code behind communication with the model layer and business layer are maintained. All the button click events are coded in code behind using c#.

```
<%@ Page Title="" Language="C#" MasterPageFile "~/Site1.Master" AutoEventWireup="true" CodeBehind="Garages.aspx.cs" Inherits="WebAp
<%@ Register Assembly="DevExpress.Web.v15.1, Version=15.1.7.0, Culture=neutral, PublicKeyToken=b88d1754d700e49a" Namespace="DevExpr

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="cpPageContainName" runat="server">
    <i class="fa fa-wrench"></i>&ampnbspGarages
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="cpFprmContainName" runat="server">
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="cpForm" runat="server">
    <br>
    <asp:ScriptManager runat="server"></asp:ScriptManager>
    <div class="form-inline">
        <button type="button" class="btn btn-primary horizontal-bar" id="btnAdd" runat="server" onserverclick="btnAdd_ServerClick">
            Add
        </button>
        <button type="button" runat="server" class="btn btn-warning horizontal-bar" id="btnEdit" onserverclick="btnEdit_ServerClick">
            Edit
        </button>
        <button type="button" class="btn btn-danger horizontal-bar" data-toggle="modal" data-target="#DeleteModal">
            Remove
        </button>
    </div>

```

Figure 43: Sample from Front End

Source: Author

```

public Garage_ML GarageEditID;

protected void Page_Load(object sender, EventArgs e)...

private void GetGarageData()...

private void CloseModal()...

protected void txtLocationSearch_TextChanged(object sender, EventArgs e)
{
    GetGarageData();
}

protected void btnSave_ServerClick(object sender, EventArgs e)
{
    try
    {
        var ml = new Garage_ML()
        {
            GarageName = txtName.Text,
            GarageTP = txtTp.Value,
            GarageLocation = txtLocation.Value,
            Email = txtEmail.Value
        };
        var result = new Garage_BL().AddGarage(ml);
        if (result)
        {
            GetGarageData();
        }
    }
    catch (Exception)
}

```

Figure 44: Sample from Code behind

Source: Author

### 6.5.2 Model Layer

Model layer was used to define the model of the available class objects. Defaults constructor as being overridden in order to set values in the instance of creating objects. After creating an object of a particular class it's being passed as parameters for other layers when processing. A sample of a model class is illustrated in the below code.

```

namespace ModelLayer
{
    public class Vehicle_ML
    {
        public int VehicleID {get;set;}
        public int VehicleTypeID {get;set;}
        public int ManufactureId {get;set;}
        public string VehicleTypeName { get; set; }
        public string ManufactureName { get; set; }
        public string Model {get;set;}
        public string MakeYear {get;set;}
        public string FuelType {get;set;}
        public string EngineCapacity {get;set;}
        public string seatingCapacity {get;set;}
        public string CarryingCapacity {get;set;}
        public float PresentValue {get;set;}
        public float DutyFreeValue { get; set; }

        public Vehicle_ML(){ }

        public Vehicle_ML(string id, string typeid, string manid, string model, string makeyear,
                          string fueltype, string engincap, string seatcap, string carrycap, string presentval, string dutyval)
        {
            if (id!=null)
            {
                VehicleID = Convert.ToInt32(id);
            }
            if (typeid != null)
            {
                VehicleTypeID = Convert.ToInt32(typeid);
            }
            if (manid != null)
        }
    }
}

```

Figure 45: Sample Class from Model Layer

Source: Author

### 6.5.3 Business Layer

This layer handles all the business logics and communications with the data layer. All the parameters required for a transaction with the database are provided at the business layer.

```

using ModelLayer;
using DataLayer;

namespace BusinessLayer
{
    public class Garage_BL
    {
        public DataTable GetGarageData(Garage_ML ml)...
        public bool DeleteGarage(Garage_ML ml)...
        public bool AddGarage(Garage_ML ml)
        {
            try
            {
                var DataDic = new Dictionary<string, object>
                {
                    {"@GarageName", ml.GarageName},
                    {"@GarageLocation", ml.GarageLocation},
                    {"@GarageTP", ml.GarageTP},
                    {"@Email", ml.Email}
                };
                return new DBAccessController().InsertRecord(StoredProcedures.sp_AddGarage, DataDic);
            }
            catch (System.Exception)
            {
                throw;
            }
        }
    }
}

```

Figure 46: Sample Class from Business Layer

Source: Author

#### 6.5.4 Data Layer

Data layer is the responsible layer for communication with the database. All the constants need for are stored in a separate class called Constants. This layer contains the data base connection class as well. Data connection string is stated in the web.config file so that it can be easily changed in the implementation phase.

```

<connectionStrings>
    <add name="DBConnection" connectionString="Data Source=3UDDY\SQLEXPRESS;Initial
        Catalog=db_mvcs;Integrated Security=True" providerName="System.Data.SqlClient" />
    <add name="DBConnectionLaptop" connectionString="Data Source=3UDDY-LAPTOP;Initial
        Catalog=db_mvcs;Integrated Security=True" providerName="System.Data.SqlClient" />
</connectionStrings>

```

Figure 47: Connection Strings

Source: Author

It will be inherited by the database access controller class. Following figure shows a sample code from a data layer class. This layer classes are developed such that it can be reusable as a DLL reference for a different project.

```

public class DBAccessController : DBConnection
{
    public DataTable RetrieveAllRecords(string storedProcedureName)
    {
        var dt = new DataTable();
        try
        {
            con = new SqlConnection(connectionString);
            con.Open();

            cmd = new SqlCommand(storedProcedureName, con) { CommandType = CommandType.StoredProcedure };
            adp = new SqlDataAdapter(cmd);
            adp.Fill(dt);
            return dt;
        }
        catch (System.Exception)
        {
            throw;
        }
        finally
        {
            con.Close();
            con.Dispose();
        }
    }
}

```

Figure 48: Sample Code from Data Layer Class

Source: Author

### 6.5.5 WCF Web service

This is the layer where the web service is created for the communication with tab application and mobile application. Web service will also communicate with the database through data layers. In the service there is an interface class also to abstract the exposed methods. In the service class all the implementations of the methods are done. In web services simply method calls are done.

```

namespace MVCS_WCF_Service
{
    [ServiceBehavior(IncludeExceptionDetailInFaults = true)]
    public class Service1 : IService1
    {
        public string GetData(int value)...
        public string Authenticate(string username, string password)
        {
            return new UserLogin_BL().UserAuthentication_App(new UserLogin_DL() { username = username, password = password });
        }
        public string GetPolicyInfo(int policy_ID)...
    }
}

```

Figure 49: Sample Code from Web Service

Source: Author

### 6.5.6 Communication between Layers

This section will discuss how the communication will happen between the layers in the time of execution. Let's consider a situation where we are in add a garage user interface and we are adding a new garage in to database. When the save button is clicked from the front end aspx file save button click event will be fired.

```
protected void btnSave_ServerClick(object sender, EventArgs e)
{
    try
    {
        var ml = new Garage_ML()
        {
            GarageName = txtName.Text,
            GarageTP = txtTp.Value,
            GarageLocation = txtLocation.Value,
            Email = txtEmail.Value
        };
        var result = new Garage_BL().AddGarage(ml);
        if (result)
        {
            GetGarageData();
        }
    }
    catch (Exception)
```

In that method a new object will be create from Garage model class in the model layer. Then in the object initialize constructor values from the text boxes will be assigned to the models attributes. Then a new object from the Garage business layer class will be created and a method name AddGarage () will be invoked passing the model object we created as the parameter.

```

public bool AddGarage(Garage_ML ml)
{
    try
    {
        var DataDic = new Dictionary<string, object>
        {
            {"@GarageName", ml.GarageName},
            {"@GarageLocation", ml.GarageLocation},
            {"@GarageTP", ml.GarageTP},
            {"@Email", ml.Email}
        };

        return new DBAccessController().InsertRecord(StoredProcedures.sp_AddGarage, DataDic);
    }
    catch (System.Exception)
    {
        throw;
    }
}

```

In the business layer Garage classes AddGarage() methods a new Dictionary type object will be created with passed values through the Garage model object. Dictionary is a visual studio data structure which uses key value pairs. Then a new object will be created from data layers DBAccessController class and a method name Insert Record will be called passing stored procedure and data dictionary as parameters. In here the stored procedures are stored in a class called constants.

```

namespace DataLayer
{
    public sealed class StoredProcedures
    {
        public const string sp_GetSparepartManufacturers = "sp_GetSparepartManufacturers";
        public const string sp_GetSparepartCategories = "sp_GetSparepartCategories";
        public const string sp_GetSpareparts = "sp_GetSpareparts";
        public const string sp_AddCategory = "sp_AddCategory";
        public const string sp_GetCategoryDetails = "sp_GetCategoryDetails";
        public const string sp_RemoveCategory = "sp_RemoveCategory";
        public const string sp_AddEmployee = "sp_AddEmployee";
        public const string sp_UpdateCategory = "sp_UpdateCategory";
        public const string sp_GetEmployeeDetails = "sp_GetEmployeeDetails";
        public const string sp_RemoveEmployee = "sp_RemoveEmployee";
        public const string sp_UpdateEmployee = "sp_UpdateEmployee";
        public const string sp_AddGarage = "sp_AddGarage";
        public const string sp_RemoveGarage = "sp_RemoveGarage";
        public const string sp_UpdateGarage = "sp_UpdateGarage";
        public const string sp_GetGarageDetails = "sp_GetGarageDetails";
        public const string sp_AddManufacturer = "sp_AddManufacturer";
        public const string sp_GetManufacturer = "sp_GetManufacturer";
        public const string sp_RemoveManufacturer = "sp_RemoveManufacturer";
        public const string sp_UpdateManufacturer = "sp_UpdateManufacturer";
        public const string sp_AddTowTruckService = "sp_AddTowTruckService";
        public const string sp_GetTowTruckServices = "sp_GetTowTruckServices";
        public const string sp_RemoveTowTruckService = "sp_RemoveTowTruckService";
        ...
    }
}

```

It is considered as a good coding practice to keep constants in a one separate sealed class. The data layers insert method is shown below.

```
public bool InsertRecord(string storedProcedureName, Dictionary<string, object> dataset)
{
    try
    {
        con = new SqlConnection(connectionString);
        con.Open();
        cmd = new SqlCommand(storedProcedureName, con) { CommandType = CommandType.StoredProcedure };

        foreach (var item in dataset)
        {
            cmd.Parameters.AddWithValue(item.Key, item.Value);
        }
        cmd.ExecuteNonQuery();
        return true;
    }
    catch (System.Exception)
    {
        throw;
    }
    finally
    {
        con.Close();
        con.Dispose();
    }
}
```

In this method it is looped though the data dictionary to get the values and pass them as parameters for the stored procedures. By coding this way it is possible to use this same insert method to carry out any other insert operation of data. Similar approach has being use for other operations for example update and remove also.

```
public bool DeleteRecord(string storedProcedureName, Dictionary<string, object> param)
{
    try
    {
        con = new SqlConnection(connectionString);
        con.Open();
        cmd = new SqlCommand(storedProcedureName, con) { CommandType = CommandType.StoredProcedure };

        foreach (var item in param)
        {
            cmd.Parameters.AddWithValue(item.Key, item.Value);
        }
        cmd.ExecuteNonQuery();
        return true;
    }
    catch (System.Exception)
    {
        throw;
    }
    finally
    {
        con.Close();
        con.Dispose();
    }
}
```

When considering the WCF web service it will also communicate with the database through above discussed communication methodology.

## 6.6 Tablet PC and Smartphone Application Development Architecture

Similar architecture has been used here also namely layered architecture. Since there are very less business logics have been written in tab application only two layers namely view and model layers. Interfaces are developed with XAML and backend coding was done using c#. In the code behind call events for web services is coded.

```
private async void GetGarageinfo()
{
    string location = txtLocation.Text.Trim();
    ServiceReference1.Service1Client client = new ServiceReference1.Service1Client();
    string result = await client.GetGarageInfoAsync(location);
    try
    {
        JArray v = JArray.Parse(result);
        lstvGarages.ItemsSource = this.GetGarageList(v);

    }
    catch (Exception ex)
    {
        txtLocation.Text = ex.ToString();
    }
}
```

## 6.7 Summary

This chapter was dedicated to provide a discussion about development and development techniques as well as the development process of the application by following best practices of the software development. Tools techniques, coding impairments, architectures that used in the development of the applications were also discussed in this chapter.

# Chapter 7

## Testing and Implementation

### 7.1 Introduction

Testing of the System is critical for the delivery of a successful project. The key purpose of testing is to check whether that the developed system is fulfilled the user requirements and confirms to the test results expected under a wide range of conditions which could have been tested. To accomplish a successful system implementation, all aspects of the testing segment should be reviewed with the contribution of the developer as well as users. In this chapter it is illustrated about the types of testing used for developed software product and test plans for used in the process. As the last main topic in this chapter it is discussed about the implementation plan, which includes the factors need to be considered when system is implemented in the real environment.

### 7.2 System Testing Objectives

System testing enables the developed solution to be analyzed entirely in order to ensure that the specifications and business functions which it was intended are being met. The components of the system are being tested at the development of those components. But it is necessary to test those components to confirm the combinations of the system components were properly done.

System testing is conducted by checking the fallowing checkpoints are tailored with the developed system.

- Performance –This testing is performed to verify the system operate under peak and continuous loads of processes at a significant speed.
- Accuracy – Inaccurate information leads the whole system failure if the information is not stable. The test is performed to verify the outputs of the system are accurate in various operational environments.
- Functionality – Test is performed to make sure the system meet requirement specifications and hence supports business requirements of the company. Functionalities should address the problems existed in current system and its processes.
- Interfaces – New solution components are developed to provide more information using less number of interfaces and users should be able identify them easily and separately. Testing is performed to make sure the interfaces are done according to the specifications and the interfaces are linked each other in well-organized manner.

- Security – Since the system handles sensitive information of the organization's and its users security of the system should be well-ensured, this testing is performed to verify the feature of the system ensure access, integrity and recovery features operate as expected.

## **7.3 Testing Strategy**

A proper testing strategy is needed for the finishing of a good product. The developers intend to give the best finished product to the buyers. Testing is done in different methods and such methods are elaborated below. It is important that the testing is done in a step by step basis and that the different types of tests are related to each other.

### **7.3.1 Unit Testing**

Unit testing which also known as module is testing is a kind of procedure used to verify that each unit of source code is functioning correctly. In an application, the smallest testable part is the unit. Units of a system are differentiated from modules and those modules are made up of units. Unit testing will only test the components of the system and it will not help to recognize every fault in that system such as integration errors, performance errors and any system related issues. Unit testing will only be effective when it used along with other testing techniques. Typically this testing is done only by the system developers and not by the users who are going to use the system finally at the real operational environment. This was conducted while developing the solution. Each written piece of code is tested for errors.

### **7.3.2 Integration Testing**

Integration testing which is also known as Integration and Testing is a kind of testing in which the individual modules in the software are combined and tested those as a cluster. Purpose of conducting the integration testing is verifying the functional and performance requirements defined on project design specifications. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing. There can be many types of integration testing such as back bone, bottom-up, top-down and big bang. Integration tests can not include system-wide change testing.

### **7.3.3 System Testing**

Overall functionalities of the system including the modules integrated modules and interfaces will be tested out of the system testing. System testing will not receive many errors since the input which is taken to the system testing is the system and modules which are

tested under unit testing and integration testing. But the errors which identify during this testing process will be critical since it will be affected to the integration of the solutions various modules with each other and leads to changes of the system as a whole.

