

**DESIGN & DEVELOPMENT OF A WEB BASED APPLICATION FOR A
COMPLAIN MANAGEMENT SYSTEMS**

BY

**MOST. MAROFA KHATUN
ID: 122-15-1962**

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

MD. Mahmudul Hasan
Senior Lecturer
Department of CSE
Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY

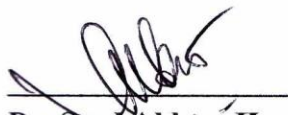
DHAKA, BANGLADESH

AUGUST 2015

APPROVAL

This Project titled “**Design & Development of a Web Based Application for a Complain Management Systems**”, submitted by Most. Marofa Khatun, ID- 122-15-1962 to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 20 August 2015.

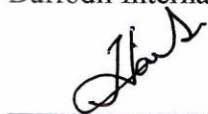
BOARD OF EXAMINERS



Dr. Syed Akhter Hossain
Professor and Head

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

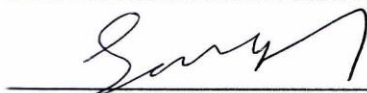
Chairman



Dr. Sheak Rashed Haider Noori
Assistant Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Muhammad Sarawar Jahan Morshed
Assistant Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Dr. Mohammad Shorif Uddin
Professor and Chairman


Department of Computer Science and Engineering
Jahangirnagar University

External Examiner

DECLARATION

I hereby declare that, this project has been done by us under the supervision of **MD. Mahmudul Hashan, Senior Lecturer, and Department of CSE Daffodil International University**. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

_____

Md. Mahmudul Hasan

Senior Lecturer

Department of CSE

Daffodil International University

Submitted by:

_____

Most. Marofa Khatun

ID: 122-15-1962

Department of CSE

Daffodil International University

ACKNOWLEDGEMENT

First I express my heartiest thanks and gratefulness to almighty Allah for His divine blessing me possible to complete this project successfully. It's my pleasure to take this occasion to thank a few people who have assisted, encouraged, directed and supported us throughout our project-work.

Firstly, I want to pay our gratitude to Md.Mahmudul Hasan, Senior Lecturer, Department of CSE and Mr. Sadekur Rahman, Lecturer, Department of CSE, Daffodil International University Bangladesh for active cooperation and for valuable time towards enhancing the understanding of the systems and their guidelines throughout the period of preparing the report.

Finally, I would like to express our appreciations and gratefulness to them for their cooperation without which it would not have been possible for us to learn so many different things and work so closely on a web based Application.

ABSTRACT

The purpose of this project is to build a web application which manages complain to provide good governance. This applications can enhance the capabilities of the administrative sectors of our country. It is developed to help people to find out their missing vehicles which might got stolen unfortunately. On the other hand, there is additional feature in this application which will be able to receive any kinds of complains which are happened on the roadside or anywhere else around the country. Hence, people will require safety zone to drop their complain and therefore, the web-based application can rectify a safety zone detection feature which will detect and show the nearest safety zone while any unwanted incidents occurs. Lastly, the web application will have all the emergency contacts throughout the country and will only show the nearest possible contact of respective police station from wherever the person request to provide helps.

TABLE OF CONTENT

CONTENS	PAGE
Board of examiners	i
Declaration	ii
Acknowledgements	iii
Abstract	iv
CHAPTER	
Chapter 1: Introduction	01-03
1.1 Overview	1
1.2 Objectives	2
1.3 Working Report	2
1.4 Technology	3
1.5 Tools	3
Chapter 2: Methodology Analysis	04-08
2.1 Methodology	4
2.2 The Waterfall Model	5
2.3 Structured System Analysis and Design Method (SSADM)	6
2.4 Web IS Development Methodology (WISDM)	7
2.5 Software Prototyping	8
Chapter 3: Requirements Analysis	10-21
3.1 Web Technical Standards	10
3.1.1 Usability	10