### 7.3.4 Acceptance Testing

Acceptance testing is a black-box testing which will be the next to be conducted. It is designed to ensure that all the changes made come across with the original system specifications and user requirements of the solution during the design, development and other initial stages. Basically this will decide the how the extend solution has completed the user requirements. Acceptance phase may also act as the final quality gateway, where any quality defects not previously detected may be uncovered

## 7.4 Test Plan

Test plan defines a systematic approach to test a system. This will discuss the process of conducting the above mentioned testing strategies. Under test plan, test strategies which are identified as to be carried, will be separated into to testing plans as unit testing and system testing. Components of the system and integration of components will be carried under unit testing. System testing and acceptance testing will be carried under System testing. Concerns made when designing the test plan are,

- Identifying the components and features to be tested and not tested
- Ensuring all required elements are in place for testing
- Who conducts the testing for particular component or feature
- Plan for make necessary changes for issues arise on testing

Below table illustrate the system test plan for web application

Test Scenario	Description
Test and validate login module for web application	Check whether the login interface is aligned with the user experience guides, results when the credentials are correct and wrong, session control and the logout function from the system.
Test and validate the functions of policy module	Check whether the policy module is loading properly. Check whether it is possible add a new policy for a client filling out the necessary details. Once it is submitted record should be visible in the data grid. Grant the permission to do necessary modifications to certain details of a policy.

Test and validate the claim request handling module	Check whether the requests sent by the clients are received by the systems in a quick manner. And view the request and respond to the request and change the state of the request.
Test and validate the functions of reviewing and approving the claim	Check whether manager is being able to view the claim report submitted by the claim agent and review it and approve it for money transfer.
Test and validate the functions of garage services module.	Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is working providing indicated parameters. Check the ability to add, edit or delete data from/to database.
Test and validate the functions of tow truck services module.	Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is working providing indicated parameters. Check the ability to add, edit or delete data from/to database.
Test and validate the functions of vehicle templates module.	Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is working providing indicated parameters. Check the ability to add, edit or delete data from/to database.
Test and validate the functions of spare parts module.	Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is working providing indicated parameters. Check the ability to add, edit or delete data from/to database.
Test and validate the functions of manufacturers' module.	Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is working providing indicated parameters. Check the ability to add, edit or delete data from/to database.
Test and validate the functions of spare categories module.	Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is

	working providing indicated parameters. Check the ability to add, edit or delete data from/to database.
Test and validate the functions of employees and user accounts modules.	Check whether existing data are retrieved from database and displayed on the data grid. Check whether the searching is working providing indicated parameters. Check the ability to add, edit or delete data from/to database.

Table 4: System Test Plan-Web Application

Source: Author

Below table illustrate the system test plan for tab application

Test Scenario	Description
Test and validate login module for tab application	Check whether the login interface is aligned with the user experience guides, results when the credentials are correct and wrong, and the logout function from the system.
Testing and validating Policy information module.	Test whether the system identifies the customer's policy number and allows the agent to retrieve the relevant data of the relevant module from the company database through the web service.
Test and validate the claim form filling procedure	Make sure that only after giving the policy number agent can go to this screen. All the required fields must be checked before submitting the response. Finally agent must be able to submit the data and get the success message.
Testing and validating Vehicle component information module.	Test whether the system chooses the particular parts list after the part category is selected from the pick list, and whether the right price is given, of the damaged spare part chosen. Test whether the system can add /remove the damaged spare parts to claim and whether it displays on screen. When each item is added, the value and details of the item should be shown in a table.

Test and validate garage services module in tab application	Check whether the data are retrieved through the web service and also check whether filtering works.
Test and validate tow truck services module in tab application	Check whether the data are retrieved through the web service and also check whether filtering works.

Table 5: System Test Plan-Tab Application

Source: Author

Below table illustrate the system test plan for WCF service.

Test Scenario	Description
Test and validate the WCF service hosted by looking in to exposed methods	Check whether all the necessary methods are exposed and check in the WCF test client whether they all response with correct set of data.

Table 6: WCF Service Test Plan

Source: Author

Below table illustrate the system test plan for web application

Test Scenario	Description

Table 7: System Test Plan-Mobile Application

Source: Author

## 7.5 Test Results

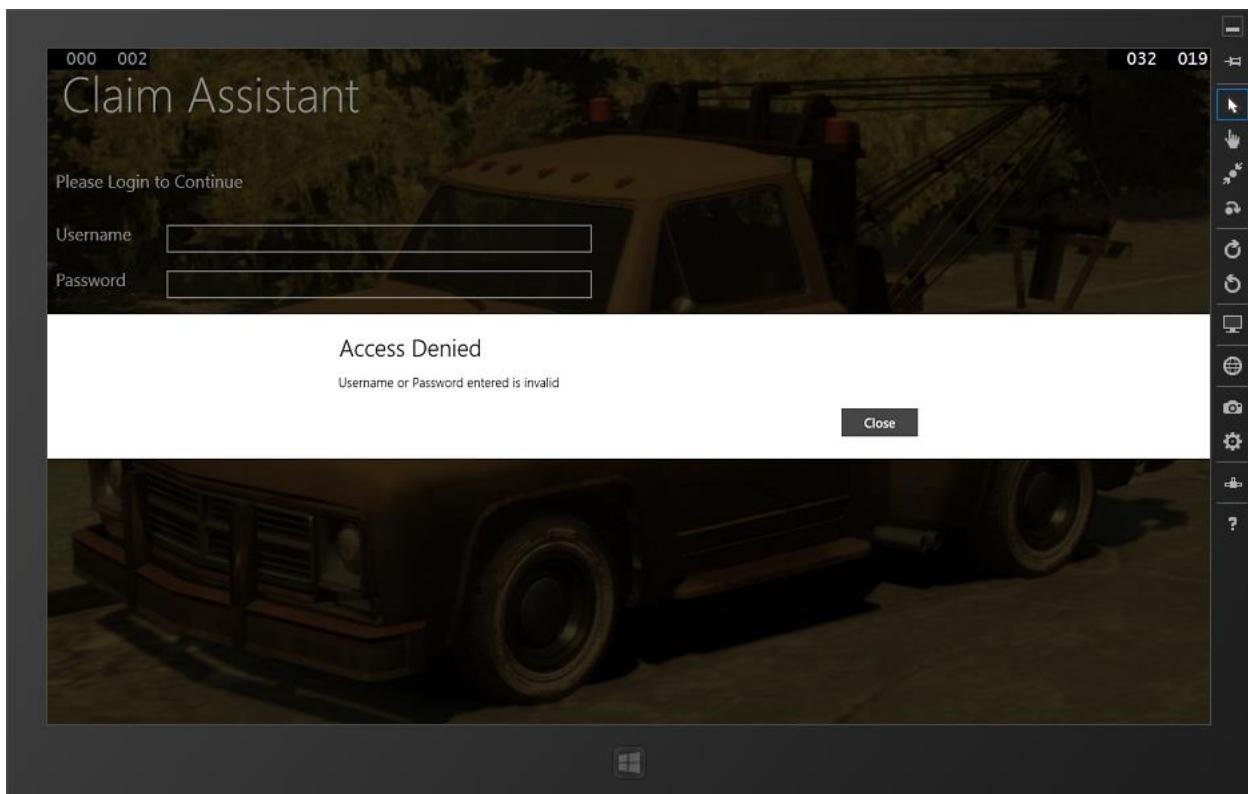
**Test Case:** Testing and validating login function

**Description:** Test the user login for the valid user name and password. The user login should identify the user's privileges, id and the user's name. Only the authenticated users who are registered to use the tab application system should be allowed to login.

**Test Data:**

1. Try to login without credentials

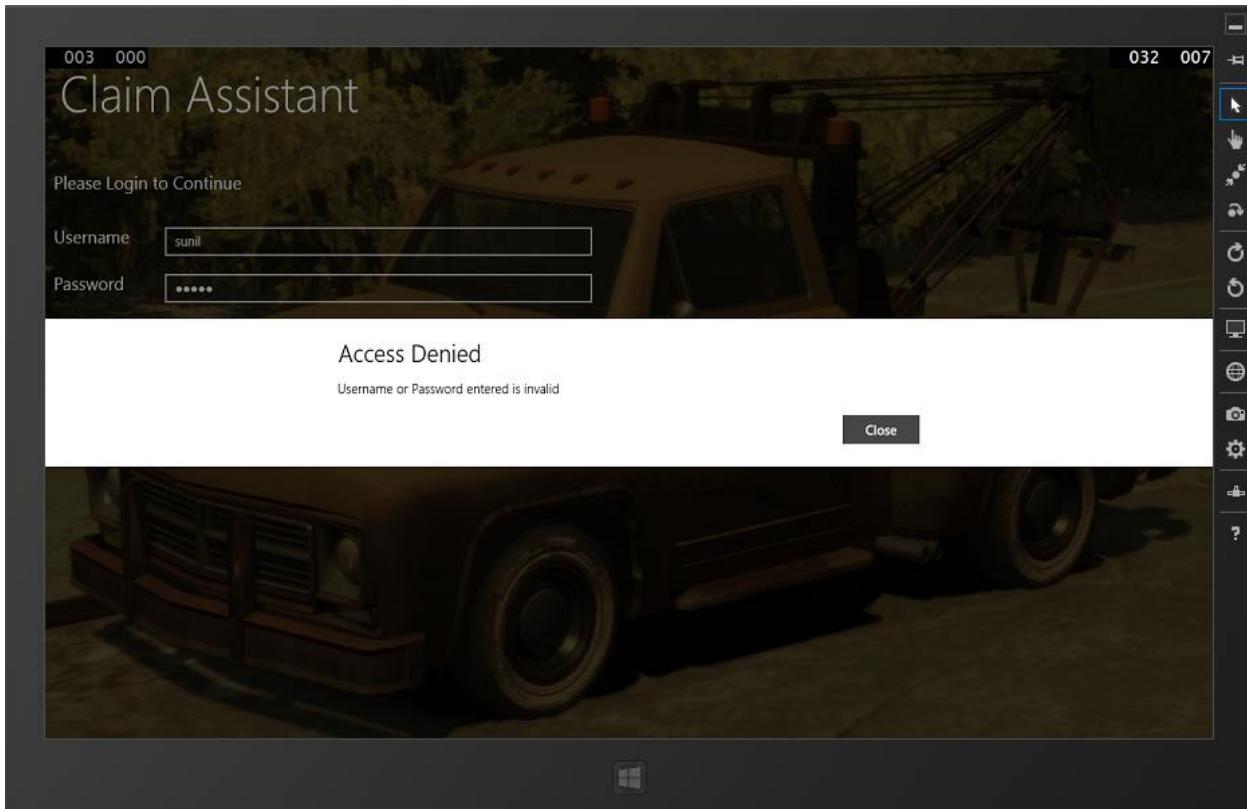
**Expected Output**



**Test Result:** Successfully received the expected message

2. Try to login with invalid credentials

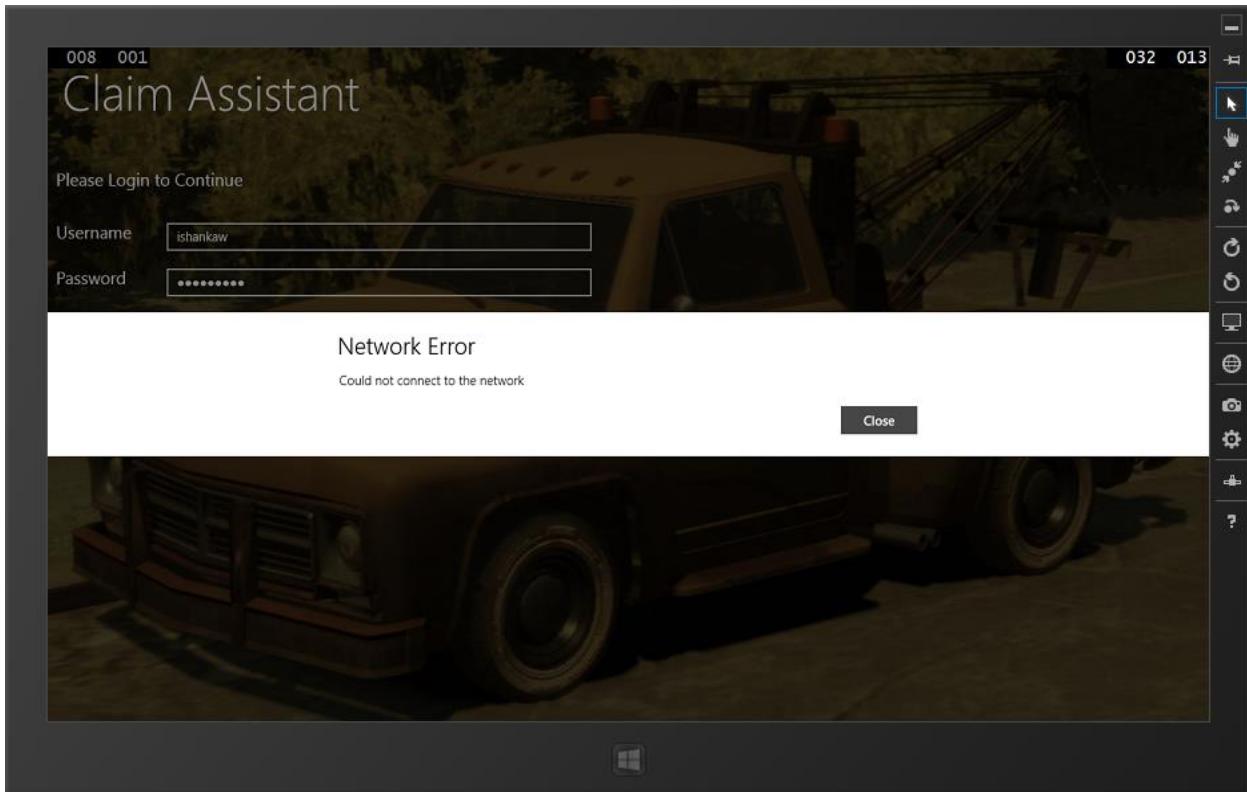
### Expected Output



**Test Result:** Successfully received the expected message

3. Try to login when network unavailable

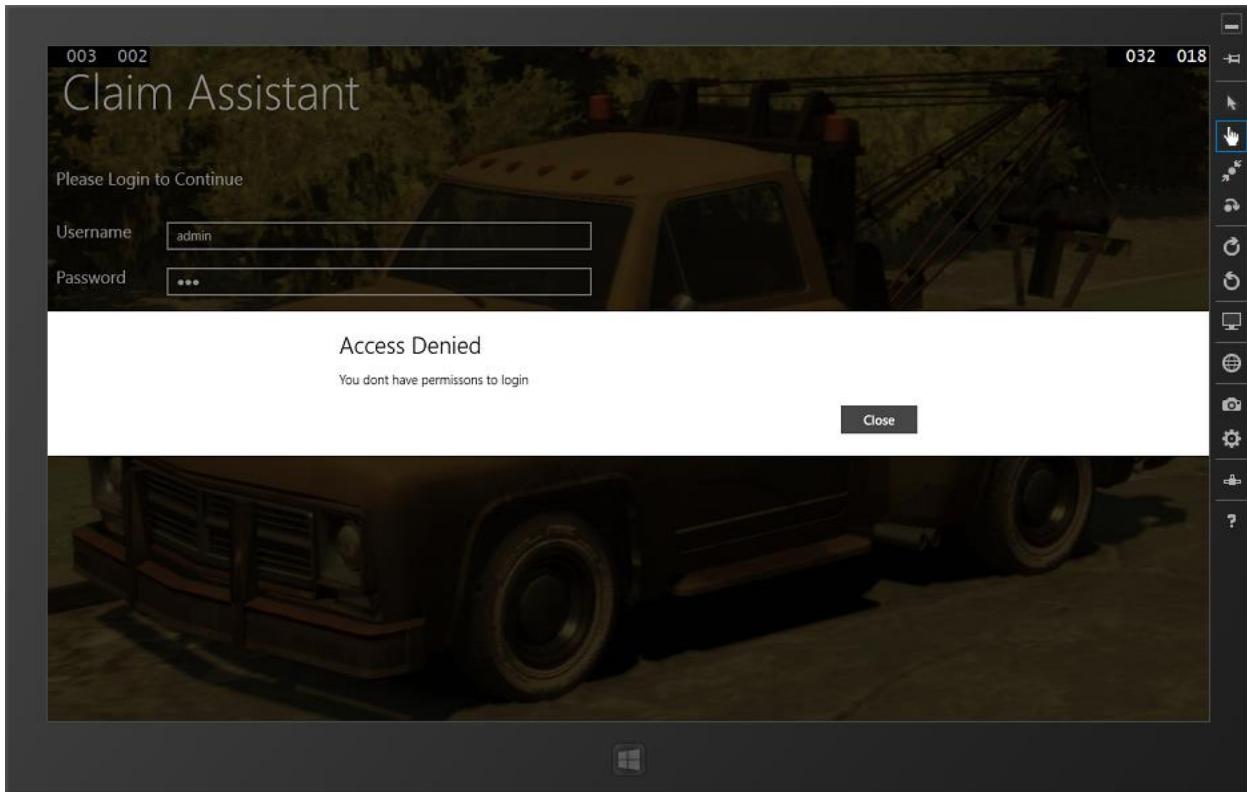
### Expected Output



**Test Result:** Successfully received the expected message

3. Try to login when without enough access.

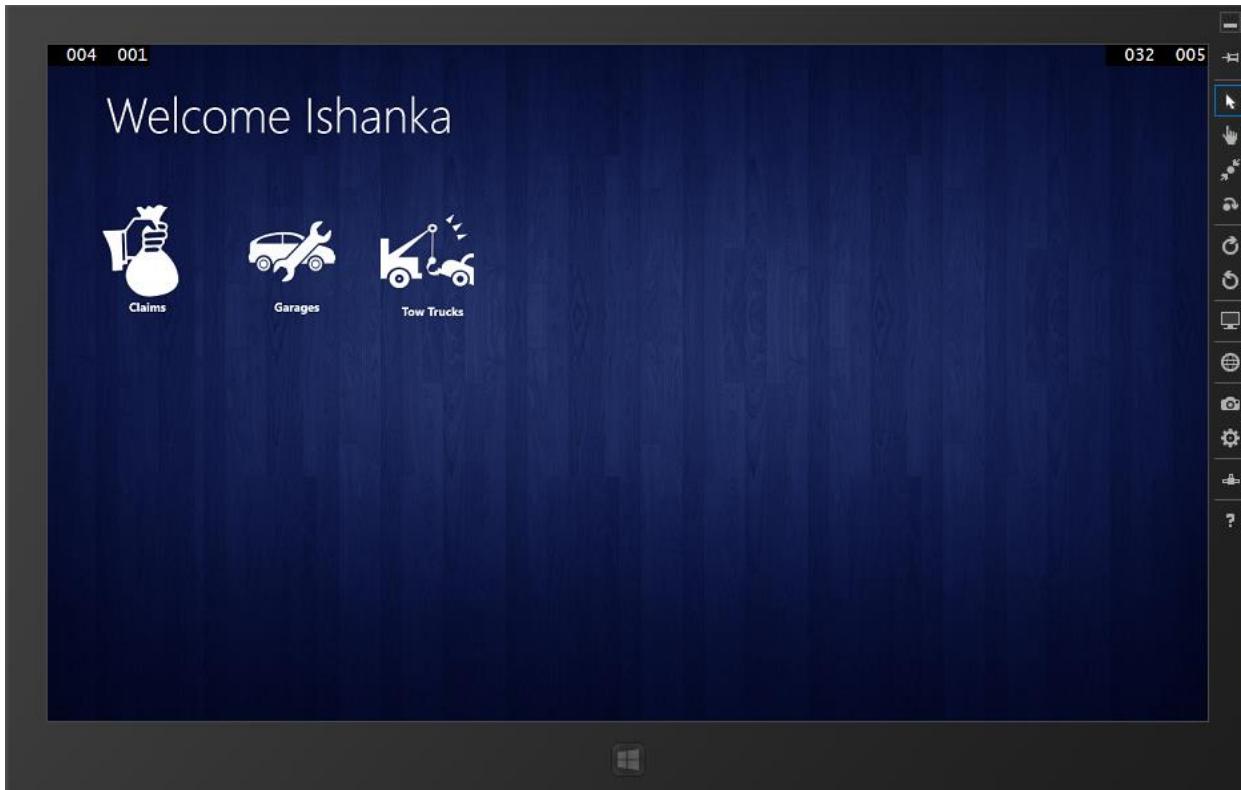
### Expected Output



**Test Result:** Successfully received the expected message

4. Try to login with valid credentials

### Expected Output



**Test Result:** Successfully received the expected output. User logged in successfully. Next screen displayed.

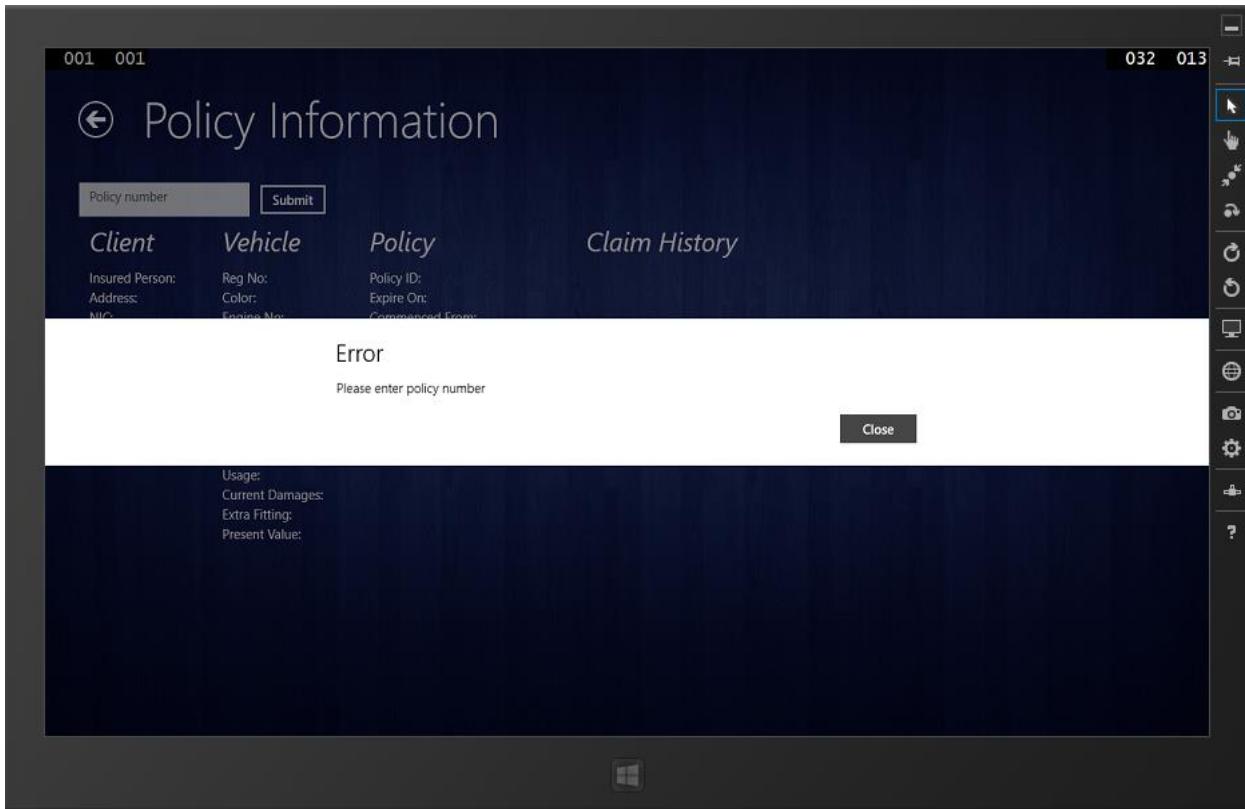
**Test Case:** Testing and validating Policy information module.

**Description:** Test whether the system identifies the customer's policy numbers and allows the agent to retrieve the relevant data of the relevant module from the company database through the web service.

**Test Data:**

1. Submit without policy number

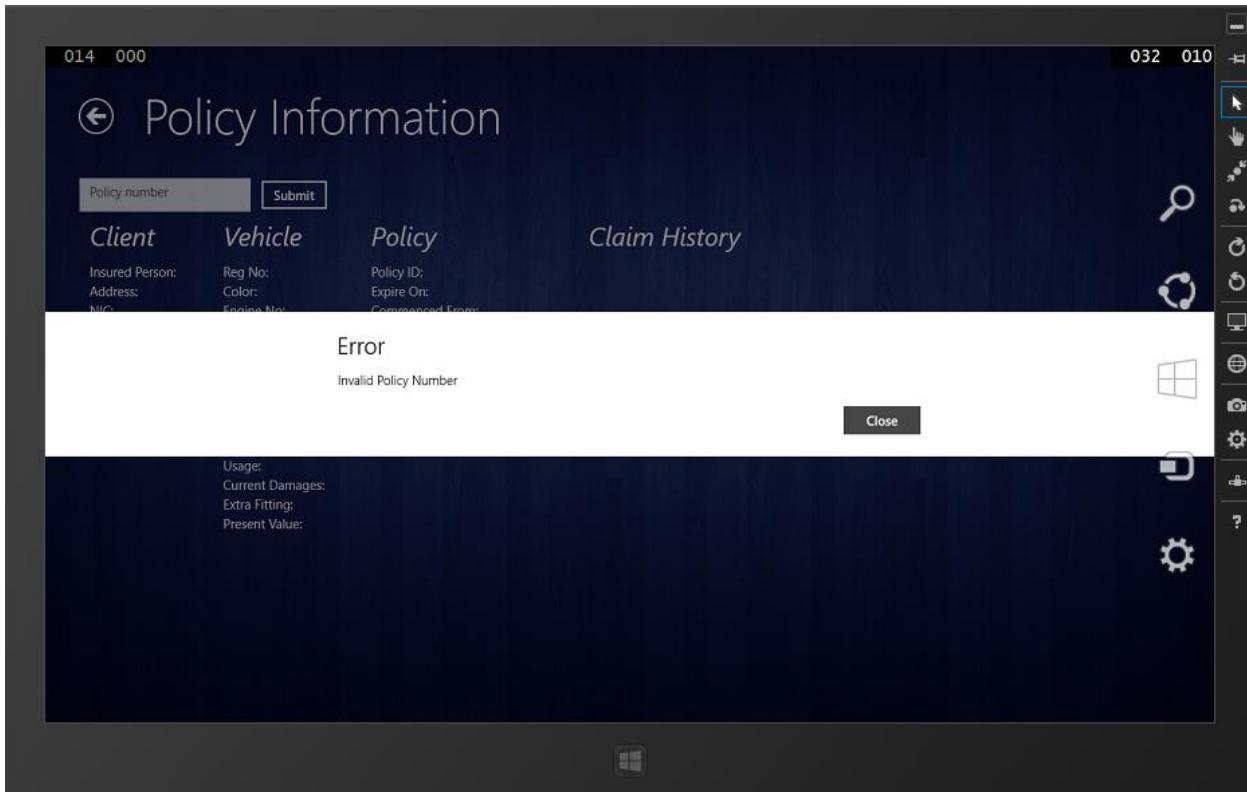
### Expected Output



**Test Result:** Successfully received the expected message

2. Submit with invalid policy number

**Expected Output**



**Test Result:** Successfully received the expected message

3. Submit with valid policy number

### Expected Output

Client	Vehicle	Policy	Claim History
Insured Person: Mr WMIL Wasalage Address: 8b kadhwala Ratmalana NIC: 912701395v Contact No: 0716405220 Email: isankalakshan@gmail.com	Reg No: 253-3402 Color: White Engine No: 548654aa5454 Chassis No: 654987646541ad Manufacturer: Toyota Model: Allion Year: 2012 Engine Capacity: 1400cc Absolute Owner: Loan Financial Rights: Yes Usage: private Current Damages: no Extra Fitting: Spoiler Present Value: 2300000	Policy ID: 4 Expire On: 12/5/2018 12:00:00 AM Commenced From: 12/5/2015 12:00:00 AM Natural Disaster: No Passenger Compensation: Rs. 0 Driver Compensation: Rs. 0 Towing Charges: Rs. 0 Terrorism Cover: Rs. no Vandalism Cover: Rs. Yes	10/31/2015 7:37:32 PM 11/23/2015 1:43:40 AM 11/23/2015 1:48:52 AM

**Test Result:** Successfully received the expected information

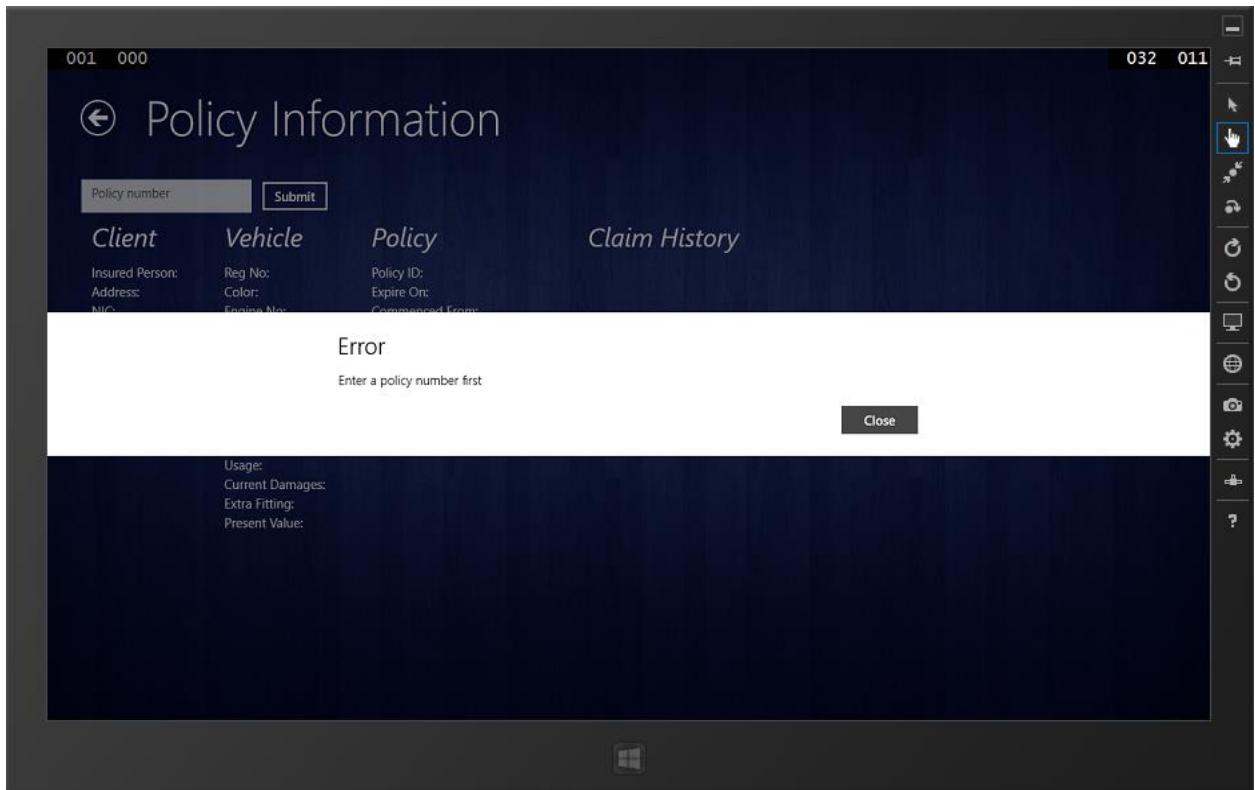
**Test Case:** Testing and validating claim assessing

**Description:** Test whether policy number has been entered before moving to the claim form filling. Then check whether all the required fields are entered. If entered can move to the next screen and finally submit.