3.2 User Interfaces	10
3.3 Web Standards	11
3.4 Ease of Maintenance	12
3.5 Technical Analysis	12
3.5.1 Presentation Layer	13
3.5.1.1 HTML	13
3.5.1.2 XHTML	13
3.5.1.3 Cascading Style Sheets (CSS)	14
3.5.1.4 Evaluation of Presentation Layer Technology	14
3.5.2 Client-side & server-side Technologies	15
3.5.2.1 JavaScript-Client-side	15
3.5.2.2 Pre Hypertext Processor (PHP)- Server-side	15
3.5.3 Server-side Considerations	16
3.5.3.1 Security	16
3.5.3.2 Performance	17
3.5.4 Web Server Technologies	17
3.5.5 Database Technologies	18
3.5.5.1 MySQL	18
3.5.5.2 Evaluation of Database Technologies	18
3.5.5.3 Chosen Technologies	19
3.6 Systems Analysis	19
3.6.1 Why Do Systems Fail?	19
3.6.2 User Requirements	20
3.6.3 Feasibility Study	20
3.6.4 Technical	21
3.6.5 Economical	21
Chapter 4: Design	22-30
4.1 Database	22
4.1.1 Table Schemas	22

4.1.2 Entity Relational Diagram	23
4.1.3 Integrity Constraints	25
4.1.4 Normalization	25
4.2 Server-side Considerations	26
4.2.1 Server-side Security	26
4.2.2 Performance & Maintainability	26
4.3 Presentation Layer	26
4.3.1 Web Interface	26
4.3.2 Navigation	27
4.3.3 Colors	28
4.4 System Use-case diagram	29
4.4.1 Secure Area	30
Chapter 5: Implementation	31-35
5.1 Database	31
5.2 Server-side Considerations	31
5.2.1 Security	31
5.2.2 Performance and Maintainability	31
5.3.1 Layout	31
5.3.2 Navigation	32
5.4 Secure Area	32
5.4.1 Login	32
5.4.2 User Area	33
5.4.3 Complain for Harassment	34
5.4.4 Vehicle Missing Complain Box	35
Chapter 6: Testing	36-37
6.1 Usability Inspection	36
6.2 Unit Testing	36

6.3 Browser Testing	37
Chapter 7: Conclusion	38
7.1 Conclusion & Future Work	38
7.2 Achievement	38
7.3 Limitations	38
APPENDIX	39
REFERENCES	40

LIST OF FIGURES

FIGURES	PAGE NO
Figure 2.1: Development Methodology	6
Figure 4.1: Entity Relational Diagram	24
Figure 4.2: User Home Page	27
Figure 4.3: User Dashboard Main Menu	27
Figure 4.4: System Use-case diagram	29
Figure 5.1: Login Area	32
Figure 5.2: User Dashboard	33
Figure 5.3: Complain for harassment	34
Figure 5.4: Vehicle Missing Complain Box	35
Figure Appendix: UML Class Diagram	39

CHAPTER 1

Introduction

1.1 Overview

This is a web-based software project which will ease people's life with a solution of theft, harassment, misconduct and complaint management. These four major things are closely related with our daily life, where we can have peace from different kinds of unexpected happenings. So, no more hectic days, no more car missing, no more harassment, eve-teasing and so on.

As an underdeveloped country, Bangladesh is completely affected with theft and various kinds of misconduct. In addition, our country is far behind from the developed country in technological development as well. This is a very common news we see almost every single day while opening our morning newspaper that, someone's car has been stolen, a girl has been teased/raped, people taking bribes and honest people are not getting justice at all. Hence, we are here to initiate a software for the first time in Bangladesh which will let general people connect with the final administration of the country. These four modules of the web-based software will initially reduce and ultimately aims to remove theft, harassment and all kinds of misconducts with the help of general, honest and patriotic people around the country.

Right now, we are initiating this software as web-based. In near future, we aim to make it as a mobile app-based software- which will become much easier for people to respond quickly on the spot.

1.2 Objective

The main objectives of the report are as follows:

- To provide service to vehicle owners for missing vehicles.
- People Find Emergency Contact Number
- To Provide Police Control Room any Complaint.
- People Find nearest Safety Zone.
- To learn how to make web design and development, software Planning.
- To acquire practical knowledge of Web Development.