**Test Data:**

1. When agent tries to claim without entering clients policy number

**Expected Output**



**Test Result:** Successfully received the expected message

2. When user enter policy number then press claim button.

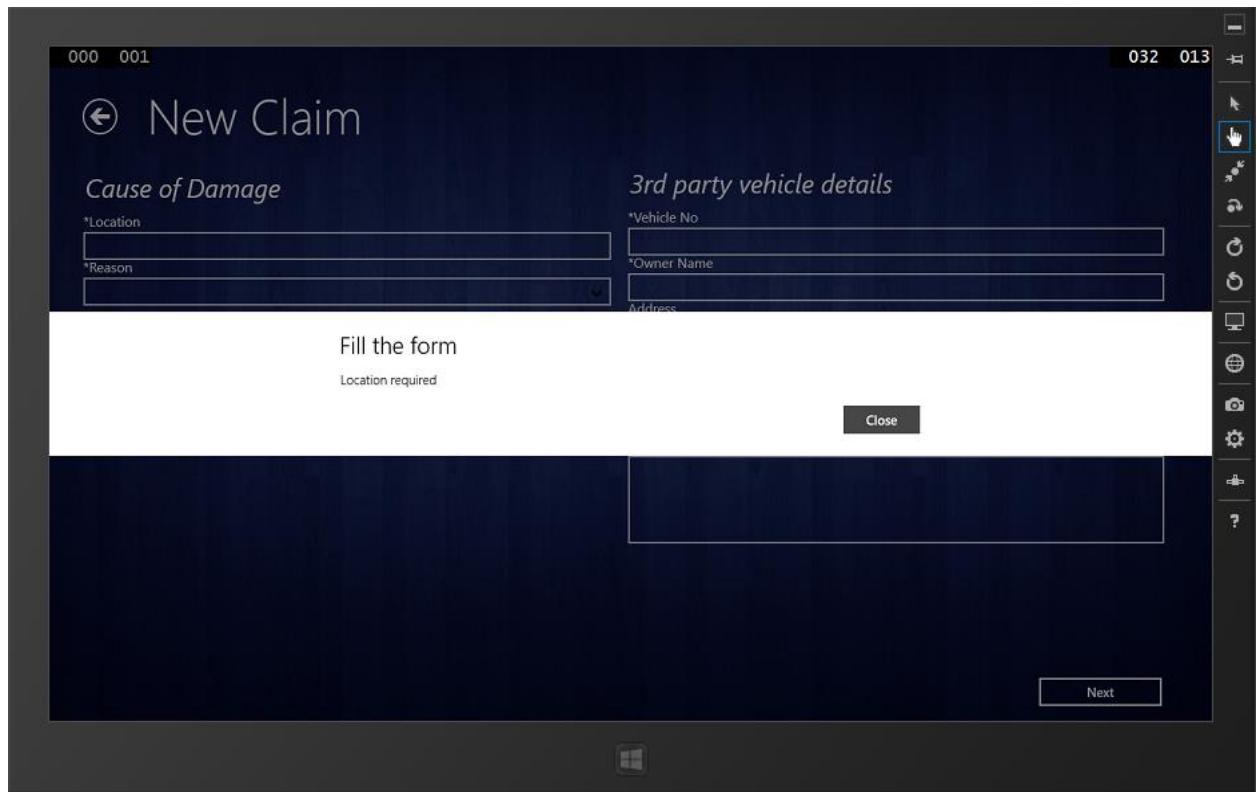
**Expected Output**

The screenshot shows a software application window titled "006 002" at the top left and "032 016" at the top right. On the far right, there is a vertical toolbar with various icons. The main interface is titled "New Claim" with a back arrow icon. It is divided into two main sections: "Cause of Damage" on the left and "3rd party vehicle details" on the right. The "Cause of Damage" section contains fields for "Location" (dropdown), "Reason" (dropdown), "Knocked On" (dropdown), and "Custom" (dropdown). The "3rd party vehicle details" section contains fields for "Vehicle No" (dropdown), "Owner Name" (dropdown), "Address" (dropdown), "Contact No" (dropdown), and "Renewal Date" (dropdown set to November 28, 2015). There is also a "Special Notes" text area and a "Next" button at the bottom right.

**Test Result:** Successfully received the expected message

3. Checking for required filed in claim form

**Expected Output**



**Test Result:** Successfully received the expected message

4. On successful completion of filling first form and press Next button

**Expected Output**

The screenshot shows a Windows application window titled "New Claim". The window has a dark blue header bar with "000 003" on the left and "032 005" on the right. On the far right, there is a vertical toolbar with various icons. The main content area is divided into two main sections: "Driver Details" on the left and "3rd Party Damages" on the right.

**Driver Details:**

- Is the driver owner?  Yes  No
- Driver Name: [Text Box]
- License: [Text Box]
- Categories: [Text Box]
- Expire Date: [Select Box] November 28 2015
- NIC: [Text Box]
- Date of Purchase: [Select Box] November 28 2015
- Vehicle used for?: [Text Box]

**3rd Party Damages:**

- Victim Name: [Text Box]
- Address: [Text Box]
- Damage Nature: [Text Box]
- Third party Claimant: [Text Box]
- Amount Claimed Rs.: [Text Box]

A large circular arrow icon is positioned in the center of the "Driver Details" section. At the bottom right of the main content area is a "Next" button.

**Test Result:** Successfully received the expected message

5. On successful completion of filling second form and press Next button

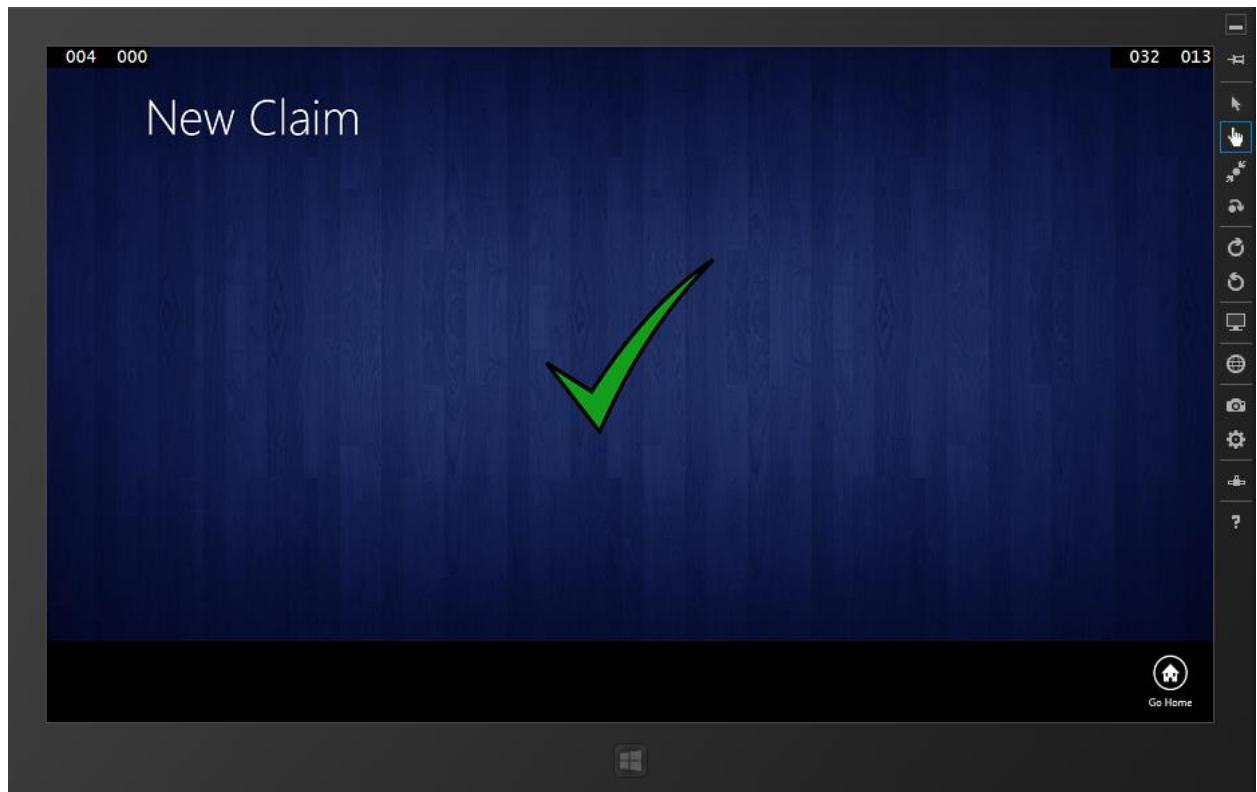
**Expected Output**

The screenshot shows a software application window titled "001 001" at the top left and "032 015" at the top right. The main title is "New Claim" with a back arrow icon. On the left, there is a section labeled "Attach Photos" with a "Choose Image" button and a circular "+" button. On the right, there is a "Payments" section. It includes fields for "Sparepart Category" (set to "Engine"), "Manufacturer" (empty), "Sparepart" (set to "Radiator"), "Year" (set to "2011"), "Quantity" (set to "1"), and buttons for "Remove" and "Add". Below these, a message says "Radiator- Rs 35000/= Qty: 1". Further down, there are fields for "Spare parts total cost" (set to "34 000"), "Garage Costs" (empty), "Any Other Costs" (empty), and "Amount Payable" (set to "30 000"). The background of the application has a dark blue gradient. A vertical toolbar on the right side contains icons for various functions like zoom, crop, and help.

**Test Result:** Successfully received the expected message

6. On successful submit of claim

**Expected Output**



**Test Result:** Successfully received the expected message

**Test Case:** Testing and validating vehicle spare part information module.

**Description:** Test whether application retrieves spare part information from the company database and the system can add /remove the damaged spare part/parts and weather it displays on screen. When each item is added, the value and details of the item should be shown in a table.

**Test data:**

1. Check whether data are retrieved

**Expected Output**

The screenshot shows a Windows application window titled "New Claim". On the left, there is a section labeled "Attach Photos" with a "Choose Image" button. On the right, there is a "Payments" section containing a table of spare parts. The table has columns for "Sparepart Category", "Year", and "Manufacturer". The data in the table is as follows:

Sparepart Category	Year	Manufacturer
Windscreen	2015	Rs 13000 /=
Radiator	2011	Rs 35000 /=
Caps	2010	Rs 2010 /=

Below the table, there are input fields for "Spare parts total cost" (34 000), "Garage Costs", "Any Other Costs", and "Amount Payable" (30 000). At the bottom right is a "Submit" button.

**Test Result:** Successfully received the expected output.

2. Check whether an item can be added to the claim

**Expected Output**

The screenshot shows a software application window titled "001 001" at the top left and "032 015" at the top right. On the left, there is a section labeled "Attach Photos" with a "Choose Image" button and a circular "+" button. To the right, under the heading "Payments", there are several input fields:

- "Sparepart Category": Engine
- "Manufacturer": (empty)
- "Sparepart": Radiator
- "Year": 2011
- "Quantity": 1
- "Remove" and "Add" buttons

Below these, a message states "Radiator- Rs 35000/= Qty: 1". Further down, there are more fields:

- "Spare parts total cost": 34 000
- "Garage Costs": (empty)
- "Any Other Costs": (empty)
- "Amount Payable": 30 000

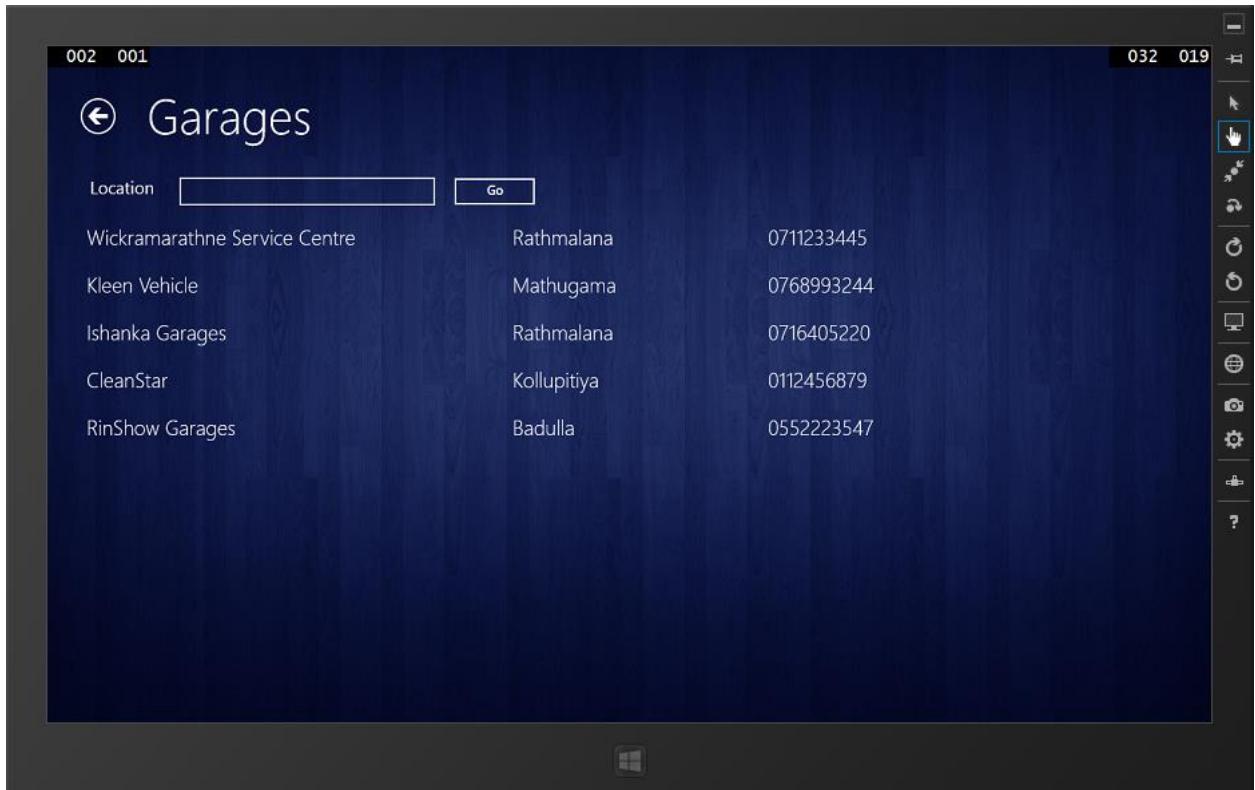
**Test Result:** Successfully received the expected output.

**Test Case:** Testing and validating vehicle spare part information module.

**Description:** Test whether application retrieves garage and tow truck services information from the company database. And also checking whether searching items by location function properly.