1.3 Working Report

- Front-End
 - Sign Up
 - Sign In
- User Panel
 - Dashboard
 - Create a Complaining
 - vehicles Missing Complain
 - Find Emergency Contact Number
 - Find Google Map Nearest Safety Zone
 - Message
 - User Profile Setting
- Admin Panel
 - Dashboard
 - User [add/ edit/ delete/ view/ block]
 - Location
 - Massage
 - Admin Profile Setting
 - Complain Box

1.4 Technology

- Doctrine Query Language
- GMAP JS Plugin
- PHP5
- HTML5
- Bootstrap
- CSS3
- MySQL DB
- Jaquary

1.5 Tools

- Notepad++
- WampServer
- SQLyog MySQL GUI
- Firebug 2.0.11
- File management tool FileZilla
- Google chrome

CHAPTER 2

Methodology Analysis

2.1 Methodology

Prior to the existence of software development methodologies, systems were usually poorly designed with the final product failing to satisfy user requirements. Early approaches to software development placed too greater an emphasis on programming, and the absence of any predefined structure and methodology meant project scheduling dates and milestones were never met on time. explains that programmers would *“instead use their rule of thumb and rely on experience;”* evidently an invaluable asset to any software development project, but inadequate and unreliable when considering it as the sole factor for approaching development [1].

Methodologies provide a more systematic approach to software development, Adison and defines system design methodologies as *“A collection of procedures, techniques, Tools and documentation aids which will help the system developers in their efforts to implement a new information system”*. This approach clearly defines tasks, providing guidance in order to fulfil these tasks, allowing greater management control and thus avoiding cost and schedule overruns. Therefore, it is vital that the project follows a methodical approach and has a well-defined structure, and with a wide varying range of methodologies available it is of paramount importance that one is chosen that is appropriate for the project. In turn this section will review a number of methodologies available such as the Waterfall model, SSADM, WISDM and Prototyping; the chosen methodology will dictate the overall approach of the project and subsequently will be mirrored in this report.

2.2 The Waterfall Model

The Waterfall model otherwise known as the Information Systems Development Life Cycle (SDLC) is considered the original model, depicting the classic software life cycle. This software development process is split into 6 well-devised phases as illustrated in Appendix, and each must be completed before the next stage commences [2].

The Waterfall model has been widely adopted because of its clear separation allowing documents to be well defined for each step, enabling management to inspect the development process and assess its progress. This allows for a more accurate prediction of project schedules, and helps to prevent delays and further costs incurred.

There is an adaptation to this model which addresses the many apparent flaws of the waterfall model. The Waterfall model with iterative feedback begins with the normal phases, starting with gathering requirements, then design phase, implementation and evaluation. After this first iteration user feedback is provided, and any problems with unclear requirements, coding problems or other issues are noted and used to set the objectives of the second iteration. After the necessary number of iterations has been performed, the final two stages are completed. However, the developer has to be careful about this modified approach, because too many iterations will potentially allow the project to deviate from the waterfalls rigid development structure; resulting in a more prototyping approach. Therefore, careful consideration needs to be paid towards the amount of iterations allowed, ensuring that schedules and milestones are still maintained.

Development Methodology (Diagrams)

Classic software life-cycle (Waterfall model):

The following figure 2.1 shows Development Methodology.

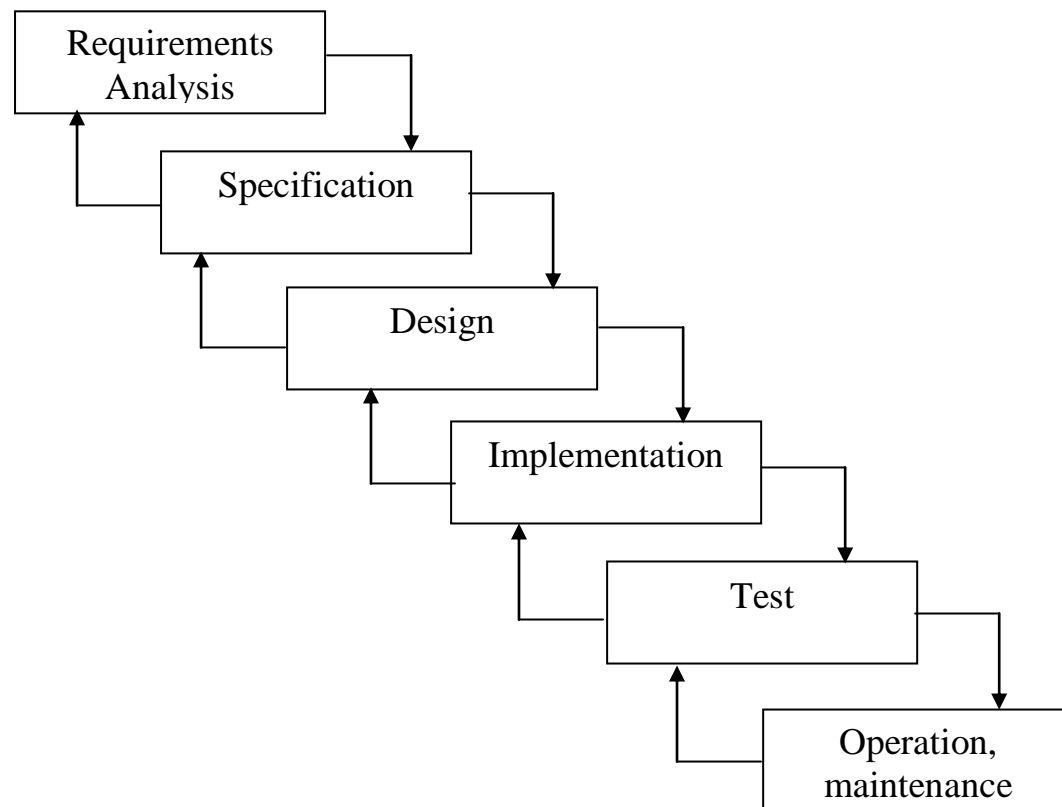


Figure 2.1: Development Methodology [3].

2.3 Structured System Analysis and Design Method (SSADM)

SSADM is a widely used development method in the UK, and is recognized as a formally specified British standard since 1994. Similar to the Waterfall model, SSADM provides a framework for managing the project in distinct steps, which lead on to the next step, although this approach focuses more on the earlier stages of the systems life cycle, in particular analysis and design. The emphasis placed on these earlier sections means user needs and requirements are communicated to the developer from the outset, ensuring the system closely matches user specifications. Overall this approach has been structured to avoid the shortcomings of the Waterfall

model providing a framework for 7 self-contained stages, with the ability to view the system from three distinct perspectives. Allowing developers to cross-check and compare each view to ensure consistency and completeness throughout.

Logical Data Modelling - The process of identifying, modelling and documenting the data requirements of the system being designed.

Data Flow Modelling - The process of identifying, modelling and documenting how data moves around the system

Entity Behavior Modelling - The process of identifying, modelling and documenting the events that affect each entity and the sequence in which these events occur [4].

There are several drawbacks concerned with this approach. SSADM is a prescriptive approach, therefore following all the steps may take some time before users actually sees a concrete deliverable, and by that stage the solution offered may not be fully satisfactory. As Sand hill explains, “there is a danger of 'paralysis by analysis', where much time is spent overanalyzing the problem that the project fails to make satisfactory headway”.