1. Get all garages services information

**Expected Output**



**Test Result:** Successfully received the expected output.

2. Get all tow truck services information

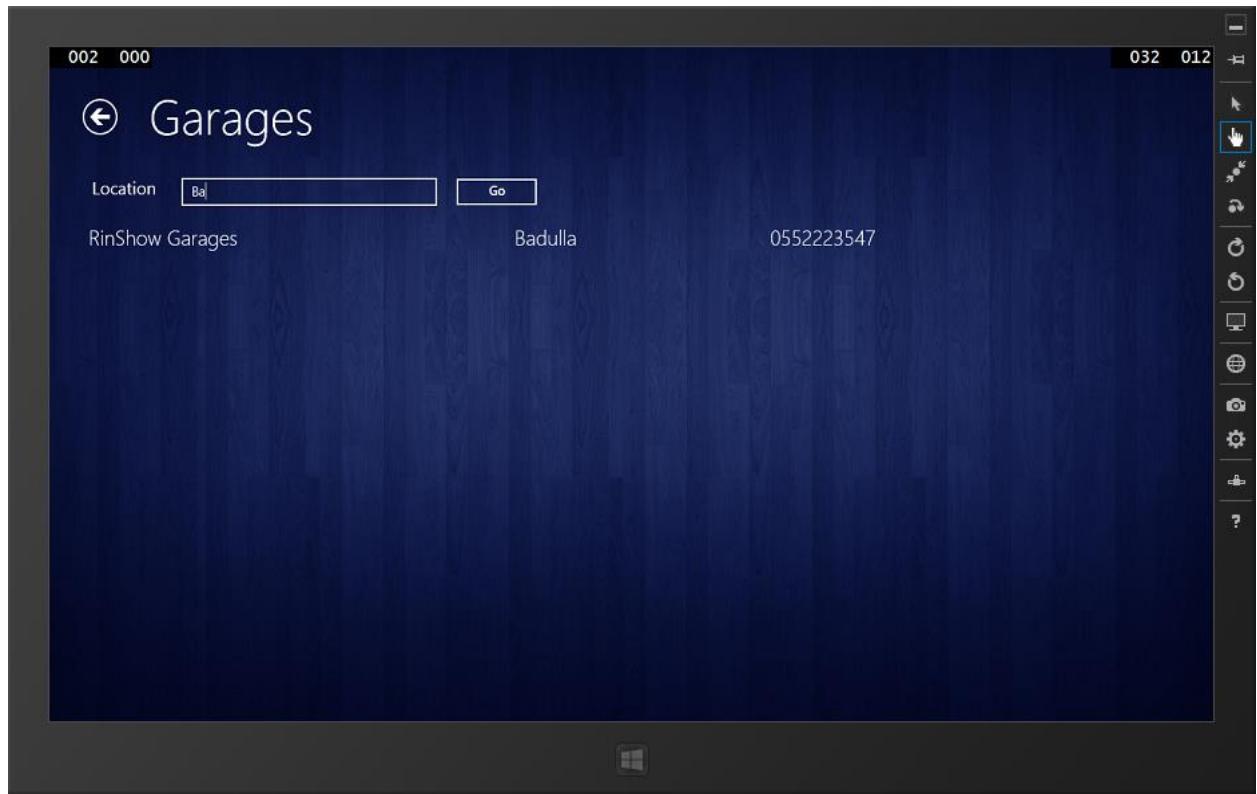
**Expected Output**



**Test Result:** Successfully received the expected output.

3. Search by location in garages services

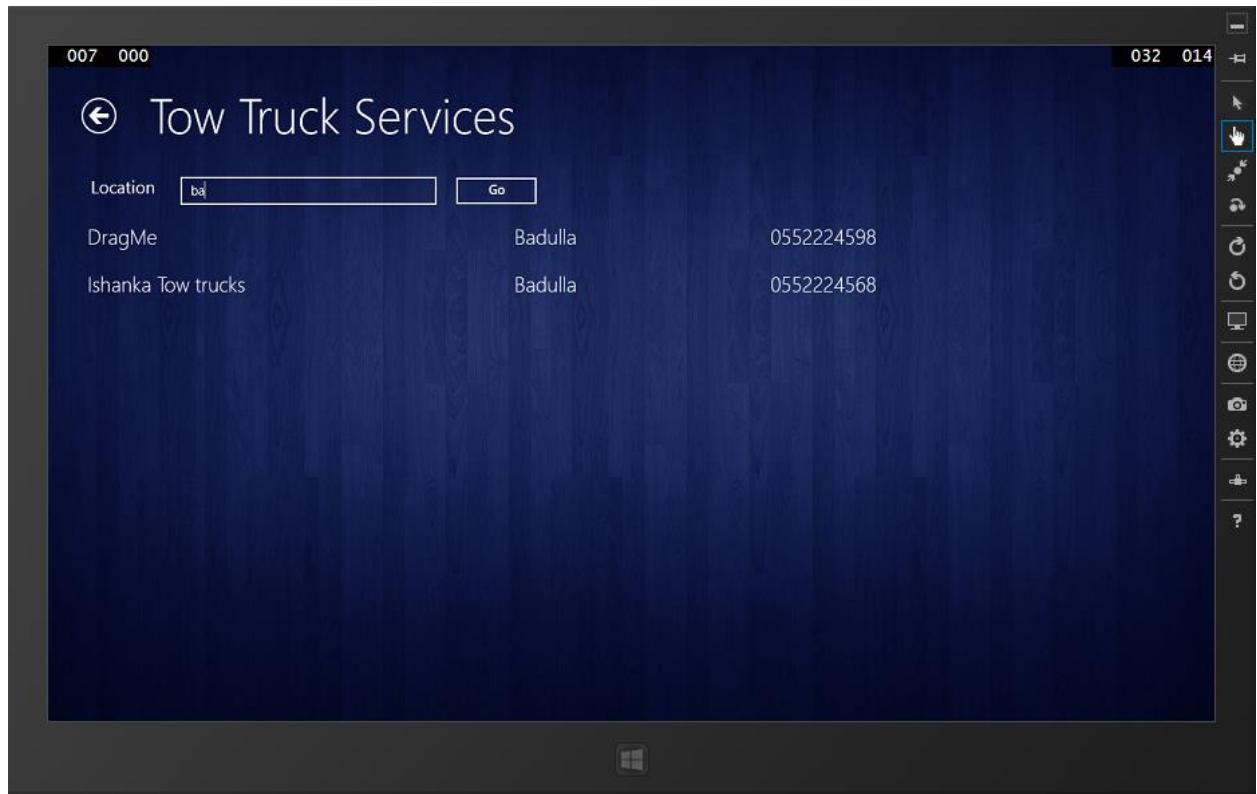
**Expected Output**



**Test Result:** Successfully received the expected output.

4. Search by location in truck services

**Expected Output**



**Test Result:** Successfully received the expected output.

**Test Case:** Testing and validating smartphone application.

**Description:** Test and validate claim request reporting module.

1. Testing when the client submits without policy number

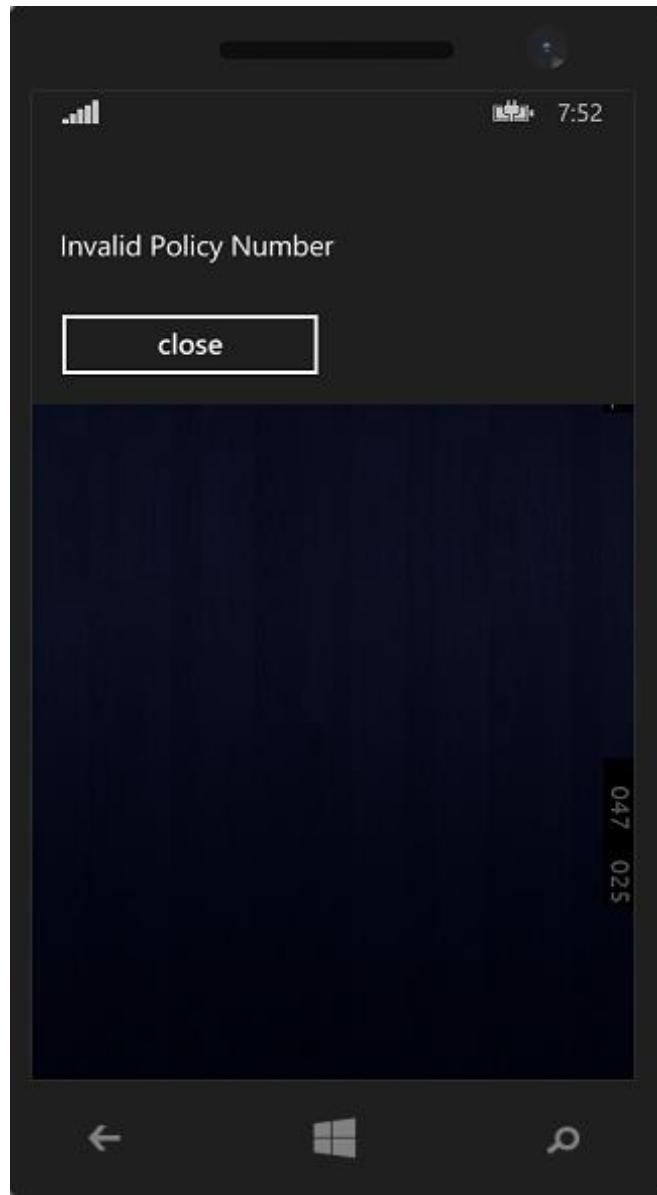
**Expected Output**



**Test Result:** Successfully received the expected output.

2. Testing when the client submits with invalid policy number

**Expected Output**



**Test Result:** Successfully received the expected output.

3. Testing when the client submits with valid policy number

**Expected Output**



**Test Result:** Successfully received the expected output.

Test cases of the web application are attached in the **Appendix D**

## **7.6 Implementation Plan**

The importance of having a good implementation plan is that the prepared system can be introduced to the client step by step. It would minimize the wastage of time and also money. A proper plan should be adhered by both the developers and the insurance company. The main activities of the implementation plan is to give a system overview, key activities on system implementation and any other vital requirements that helps to the process of implementation.

## **7.7 System overview**

Implementation of the TBMVCS would idyllically observe the departure of the old procedure of performing the claiming process manually. Therefore it is necessary to provide a smooth transition, when changing from manual to the computerized system. And as a result of this transition, users may find difficult to cope up with the new functions of the new system. So it is very essential to provide support for the user in order to handle the new functionalities of the new system.

The TBMVCS would make the Motor vehicle insurance claiming more transparent and efficient both for the employers of the company the clients who pay monthly insurance premiums. Not only this but the claiming agents would be motivated a lot since their work is cut down in half and they are able to cover more vehicle claims during a day. This would result in quicker response, better payment and more concern for customer satisfaction. The basis is to switch to smart devices technology from the use of manual handwritten filing methods.

## **7.8 Major Tasks on System Implementation**

- Planning overall implementation
- Co-ordination of the system implementation with the management of the company
- Testing the Tab application and Mobile application along with the data connections
- Identifying resource requirements for the implementation
- Finding a source to get market prices for vehicle spare parts
- Providing training for the claiming agents
- Planning data conversion methods before implementing the system
- Planning a risk management system
- Ensure all the required hardware and software is available

## **7.9 Installation**

According to the implementation plan the first activity that should be done by us is to install the system at the real operational environment. Implementation of the TBMVCS should be done with the help and the guidance of the management of the Insurance Company. Since the solution is developed commonly for insurance companies we may need to do small changes to the user interface such as adding their logo. Successfully developed and tested TBMVCS should be set up in order to reduce the perils and contingency which expressed in the above section. Procedure of the system installation is given below.

- If hosted in a separate server the operating system on server (SQL server 2012 above preferred) should be installed first. Another easier option will be hosting on Azure which is Microsoft cloud platform
- Microsoft SQL should be installed in order to support the database connectivity. Azure comes with SQL server as well.
- Connect the code repository to the website created enabling continuous integration for updating the system
- After that we have to restore the database
- Make essential configurations for the BMVCS
- Install the Tab application on company devices.

## **7.10 System Changeover**

The direct changeover method will not be followed but parallel changeover will be carried on TBMVCS system because the company would not be able to take the change so soon. The reason is that there are so many existing customers and claiming staff members who are used to the old system. Also since the management would want to find the bugs in the process of the implementation, the parallel changeover would be the better deal. This will reduce the risk of business and give the new users time to adjust to the new system. Since the 3 component are dependent on each other all components are implemented in the business process. Mobile application can be directly used since it does not have a direct impact on the company business process but a value addition.

## **7.11 Data Conversion**

The practical solution is to start the data conversion after the implementation of the system. The current claim data can be taken from the Insurance Company's database. Some of the data has to be entered fresh since the TBMVCS system intends the new features. Also the parallel method used can take the data entering from fresh new customers of the insurance company. Data conversion for the online assessment system should be entered manually and will take some time since it has to find the prices of the vehicle spare parts. It is appropriate

to mention that some of the data would be fed on a time ticking basis since the prices of vehicle spare parts could appreciate and depreciate in the market.

## **7.12 User Training**

Most of the present claiming staff are quite proficient with computers and other smart devices. In the modern world most people are familiar with the use of smart mobile phones. Since the TBMVCS system, tab application and smart phone application has a very simple user friendly interface they need not be trained much on this. Since there is no big difference in the functionality of the system and the difference is more of a convenience the motivation of the staff members would be at a high. It is necessary to inform the entire user about the functionality of the system during this implementation. There would be no need for new rules and regulations except for the safety of the devices when travelling with them. The data operators at the Head Office would have to be a very minimal staff and could also act as the monitors of the insurance claims being done around the island.

## **7.13 Summary**

The system testing that occur along with the system development and the activities that occur before the delivery of the product called implementation were expressed. In agile approach testing is concurrently conducted with the development and the testing has done in several test runs to determine all the defects. Manual testing methodologies were used since the time was not enough for dig deep into adopt to test automation methodologies. Implementation plan has the sequence order of set of activities such as installation, data conversion, and change over and user training.

# Chapter 8

## Evaluation

### 8.1 Introduction

Main reason for conducting an evaluation phase is to recognize the extent which the developed solution fulfills the requirements of the real users. And along with that to measure the usability aspects of the system. Therefore the chapter has attempted to emphasize on key areas of the system evaluation and also the evaluation of system practices

### 8.2 Types of Evaluation

Evaluation is a systematic process of acquiring and assessing the system related information in order to provide useful feedback about system. Questions such as “whether the current system process is in the right track and meeting the users' requirement?” or “whether the developed system has met the user requirements?” are answered by conducting system evaluation at different stages of the development. Formative Evaluation and Summative Evaluation are two types of evaluations performed during the system development.

Formative evaluations strengthen or improve the system being evaluated. They help form it by examining the delivery of the program or technology, the quality of its implementation, and the assessment of the organizational context, personnel, procedures, inputs, and so on. Summative evaluations, in contrast, examine the effects or outcomes of some object. This is done on finished system in order to assess the success of it. High-fidelity prototype method which uses the materials that are similar to the finished product, is used to perform the summative evaluation in order to measure the success of the system.

### 8.3 Evaluation of Project Outcome

The key purpose of the product evaluation is to check whether the system is operating as expected and has achieved the objectives. Outcomes of the product evaluation helps to determine whether the functional requirements or the user requirements of the new system had been executed during the stage of development. The functional evaluation is used to evaluate the outcomes of the project. Functionality tests are carried out by using several groups such as user group and evaluators in order to make sure whether the system had continuously attempted to meet the functionalities and to find out any faults that could be occurred during the functionality of the new system. Results of some evaluations tasks done are presented below. In each scenarios application were installed on the devices before evaluating and the user accounts were provided when necessary.

Participants	1	2	3	4	5
<b>Demographic Details</b>					
User Type		Claim Agents			
Sex	Male	Male	Male	Male	Male
Experience with smart devices	Mid	Mid	High	Low	Mid
<b>Tab Application Tasks</b>					
Login to the application	✓	✓	✓	✓	✓
Enable agent to claim	✓	✓	✓	✓	✓
Enter Policy Id and retrieve policy information	✓	✓	✓	✓	✓
Enable agent to press the claim button	✓	✓	✓	✓	✓
Enable agent to fill the necessary forms	✓	✓	✓	✓	✓
Enable agent to add/remove spare parts from list	✓	✓	✓	✓	✓
Enable agent to attach images	✓	✓	✓	✓	✓
Enable agent to complete the form and submit the claim to ground office for approval	✓	✓	✓	✓	✓
<b>Evaluation Criteria's</b>					
Time taken to submit claim	12m	11m	10m	15m	12m
Number of faults made	0	1	0	2	1
Number of times user expressed satisfaction	2	0	1	2	1
Number of times where the user got frustrated	0	1	0	1	0
Number of times users needed help	1	0	0	1	0

Table 8: Evaluation of Tab Application

Source: Author

Participants	1	2	3	4	5
<b>Demographic Details</b>					
User Type		Clients			
Sex	Male	Male	Female	Male	Female
Experience with smart devices	High	High	Mid	Low	Mid
<b>Mobile Application</b>					
Open the application	✓	✓	✓	✓	✓
Enable to type the policy number and report	✓	✓	✓	✓	✓
<b>Evaluation Criteria's</b>					
Time taken to submit claim	2m	3m	2m	3m	3m
Number of faults made	0	0	0	1	0
Number of times user expressed satisfaction	1	1	0	2	1
Number of times where the user got frustrated	0	0	0	0	0
Number of times users needed help	0	0	0	0	0

Table 9: Evaluation of Mobile Application

Source: Author

In order to make the evaluation process fruitful, it is vital to find the overall aspects of the system which could be improved further. Analyzing the design findings against the usability principle will be a very good approach to this. From evaluating the screen designs of the MVCS according to data obtained by interviewing and observing.