2.4 Web is Development Methodology (WISDM)

This methodology was proposed by Virden and is an application of the Multi view model for the purpose of developing web applications. WISDM “provides a framework for bringing together traditional systems development methods with web-based techniques to provide a rounded framework that runs from e-business strategic analysis through to implementation in software” An example of this can be seen in the adaptation of the ISD methods matrix

WISDM’s approach is divided in to 4 phases beginning with a thorough analysis of the organization who want the new system “the overall aim of organizational analysis is the consideration of how value will be added”. In the information analysis stage the systems requirements are captured and the last two stages see the system designed from a participative and physical perspective. The main advantage to this approach is

its primary use in the development of web applications, important because of its specific relevance to the projects problem domain. Furthermore WISDM's places emphasis on stakeholder involvement using UML techniques in the analysis stage. This helps to model the systems user interface taking a socio-technical approach, important with websites. The main drawback with this approach is concerned with the lack of any explicit recommendations on the identification and analysis of stakeholders and their viewpoints.

2.5 Software Prototyping

This is the process whereby an incomplete model of the fully-featured software program is created, this can be used to let the users have an idea of the completed program and will allow the clients to evaluate the prototype. The process of prototyping involves 4 steps, identifying basic requirements, developing the initial prototype, review and finally revise and enhancements. This approach can be classified in 2 ways

Throwaway or Rapid Prototyping – This particular prototype involves storyboards and mock-ups which are used in the early stages before target hardware and software have been identified. After preliminary requirements gathering is completed, a simple working model of the system is constructed to visually show the users what their requirements may look like when implemented into a finished system. This prototype is then degraded and the final working prototype is developed.

Evolutionary Prototyping – This prototyping method is quite different from the one previously discussed. The main objective is to build a very robust prototype in a structured manner that can constantly be refined. “The reason for this is that the Evolutionary prototype, when built, forms the heart of the new system, and the improvements and further requirements will be built on to it” When developing a system using Evolutionary Prototyping, the system is continually refined and rebuilt. “...evolutionary prototyping acknowledges that we do not understand all the requirements and builds only those that are well understood.

There are many advantages to using prototyping in software development. This approach focuses on reduced time and costs because prototyping improves the quality of requirements and specifications, and early determination of “what the user really wants can result in faster and less expensive software.

CHAPTER 3

Requirements Analysis

3.1 Web Technical Standards

3.1.1 Usability

Usability is an issue that faces every developer designing a system that has a user interface. When considering the problem domain at hand and the web-based solution required by the host company, usability is an undeniable focal point. Nielsen explains that “usability has assumed a much greater importance in the internet economy than it has in the past”. Nielsen continues to elucidate that “as a result of the overwhelming choice and the ease of going elsewhere, web users exhibit a remarkable impatience and insistence on instant gratification. If they can’t figure out how to use a website in a minute or so, they conclude that it won’t be worth their time. And they leave.” Therefore it is of paramount importance to consider this issue when designing the web-based solution; in order to address this issue Nielsen’s Usability Heuristics and Newell ET al’s GOMS technique have been selected.

3.2 User Interfaces

A user interface provides the way a user interacts with the system, ultimately hiding its complexities. Explains that its “focus is communicating to users what the application or web site they are using can do, what their options are and how they can initiate its use”. The design of the user interface affects the amount of effort the user must apply to provide input for the system, interpret the output from the system and how much effort it takes to learn how to do this. The amount of effort ultimately defines how effective the user interface has been designed and overall determines its success.

A well-known web-standards advocate classifies 5 features of interface design; these points will be considered and applied when designing the user interfaces.

Metaphor – Symbolic representation of areas of your site e.g. familiar images as entry points, exits and window in an environment.

Clarity – This demands that every asset on your website should have a reason for being there, and that should be apparent to the user e.g. images and buttons should perform their perceived purpose.

Consistency – The use of metaphors and navigation aids be uniformly used.

Orientation – Users must know exactly where they are at any stage of the website, this is aided by titles, headers and footers etc.

Navigation – Emphasis placed on the importance of good navigation, which can coincide with layout design issues.

3.3 Web Standards

Those who have been in web design and development as long as I have will remember that in the early and mid-1990s Netscape and Microsoft were engaged in the so-called "browser wars." During this time, one browser vendor would implement features that the other browser did not have, which led to websites that did not work properly in both major browsers. For example, you may recall seeing disclaimers indicating which browser was best for viewing a particular site.