Component	Tab Application	Web Application	Mobile Application
Usability Principle			
Keep it simple & clear	√	√	√
Visibility of system status	√	√	√
Match between system and the real world	√	√	√
User control and freedom	√	√	√
Familiarity	√	√	√
Help users recognize, diagnose and recover from errors	√	x	√
Error prevention	x	√	√
Recognition rather than recall	√	√	√
Flexibility and efficiency	x	√	√
Aesthetic and modern UI design	√	√	√
Help users recognize, diagnose, and recover from errors	√	√	√

Table 10: User Interface Evaluation

Source: Author

## 8.4 Significance of the Solution

As an overall by using the tablet application agent were able to access and confirm the policy details real time in a speedy manner. Another functional requirement was to provide agent with the real updated value of the vehicle parts avoiding the estimating the market value of spare parts according to the agents experience. It was also successfully achieved through allowing the agent to access the exact market price by accessing to the company's database via the web service. Agent can be easily accessed to the database and get the details of added vehicle component of a particular vehicle. All the documents that the agent needs to carry out are replaced by a small portable tablet application which includes connects with the company database. Photos which are used to perform the claim process accurately can take instantly via that tablet application and can be uploaded to the companies' database at that instance.

Solutions to the Problem Domain	
Problem to Address	Solution Provided
Agent needs to contact the head office in order to verify the particular customers' policy number and to get the relevant details under that policy number. This requires various resources like communication tools and time.	Agent can do this by entering the policy number to the tablet application. Solution to the mentioned problem was provided.
Agent will only name the current market value for the damaged vehicle part by his market experience and that value may not be the real updated value for that particular vehicle spare part.	Agent can gain the exact market price by accessing to the company's database via the web service.
Special components (such as high tech car setups, defending buffers) added to the vehicle cannot be easily assessed due to the lack of information (since the agent has no information regarding the insurance of those components and verification provided by the insurance company).	Agent can be easily accessed to the database and get the details of added vehicle component of a particular vehicle.
An agent needs to carry out and review lots of documents in order perform an accurate claiming process	All the documents can be replaced by a small portable yet powerful tablet pc which includes the access to company database
Agent needs several instruments such as cameras in order to perform an accurate claiming process. And also take considerable amount of time to approve the claiming amount by the claiming officer at the head office.	All devices and utilities needed are integrated in to single device and can upload images instantly.
Client have to call and specifically tell the location of the accident which may be sometimes not clear to the agent.	Client can easily report the accident with a one click along with the GPS coordinates

Table 11: Solutions to Problems Summary

Source: Author

## **8.5 Evaluation of Project Practices**

Evaluation of project practices is mainly performed to assess the process and actions in order to evaluate how it was done, to assess the level of success, find out any alternative ways of doing process & actions and to figure out any limitations that were arise at the each stage of the software development.

Since the agile approach was used in developing the system, requirement gathering, designing, development, testing were conducted con currently. Several meetings were conducted with the clients to get the requirements and then a working prototype was developed. By demonstrating the prototype to the clients feedback was obtained for the prototype as well as the additional requirements. By adapting to this method developers were able to identify the erroneous places in the requirements at the early stage in module wise. Not only that but also client feedback about the design gathered in order to provide them with more user friendly interfaces. Less complex UML techniques were also used in designing the system during development process. They were also demonstrated to customer to get the feedback. By doing so developer was able to fine tune the requirements to exactly what the client wants and it led to saving lot of development time.

Once the development started, the system was tested with the use of different types of system strategies such as unit testing, integration testing and system testing. Testing was done concurrently with the development of the system before conducting the meeting with clients. Several test cases were designed in order make sure system behaves as the client requested. After identifying the necessary changes to the system prototype was again fine-tuned with the changes before the next meeting. This whole process was conducted iteratively until clients were satisfies with the developed system. In general, the customer feedback meetings was successfully accomplished though smaller amount of shortcomings were reported. The shortcomings that were found out were mainly due to the short of knowledge on insurance industry, motor vehicle market and also for not having a similar system in the industry or any other industry at the present in Sri Lanka.

## **8.6 Summary**

This chapter was dedicated to provide a detailed evaluation on the project practices and the product developed to the reader. Process evaluation was done by evaluating phase by phase while product was evaluated discussing the problems of the current system and functional requirements of the new system.

# Chapter 9

## Conclusion

### 9.1 Introduction

In this final chapter of the project report, an overall summary of the development of the system is discussed. In addition to that several ideas for future enhancement of the system are suggested in order to make the system more effective. Learnings and experiences gathered by completing this development project are discussed at last in the author's note.

### 9.2 System Summary

Tablet PC based Mobile Vehicle Claiming System is a solution for automating the insurance claiming process through a smart tablet pc. The overall system was developed in general for insurance companies without targeting a specific company. The current process is conducted manually by writing the claim information on a document. Later it is entered on the database at the Head Office. The longstanding system had a few drawbacks such as claiming officers had to take guesses of the prices of the vehicle components most of the time and the system was open to foul play. A complete automation of the system and instant communication online was the solution provided by the TBMVCS solution.

The insurance claiming agent visits the accident location and login to the tab application in the solution. It will let him the full real time access to the company database proving access to all the details such as about policy, customer, vehicle, spare parts and so on. Then he can fill out the form from the tab application and instantly submit it to the database with attaching images as well. The claiming agent need not have to do guessing work or calculations behind the customer. The system is done so that the customer would not get a low evaluation for the insurance claim and the company wouldn't lose much due to over evaluation. The agent is allowed to work more independently and without the hassle of carrying documents when traveling in motor bikes.

The solution definitely profits the company because it reduces time and the cost involved in over evaluating claiming amounts. It also gives an advantage in monitoring the activities and whereabouts of their agents who are scattered across the country. This relieves the burden on the management and the management could be centralized in the head office

## **9.3 Future Enhancement**

It gave me a great pleasure, because as a developer I was successful enough to meet the requirements of the users and I was also able to develop a successful system prototype which was accepted by majority of users. Though it was able to fulfill the current requirements, it doesn't mean that we are not in a position to improve the solution further. With all the undergraduate academic works and industrial training as the product owner and the developer I needed to keep the scope to a manageable level because of time constraints and cost constraints mainly thus limiting the system functionalities. But with the improvements in the technology there are so many enhancements which can be integrated to the solution which few of them are explained below.

### **Introduce easy claim pay system**

A module can be integrated to the solution for sending the claim amount to the customers' bank account through the system automatically once it is approved. In the current developed version all the claim is submitted through the system money still have to be manually handed over to the client. As a future enhancement that step also can be automated.

### **Introduce navigational facilities**

When an accident happens although the client can report it along with the location through the smartphone app it will only visible on the map for ground office staff who will handle the incoming alerts. As an upgrade it can be implemented such that requests will be [processed and sent it to the claim agent who is nearest to the accident location automatically. Then in the tab application the request will be processed and provide GPS navigation to reach the accident location.

### **Integrating the system with current HR systems**

In the current solution only the basic information are stored about the employees. By integrating this with the company's current HR system, company can evaluate the activities done by claim agents separately and take them into consideration when offering year end promotions and bonuses.

### **Introduce the API for vehicle spare part prices**

Vehicle spare part prices will change constantly. It will be much more easier if a web API can be developed and integrate it with the new solution to automatically update all the spare part prices regularly which will save time.

## **9.4 Authors Conclusion**

From the initial discussions with some insurance companies I knew that the system was going to be very complicated due to the volume of data that would be involved in the process of developing the software. It will be more complex when developing 3 kinds of applications which are different to each other. It was a challenge for me to learn all the development platforms which kept me motivated to produce a successful solution while learning lot of things. It made the project highly interesting and I believe I ended up having a better product than I intended to have. In developing the software had a very precise concentration on using object oriented concepts to the core. The system was very successful at the implementation and a lot of encouragement was given to me by the Insurance Company as well as my supervisor.

I was able to collect vast amount of knowledge and experiences on various matter throughout this project. Since this my last project of my degree I wanted to use all my knowledge that I gained from different subject areas. This project helped me to put all the theoretical knowledge which I learnt in my university academics to practice. Especially areas like agile development, user experience, data base management, object oriented programming concepts and so many other things. Since the project scope included a web application tab application, smartphone application and a web service it helped me to get in depth idea about those technologies.

This project not only helped to improve my knowledge on technology jargon but also it helped me to improve my research skills and analytical skills as well. I was a great experiences to conduct real time interviews , questionnaire sessions and as an undergraduate and I was able to experience real time difficulties of conducting those sessions with real staff members and customers. It also helped me to prepare an effective questionnaires and interview questions. This project also helped me to learn the real meaning of punctuality, time management and patience.

One of the most important lesson I learnt in the development of the project is that serious discipline and methodical work planning is needed to finish a project. The beginning of the project was a clueless time period in which I did not know where to start and whom to talk to. But with the continuous encouragement from the supervisor I was put to the right track. After I started my industrial training development was considerably sped up because of the coding techniques and practices that I learnt.

I would also happy to say that I was fortunate enough to gain in depth knowledge about the insurance industry too.

## **9.5 Summary**

The final chapter is the conclusion on the whole of the project. The conclusion is that this was successful in attaining its objectives and ready to use with a few more modifications and enhancements. I have given my own experience in the author's note and what was my learning outcome of this project.

## References

Baecker, O., & Bereuter, A. (2010), *Technology-Based Industrialization of Claims Management in Motor Insurance*, Institute of Technology Management, University of St. Gallen.

Retrieved September 10<sup>th</sup> 2014, from the University of St. Gallen i-lab website:  
[http://www.i-lab.ch/docs/04\\_technology](http://www.i-lab.ch/docs/04_technology)  
based\_industrialization\_of\_claims\_management\_in\_motor\_insurance.pdf

Baecker O., Fleisch E., Ackermann W., Ackermann L 2010, Mobile Claims Management: Smartphone Apps in Motor Insurance, Trendmonitor 4 . 2010

Barry D., K. n.d, *Web Services Explained*, Viewed 9th December 2014  
[http://www.service-architecture.com/articles/web-services/web\\_services\\_explained.html](http://www.service-architecture.com/articles/web-services/web_services_explained.html)

Capegemini 2013, Mobile Solutions for Claims Management, Viewed 10th October 2014  
[http://www.capgemini.com/resource-file-access/resource/pdf/mobile\\_solutions\\_for\\_claims\\_management\\_0.pdf](http://www.capgemini.com/resource-file-access/resource/pdf/mobile_solutions_for_claims_management_0.pdf).

Dialog Axiata 2014, *Coverage Map*, Viewed 26<sup>th</sup> November 2014  
<http://www.dialog.lk/support/mobile/coverage-service-points/coverage-map/>

Flitch Ratings 2013, *2014 Outlook: Sri Lanka Insurance Sector*, Outlook Report.  
Retrieved September 17<sup>th</sup> 2014 from [www.fitchratings.lk/download-article-216.pdf](http://www.fitchratings.lk/download-article-216.pdf)

'Gartner' 2014, *Gartner Says Annual Smartphone Sales Surpassed Sales of Feature Phones for the First Time in 2013*, Viewed 29<sup>th</sup> October 2014  
<http://www.gartner.com/newsroom/id/2665715>

IBM 2009, Fast path claims from IBM. Viewed 10th October 2014  
<http://www-03.ibm.com/industries/financialservices/doc/content/solution/1231013103.html>

Iqbal I. 2014, *Rapid Growth of Mobile Internet Usage in Sri Lanka*, Viewed 25<sup>th</sup> November 2014

<http://infasiqbal.com/rapid-growth-of-mobile-internet-usage-in-sri-lanka/>

Lee C, Cheng H., K., Cheng H. 2007, an empirical study of mobile commerce in insurance industry: Task-technology fit and individual differences, *Decision Support Systems*.

Mobitel 2104, *Mobitel GSM and 3G Coverage*, Viewed 26<sup>th</sup> November 2014

[http://www.mobitel.lk/images/English/common/Srilanka\\_GSM\\_Coverage%2030112013.j pg](http://www.mobitel.lk/images/English/common/Srilanka_GSM_Coverage%2030112013.j pg)

Nielsen 2013, Sri Lanka 2013 review and opportunities in 2014, Viewed 10th October 2014

<http://www.nielsen.com/lk/en/insights/reports/2014/sri-lanka-2013-in-review-and-opportunities-in-2014.html>

Susan W. 2013, *Widespread Adoption of Tablets and Smartphones Is Driving Growth in the Internet of Things*, Viewed 29<sup>th</sup> October 2014

<https://www.scisbo.com/web/business-solutions/blog/-/blogs/widespread-adoption-of-tablets-and-smartphones-is-driving-growth-in-the-internet-of-things>

TDK Global 2012, *Devising Total Solution for Condensing a Multitude of Function within a Limited Space*, Viewed 29<sup>th</sup> October 2014

<http://www.global.tdk.com/ir/ir library/annual/2012/html/strategy/str01.htm>

Trading Economics 2014, *Mobile cellular subscriptions in Sri Lanka*, Viewed 29<sup>th</sup> November 2014

<http://www.tradingeconomics.com/sri-lanka/mobile-cellular-subscriptions-wb-data.html>

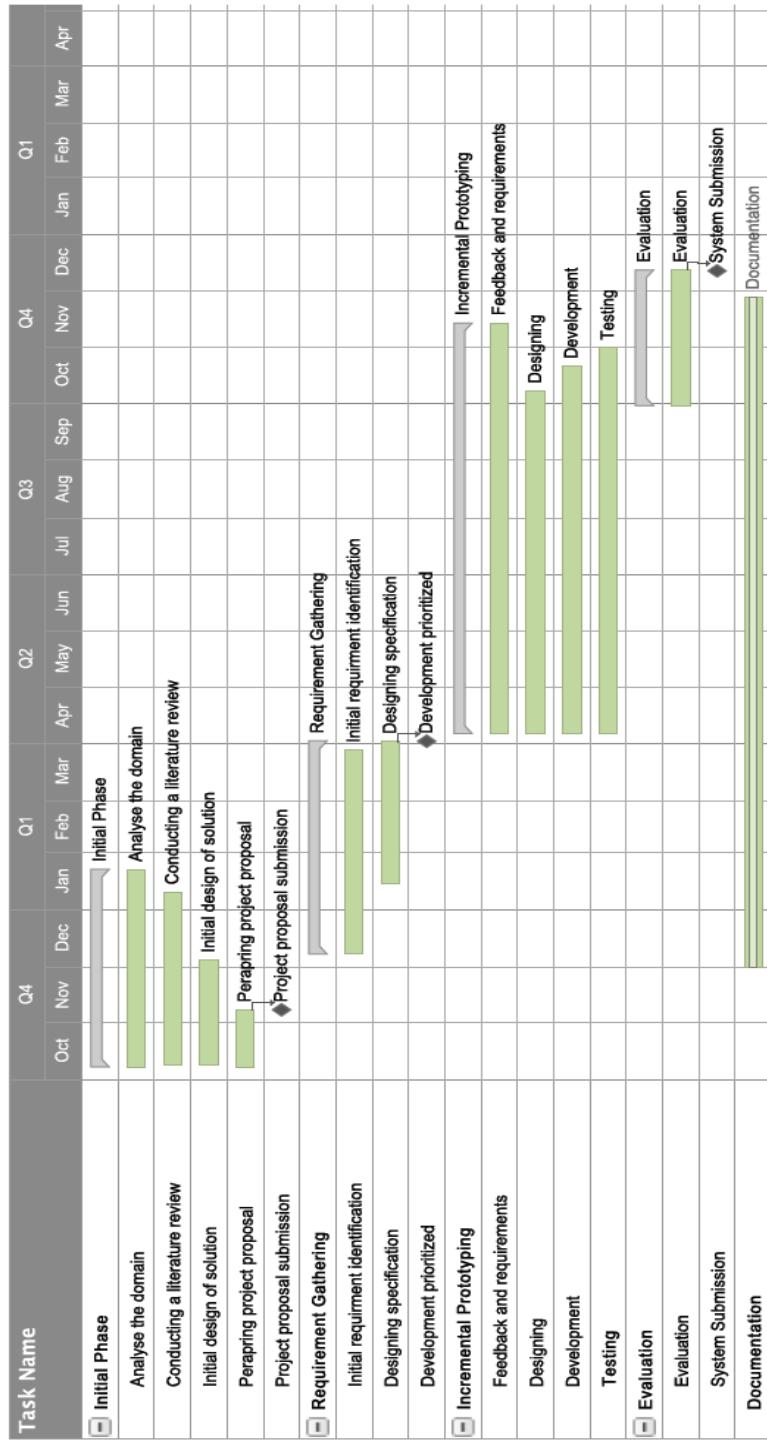
w3schools 2104, *Introduction to Web Services*, Viewed 9th December 2014

[http://www.w3schools.com/webservices/ws\\_intro.asp](http://www.w3schools.com/webservices/ws_intro.asp)

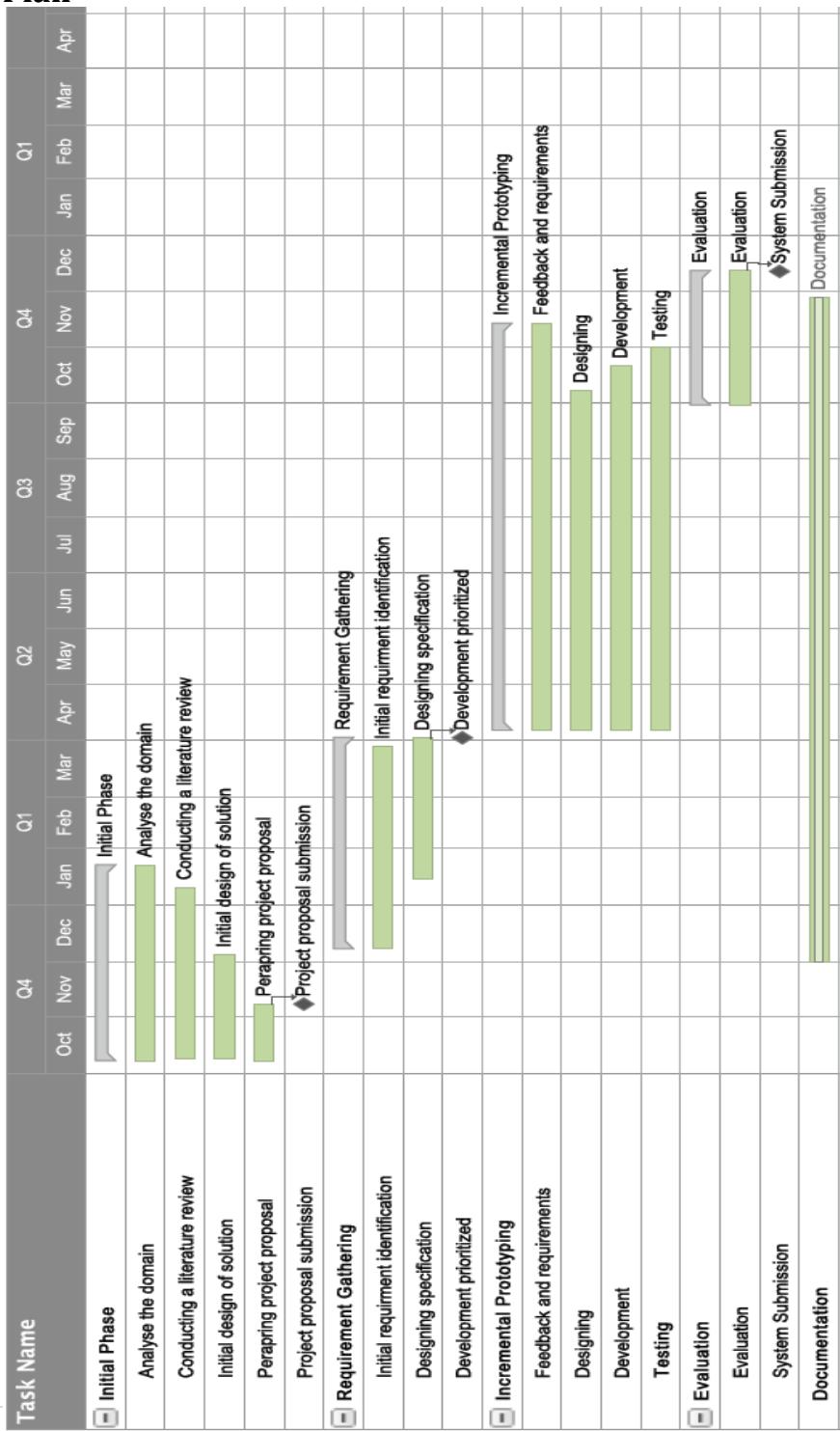
Want R 2009, When Cell Phones Become Computers, *IEEE Pervasive Computing*.

# Appendix A

## Planned Project Plan



## Actual Project Plan



Major change that was seen between planned and actual plan was the project methodology. Initially it was planned to do the project in a methodology that is lean more towards the traditional approach. But after realizing the importance in agile methodology projects requirement identification, designing development, testing were done concurrently. Constant user feedback was a big advantage for the developers.

As this is an individual project, first six month was conducted while studying at the university. In that period also there were allocating time for development and documentation while doing loads of assessment works and exams. After exams were over industrial trainings started. After going to the industry also time was more limited to the project. Slight deviations in the plan are caused by those factors. But as an overall project was conducted within the time period very successfully.

# Appendix B

## Data Gathering and Analysis

### Pre Prepared Question Plan for Interviews

Question	Resource Person Type
What are your responsibilities? Can you explain your task briefly?	Claim Agent
What are the difficulties you face when travelling to the accident location?	Claim Agent
What kind of information and equipment do you carry when you travel to accident locations.	Claim Agent
How do you validate the client information with company database?	Claim Agent
What kind of difficulties do you face when handling claim requests?	Operating Staff
What do you think about current practices conducted when assessing claims?	Managerial Level
Do you think automating the claim assessment to some extent will help to increase profit margins?	Managerial Level

Table 12: Interview question plan

Source: Author

## **Summary of Fact Findings**

Conducting interviews with insurance sector employees from time to time in the development process helped me to understand the types of employees, agents and their current performance as average, power of human resource, level of technological skills of employees and their general attitude and capability of being flexible for any type of technologically sophisticated environment. Further it was discussed about the steps of creating vehicle insurance policy, customer registration, detail description about the agreements, legal situation, especially how they give approval to the claim currently and existing verification methods and current annual and premium payment methods.

Drawbacks of the current procedure were found out by interviewing customers and internal stakeholders. As an example customer have to spend lot of time in order to collect their claiming amount, and sometimes customers are receiving lesser amount than what they deserve due to inaccurate assessment performed by the claiming insurance agent. Company also face so many problems by using the current procedure, like, when the assessed claiming amount is more than the actual claiming amount due to the manual calculation performed by the agent.

After several interviews with claim agents, their main roles and responsibilities were identified. They discussed their difficulties when handling with documents and other equipment's such as camera as well. After getting friendly with them it led to the conduct of observation technique as well. As the developer I got a chance to witness a real time claim assessment procedure.

Apart from interviews and observations other main source of data were the document reviews. Certain insurance registrations were provided. Not only that claim assessment forms also were supplied for authors reference which helped author to design the database. Since the resource persons have asked not to publish the forms sample are not attached.

# Appendix C

## System Design

### Database Tables

This table contains the payment details related to a claim record. Payment id act as a foreign key to claim table. The below tables driver table, spare part payment table, third party details tables also contain details related to a particular claim. Therefore they are also connected to claim tables via foreign key.

Claim Payment Table		
Attribute	Data Type	Length
Paymentid	Int	-
Garagecost	Float	-
Othercosts	Float	-
Deductions	Float	-
InsurancePercentage	Float	-

Table 13: Claim Payment Table

Source: Author

Driver Table		
Attribute	Data Type	Length
Driverid	Int	-
Drivername	nvarchar	50
Licenceno	nvarchar	20
Expirationdate	DateTime	15
Drivernic	nvarchar	20

Table 14: Driver Table

Source: Author

Spare part Payment Table		
Attribute	Data Type	Length
Paymentid	Int	-
Sparepartid	Int	-
Sparepartqty	Float	-
Sparepartcost	Float	-

Table 15: Spare part Payment Table:

Source: Author

Third Party Details Table		
Attribute	Data Type	Length
Thirdpartydetailid	Int	-
Vehiclerегистrationno	nvarchar	50
Ownername	nvarchar	50
Owneraddress	nvarchar	100
Reneweldate	DateTime	10
Specialnotes	nvarchar	Max
Victimname	nvarchar	50
Victimaddress	nvarchar	100
Damagenature	nvarchar	100
Thirdpartyclaimant	nvarchar	50
Claimamount	Float	-

Table 16: Third Party Table

Source: Author

Vehicle Table		
Attribute	Data Type	Length
Vehicleid	Int	-
Vehicletypeid	Int	-
Manufactureid	Int	-
Model	nvarchar	25
Makеyear	DateTime	10
Fueltype	nvarchar	10
Enginеcapacity	nvarchar	20
Seatingcapacity	nvarchar	10
Carryingcapacity	nvarchar	10
Presentvalue	Float	-
Dutyfreevalue	Float	-

Table 17: Vehicle Table

Source: Author

Customer Vehicle Table		
Attribute	Data Type	Length
Vehicleid	Int	-
Customerid	Int	-
Registrationno	nvarchar	15
Color	nvarchar	20
Enginenо	nvarchar	50

Chassisno	nvarchar	50
Currentdamages	nvarchar	Max
Absoluteowner	nvarchar	10
Financialrights	nvarchar	50
Extrafittins	nvarchar	Max
Usage	nvarchar	Max

Table 18: Customer Vehicle Table

Source: Author

Claim request table holds claim request data which are sent by the clients. When a new record is added state will be always pending. After an agent respond to it they can change it to responded state.

Claim Request Table		
Attribute	Data Type	Length
Id	Int	-
Policyid	Int	-
Gps	nvarchar	50
State	nvarchar	50
Submittime	DateTime	-
Respondtime	DateTime	-

Table 19: Claim Request Table

Source: Author

Manufacturer Table		
Attribute	Data Type	Length
ManufactureId	Int	-
ManufactureName	nvarchar	50

Table 20: Manufacturer Table

Source: Author

Spare part Category Table		
Attribute	Data Type	Length
Sparecategoryid	Int	-
Sparecategoryname	nvarchar	50

Table 21: Spare part Category Table

Source: Author

Spare part Table		
Attribute	Data Type	Length
Sparepartid	Int	-
Sparepartname	nvarchar	50
Sparepartcategory	Int	-
Sparepartmanufacturer	Int	-

Spareparmanufacyear	nvarchar	10
Sparepartunitcost	Float	-

Table 22: Spare part Table

Source: Author

Vehicle Type Table		
Attribute	Data Type	Length
VehicleTypeID	Int	-
VehicleTypeName	nvarchar	50

Table 23: Vehicle Type

Source: Author

Garage Table		
Attribute	Data Type	Length
GarageID	Int	-
GarageName	nvarchar	50
GarageLocation	nvarchar	50
GarageTP	nvarchar	20
Email	nvarchar	100

Table 24: Garage Table

Source: Author

Tow Truck Table		
Attribute	Data Type	Length
Id	Int	-
Name	nvarchar	50
Location	nvarchar	50
Telephone	nvarchar	20
Email	nvarchar	100

Table 25: Tow Truck Table

Source: Author

Employee Table		
Attribute	Data Type	Length
EmpId	Int	-
EmpName	nvarchar	50
EmpPhone	nvarchar	20
EmpEmail	nvarchar	100
EmpBranch	nvarchar	20

Table 26: Employee Table

Source: Author

User Accounts Table		
Attribute	Data Type	Length

Account_Id	Int	-
Employee_id	Int	-
Username	nvarchar	50
Password	nvarchar	50
Type	nvarchar	20

Table 27: User Account Table

Source: Author

## Interfaces of the Online Claiming System (Web Application)

Interface for viewing garages. You can also select add/edit/remove options from this view. You can filter by location also in this view.

Garages Information				
ID	Name	Location	Contact No	Email
3006	RinShow Garages	Badulla	0552223547	isankalakshan@gmail.com
1006	CleanStar	Kollupitiya	0112456879	isankalakshan@gmail.com
1004	Kleen Vehicle	Mathugama	0768993244	isankalakshan@gmail.com
1	Wickramarathne Service Centre	Rathmalana	0711233445	4 isankalakshan@gmail.com
1005	Ishanka Garages	Rathmalana	0716405220	isankalakshan@gmail.com

Figure 50: Garages Interface

Source: Author

Interface to select add a new garage.

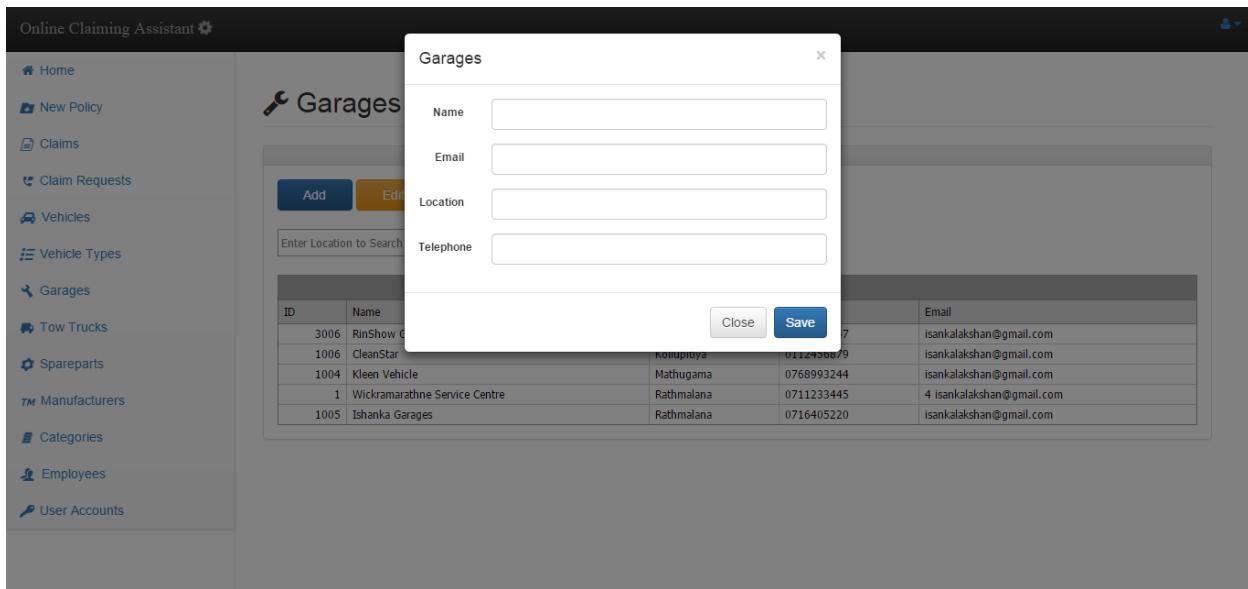


Figure 51: Add New Garage Interface

Source: Author

Interface of editing a garage. First you have to select a record from the data grid and click edit. Then it will be loaded like shown.