The goal of creating web standards, such as the HTML standard, was to eliminate these discrepancies and formalize the de facto standards, enabling you to create sites that worked reasonably well across all browsers. Standards bodies such as the World Wide Web Consortium (W3C) were created as forums to establish agreement across the industry and among vendors. Other standards groups have formed since, most notably the HTML5-focused Web Hypertext Application Technology Working Group (WHATWG). While there is often political maneuvering involved, the goal of each of these standards bodies is to find agreement and thereby formulate standards, often referred to as specifications, across the industry on the technologies that drive the web, including HTML and CSS [5].

3.4 Ease of Maintenance

CMS as “a discipline that manages the timely, accurate, collaborative, iterative and reproducible development of a web property”. Updating website content traditionally required the expertise of a web designer, or someone who has been trained for this specific purpose. Nowadays this is not the case as changes can be made by simply

filling in text fields on a web form, in turn modifying information in a database and these changes are then displayed on the web page. A system that is used for making web content easier to manage is a Web Content management System (WCMS). This system is used to manage and control a large, dynamic collection of web material and provides many key features such as automated templates, easy editable content, scalable feature set, web standard upgrades, and workflow and document management.

A CMS is an easy-to-use tool that will give the company owners the ability to manage the website beyond delivery of the system, and without any technical knowledge or extensive training. However, the purpose of CMS is as a maintenance tool rather than a website creation tool, therefore any new desired functionality will have to be developed by a web designer.

3.5 Technical Analysis

This subsection will evaluate the available technologies relevant to this project. In turn the presentation layer (front-end), client-side technology, server-side technology databases and web servers (back-end layer) have been researched, in order to choose the most appropriate technologies.

3.5.1 Presentation Layer

3.5.1.1 HTML

The official W3C specification recognizes HTML as a SGML (Standard Generalized Markup Language) conforming to the International Standard ISO 8879, and is widely regarded as the standard publishing language of the World Wide Web. SGML has been around since the middle 1980's and has remained quite stable, though much of this stability stems from the fact that the language is both feature-rich and flexible. This flexibility, however, comes at a price, and that price is a level of complexity that has inhibited its adoption in a diversity of environments, including the World Wide Web. Since HTML's inception, there has been rapid invention of new elements for use within HTML as a standard and for adapting HTML to vertical, highly specialized, markets. However, this plethora of new elements has led to interoperability problems for documents across different platforms.

3.5.1.2 XHTML

The official W3C specification for XHTML, establishes it as a family of current and future document types and modules that reproduce, subset, and extend HTML. XHTML family document types are XML based, requiring the document to be well-formed and all designed to work in conjunction with XML-based user agents. XHTML is a reformulation of the HTML and it is intended to be used as a language for content that is both XML-conforming and operates in HTML 4 (latest HTML version) conforming user agents. Developers who migrate their content to XHTML will realize the following benefits:

- XHTML documents are XML conforming. As such, they are readily viewed, edited, and validated with standard XML tools.
- XHTML documents can be written to operate as well or better than they did before in
- Existing HTML conforming user agents.

- XHTML documents can utilize applications (e.g. scripts and applets) that rely upon either the HTML Document Object Model or the XML Document Object Model (DOM).
- As the XHTML family evolves, documents conforming will be more likely to interoperate
- Within and among various XHTML environments.

The XHTML family is the next step in the evolution of the Internet. By migrating to XHTML today, content developers can enter the XML world with all of its attendant benefits, while still remaining confident in their content's backward and future compatibility.

3.5.1.3 Cascading Style Sheets (CSS)

CSS is a style sheet language used to describe the presentation of a document written in a markup language. It's most common application is to style web pages written in HTML and XHTML, but can also be applied to any kind of XML document. CSS is also a way to separate a document's structure from its presentation, and the benefits of this can be quite profound. CSS allows a much richer document appearance than HTML; CSS saves time and aids with maintainability, allowing sweeping changes to be propagated by making quick changes to one file. And document code is reduced in size and complexity, since it does not need to contain any presentational mark-up, increasing the sites performance.