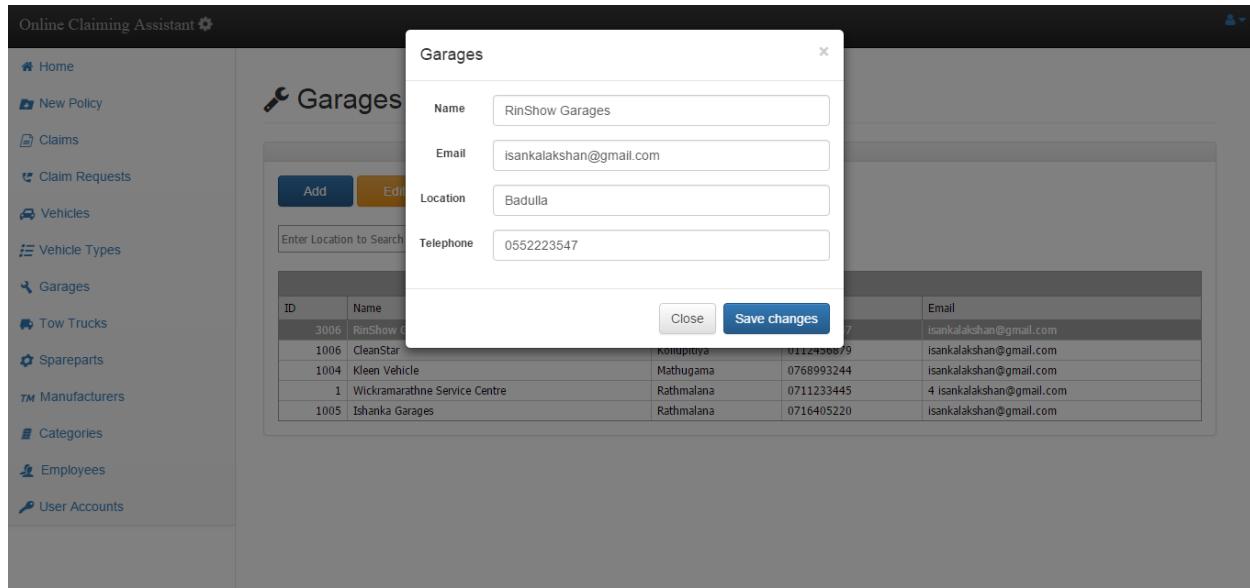


Figure 52: Edit Garage Interface

Source: Author

Interface for viewing vehicles. You can also select add/edit/remove options from this view. You can filter by several parameters in this view.

Vehicle Information										
Type	Manufacturer	Model	Year	Fuel	Engine	Seating	Carrying	Present Value	Duty Free	
Bike	Hero Honda	4 stroke	2015	Diesel	400cc	2	4	400000	5200000	
Bike	Nissan	Prius	2010	Hybrid	1500cc	5	750kg	2100000	1500000	
Car	Toyota	Allion	2012	Diesel	1400cc	3	5	2300000	0	

Figure 53: Vehicles Interface

Source: Author

Interface for viewing manufacturers. You can also select add/edit/remove options from this view. You can filter by name also in this view.

ID	Name
8	Hero Honda
4	Honda
5	Mazda
3	Micro
6	Mitsubishi
2	Nissan
7	Tata
1	Toyota
9	Toyota

Figure 54: Manufacturers Interface

Source: Author

Interface for viewing spare part Category. You can also select add/edit/remove options from this view. You can filter by name also in this view.

The screenshot shows a web-based application interface titled "Sparepart Category". At the top left is a sidebar with various menu items: Home, New Policy, Claims, Claim Requests, Vehicles, Vehicle Types, Garages, Tow Trucks, Spareparts, Manufacturers, Categories, Employees, and User Accounts. The "Spareparts" item is currently selected. The main content area has a title "Sparepart Category" with a small icon. Below it are three buttons: "Add" (blue), "Edit" (orange), and "Delete" (red). A search bar labeled "Enter Location to Search" is present. A table titled "Sparepart Category Information" lists categories with their IDs and names. The table has two columns: "ID" and "Name". The data is as follows:

ID	Name
4	Engine
7	Head Lights
5	Internal
6	Side Mirror
8	Tires
2	Windscreen

Figure 55: Spare part Category Interface

Source: Author

Interface for viewing garages. You can also select add/edit/remove options from this view. You can filter by location also in this view.

The screenshot shows the 'Tow Truck Services' section of the Online Claiming Assistant. On the left sidebar, there are various navigation links including Home, New Policy, Claims, Claim Requests, Vehicles, Vehicle Types, Garages, Tow Trucks, Spareparts, Manufacturers, Categories, Employees, and User Accounts. The main content area has a title 'Tow Truck Services' with a search bar labeled 'Enter Location to Search'. Below it is a table titled 'Garages Information' with columns for ID, Name, Location, Contact No, and Email. The table contains four rows of data.

ID	Name	Location	Contact No	Email
1	DragMe	Badulla	0552224598	isankalakshan@gmail.com
1003	Ishanka Tow trucks	Badulla	0552224568	isankalakshan@gmail.com
2	PickUp	Colombo	0112547896	isankalakshan@gmail.com
3	PullMe	Mathugama	0386456752	isankalakshan@gmail.com

Figure 56: Tow Truck Services

Source: Author

Interface for viewing employees. You can also select add/edit/remove options from this view. You can filter by name and branch in this view.

The screenshot shows the 'Employees' interface of the Online Claiming Assistant. The left sidebar includes links for Home, New Policy, Claims, Claim Requests, Vehicles, Vehicle Types, Garages, Tow Trucks, Spareparts, Manufacturers, Categories, Employees, and User Accounts. The main area features a title 'Employees' with two search input fields: 'Enter Name to Search' and 'Enter Branch to Search'. A table titled 'Employee Information' displays data across four columns: Name, Telephone, Email, and Branch. There is one entry in the table.

Name	Telephone	Email	Branch
Ishanka	0716405220	isankalakshan@gmail.com	Badulla

Figure 57: Employee Interface

Source: Author

# Appendix D

## Testing and Implementation

Test log for the web application is attached in the next page.

Module	Test Case	Expected Result	Steps	Actual Result	Remark
Login Module	Test the result when login button pressed without giving credentials	Show enter username & password error message	1. Enter the URL and enter to the system 2. Press login without providing credentials	Error message showed	<b>Success</b>
	Test the result when tried to login with invalid credentials	Show invalid username or password error message	1. Enter the URL and enter to the system 2. Enter some invalid credentials 3. Press login button	Error message showed	<b>Success</b>
	Test the result when tried to login with valid credentials	Forward to the home/main menu	1. Enter the URL and enter to the system 2. Enter some valid credentials 3. Press login button	Main menu loaded	<b>Success</b>
	Test whether session has being created & session variables are initiated	Employee name should appear in top right	1. Enter the URL and enter to the system 2. Enter some valid credentials 3. Press login button	Name appeared correctly	<b>Success</b>
New Policy Module	When New policy tile clicked form for new policy should appear	New policy form modal should pop up	1. Enter the URL and enter to the system 2. Login to the system 3. Click on New policy tile	New policy modal loaded	<b>Success</b>
	All the required fields must be entered before submitting	Should display provide all required fields error message	1. Enter the URL and enter to the system 2. Login to the system 3. Click on New policy tile 4. Leave empty some required fields and submit	Success error message displayed	<b>Success</b>
	Ability to insert a new policy successfully	Should display successfully entered message	1. Enter the URL and enter to the system 2. Login to the system 3. Click on New policy tile	successfully entered message displayed	<b>Success</b>

			4. Fill all required fields and submit	
Claims Module	When clicked claims tile load the claims submitted summary view	Should display the claims interface with data grid	1. Enter the URL and enter to the system 2. Login to the system 3. Click on claims tile	Claims interface loaded
	Test the view button function	Should display detailed view of claims	1. Enter the URL and enter to the system 2. Login to the system 3. Click on claims tile 4. Select a claim record from data grid 5. Press view button	Detailed view of claim was displayed on a modal
	Test the approve button function in detailed view of a claim	Should display the approved message.	1. Enter the URL and enter to the system 2. Login to the system 3. Click on claims tile 4. Select a claim record from data grid 5. Press view button 6. Press approve button	Approved message displayed
Claim Requests Module	When clicked claim requests tile load the respective interface	Should display the claims requests interface with data grid	1. Enter the URL and enter to the system 2. Login to the system 3. Click on claim requests tile	Claims requests interface loaded with data grid
	Load the coordinates on a map	Should display a map pinpointing the exact GPS location	1. Enter the URL and enter to the system 2. Login to the system 3. Click on claim requests tile 4. Select a request from data grid 5. Press Map button	Map loaded

	Change the state of a claim request to responded	State should be changed to responded	1. Enter the URL and enter to the system 2. Login to the system 3. Click on claim requests tile 4. Select a request from data grid 5. Press Responded button	State changed to responded  <b>Success</b>
Vehicles Module	Click the vehicles tile to load interface with filled data grid	Should display the vehicles details in data grid data grid	1. Enter the URL and enter to the system 2. Login to the system 3. Click on vehicles tile	Interface displayed with loaded data grid  <b>Success</b>
	Add a new vehicle	Should add to database and display in data grid	1. Enter the URL and enter to the system 2. Login to the system 3. Click on vehicles tile 4. Click add button 5. Fill the details and press save	Added and displayed in the data grid  <b>Success</b>
	Edit vehicle details	Should submit to database and update the data grid	1. Enter the URL and enter to the system 2. Login to the system 3. Click on vehicles tile 4. Click a record on data grid and press edit 5. Fill the details and press save changes	Submitted to database and updated the data grid  <b>Success</b>

Remove a vehicle	Should remove from database and update the data grid	<ol style="list-style-type: none"> <li>1. Enter the URL and enter to the system</li> <li>2. Login to the system</li> <li>3. Click on vehicles tile</li> <li>4. Click a record on data grid and press Remove</li> <li>5. Confirm by clicking remove.</li> </ol>	Removed from database and refreshed the data grid	<b>Success</b>
Add a new spare part	Should add to database and display in data grid	<ol style="list-style-type: none"> <li>1. Enter the URL and enter to the system</li> <li>2. Login to the system</li> <li>3. Click on spare part tile</li> <li>4. Click add button</li> <li>5. Fill the details and press save</li> </ol>	Added and displayed in the data grid	Success
Edit spare part details	Should submit to database and update the data grid	<ol style="list-style-type: none"> <li>1. Enter the URL and enter to the system</li> <li>2. Login to the system</li> <li>3. Click on spare part tile</li> <li>4. Click a record on data grid and press edit</li> <li>5. Fill the details and press save changes</li> </ol>	Submitted to database and updated the data grid	<b>Success</b>
Remove a spare part	Should remove from database and update the data grid	<ol style="list-style-type: none"> <li>6. Enter the URL and enter to the system</li> <li>7. Login to the system</li> <li>8. Click on spare part tile</li> <li>9. Click a record on data grid and press Remove</li> <li>10. Confirm by clicking remove.</li> </ol>	Removed from database and refreshed the data grid	<b>Success</b>

Add a new spare part category	Should add to database and display in data grid	1. Enter the URL and enter to the system 2. Login to the system 3. Click on category tile 4. Click add button 5. Fill the details and press save	Added and displayed in the data grid	Success
Edit spare part category details	Should submit to database and update the data grid	1. Enter the URL and enter to the system 2. Login to the system 3. Click on category tile 4. Click a record on data grid and press edit 5. Fill the details and press save changes	Submitted to database and updated the data grid	Success
Remove a spare part category	Should remove from database and update the data grid	11. Enter the URL and enter to the system 12. Login to the system 13. Click on spare part category tile 14. Click a record on data grid and press Remove 15. Confirm by clicking remove.	Removed from database and refreshed the data grid	Success
Add a new manufacturer	Should add to database and display in data grid	1. Enter the URL and enter to the system 2. Login to the system 3. Click on manufacturer tile 4. Click add button 5. Fill the details and press save	Added and displayed in the data grid	Success
Edit manufacturer details	Should submit to database and update the data grid	1. Enter the URL and enter to the system 2. Login to the system 3. Click on manufacturer tile	Submitted to database and updated the data grid	Success

		<p>4. Click a record on data grid and press edit</p> <p>5. Fill the details and press save changes</p>		
Remove a manufacturer	Should remove from database and update the data grid	<p>16. Enter the URL and enter to the system</p> <p>17. Login to the system</p> <p>18. Click on manufacturer tile</p> <p>19. Click a record on data grid and press Remove</p> <p>20. Confirm by clicking remove.</p>	Removed from database and refreshed the data grid	Success
Add a new vehicle type	Should add to database and display in data grid	<p>1. Enter the URL and enter to the system</p> <p>2. Login to the system</p> <p>3. Click on vehicle type tile</p> <p>4. Click add button</p> <p>5. Fill the details and press save</p>	Added and displayed in the data grid	Success
Edit vehicle type details	Should submit to database and update the data grid	<p>6. Enter the URL and enter to the system</p> <p>7. Login to the system</p> <p>8. Click on vehicle type tile</p> <p>9. Click a record on data grid and press edit</p> <p>10. Fill the details and press save changes</p>	Submitted to database and updated the data grid	Success
Remove a vehicle type	Should remove from database and update the data grid	<p>21. Enter the URL and enter to the system</p> <p>22. Login to the system</p> <p>23. Click on vehicle type tile</p>	Removed from database and refreshed the data grid	Success

		24. Click a record on data grid and press Remove 25. Confirm by clicking remove.		
Add a new garage	Should add to database and display in data grid	1. Enter the URL and enter to the system 2. Login to the system 3. Click on garage tile 4. Click add button 5. Fill the details and press save	Added and displayed in the data grid	Success
Edit garage details	Should submit to database and update the data grid	11. Enter the URL and enter to the system 12. Login to the system 13. Click on garage tile 14. Click a record on data grid and press edit 15. Fill the details and press save changes	Submitted to database and updated the data grid	Success
Remove a garage	Should remove from database and update the data grid	26. Enter the URL and enter to the system 27. Login to the system 28. Click on garage tile 29. Click a record on data grid and press Remove 30. Confirm by clicking remove.	Removed from database and refreshed the data grid	Success
Add a new tow truck service	Should add to database and display in data grid	1. Enter the URL and enter to the system 2. Login to the system 3. Click on tow truck service tile 4. Click add button 5. Fill the details and press save	Added and displayed in the data grid	Success

Edit tow truck service details	Should submit to database and update the data grid	16. Enter the URL and enter to the system 17. Login to the system 18. Click on tow truck service tile 19. Click a record on data grid and press edit 20. Fill the details and press save changes	Submitted to database and updated the data grid	Success
Remove a tow truck service	Should remove from database and update the data grid	31. Enter the URL and enter to the system 32. Login to the system 33. Click on tow truck service tile 34. Click a record on data grid and press Remove 35. Confirm by clicking remove.	Removed from database and refreshed the data grid	Success
Add an employee	Should add to database and display in data grid	6. Enter the URL and enter to the system 7. Login to the system 8. Click on employee tile 9. Click add button 10. Fill the details and press save	Added and displayed in the data grid	Success
Edit employee details	Should submit to database and update the data grid	21. Enter the URL and enter to the system 22. Login to the system 23. Click on employee tile 24. Click a record on data grid and press edit 25. Fill the details and press save changes	Submitted to database and updated the data grid	Success

Remove a employee	Should remove from database and update the data grid	36. Enter the URL and enter to the system 37. Login to the system 38. Click on employee tile 39. Click a record on data grid and press Remove 40. Confirm by clicking remove.	Removed from database and refreshed the data grid	Success
-------------------	--	---	---	---------

Table 28: Test Log for Web Application

Source: Author