3.5.1.4 Evaluation of Presentation Layer Technology

After the subsequent research was concluded, it became apparent that use of XHTML and CSS, as adopted technology's, would be beneficial to the system. Both technologies fundamentally will aid with the extensibility of the system; by promoting maintainability, browser support and simplicity, as a result of their architecture and general functional purpose.

3.5.2 Client-side & Server-side Technology

3.5.2.1 JavaScript-Client-side

JavaScript is an interpreted programming language with object-oriented capabilities. It is considered the most popular client-side scripting language, and its code can either be embedded in the HTML or referenced locally. The JavaScript syntax resembles C, C++ and Java, therefore my knowledge in Java should equip me well. JavaScript is used most commonly in web browsers, and its scripts allow interaction with the user, control of the web browser and alter document content that appears within the web browser

This language is open source, platform independent, and all script processing is done on the client's browser; reducing server load and overheads. All major browsers support JavaScript and this language is well documented and supported on the internet. JavaScript is a language I have used before for web sites, therefore I appreciate that its main use will be for client-side checks on forms submitted, ensuring the correct information is supplied. As well as for presentation aspects like navigation menus and other interactive features. JavaScript's main limitations is its inability

3.5.2.2 Pre Hypertext Processor (PHP) - Server-side

PHP is currently installed on more than 20 million Internet domains around the world, and is one of the undisputed leaders in web programming languages. PHP is an open-source scripting language for creating dynamic web content, which runs on all major operating systems, from UNIX variants to platforms such as Windows and Mac. It can also be used with leading web servers such as Apache, Microsoft ISS and the Netscape/planet servers. This language is very flexible and can generate any type of document format, with built in support for PDF files, GIF, JPEG, PNG images and flash animation. PHP's most significant feature is its wide-ranging support for all major databases such as MySQL, Oracle, Sybase and ODB Compliant

Databases as well many other obscure databases. In addition PHP provides a library of code to perform common tasks. Although, it is not a general purpose language like

Java, PHP is said to resemble languages such as Perl, Python and C and due to my previous experience with Python, this may aid in adopting this language for development.

Despite PHP many advantages as already mentioned, it also has several drawbacks. Code maintenance is an important issue to consider, because as determined from the interview conducted Section 3.2.3 the scope of the proposed system will see its functionality increase in the future. PHP provides a quick development cycle allowing it to be mixed with HTML code; this short-learning curve appeals to people without much programming knowledge. However, as the website expands without any clear separation the coding can become messy (spaghetti code), and maintenance becomes a nightmare. PHP is not a fully object-orientated language, therefore it lacks some of the capabilities these other languages have. Furthermore, PHP has a problem with stability and interdependences; it provides a great number of extensions but depends on external libraries for database connectivity, therefore, hunting down library versions and ensuring extensions work together can be time consuming

3.5.3 Server-side Considerations

3.5.3.1 Security

The dynamic web content required for such a system, server-side technologies are used to create CGI scripts for this reason. Common Gateway Interface (CGI) scripts are a standardized way for sending information between server and client, and this is a valuable method for providing the systems core dynamic functionality. However, because such scripts are interpreted on the server system, usually utilizing user input values, they are highly subject to exploitation in many ways. CGI-script creation requires many considerations for security, including the following:

- Poorly written scripts may leak information about the server, such as server-configuration details, which can be used for unauthorized access attempts to gain control over the server.
- Data input should always include a default value and character limitations to avoid buffer overflow exploitation.

3.5.3.2 Performance

Typically, a CGI script will read the request, execute any commands, format the result, and then return the result to server. CGI scripts must also share processor cycles with the hatted program, the operating system, and other running applications. A server that is busy executing CGI scripts will have more overhead and demands for processor cycles. Therefore scripts need to be kept as small as possible, as they will be constantly used to perform tasks. Limiting the size of the scripts will ensure less network traffic and load between the server and the client, increasing performance

3.5.4 Web Server Technologies

Essentially a web server is required to receive URL requests from users on the internet, translate them into a filename and then send the desired file (web page) back. However despite what seems a very simple task, different web servers offer different features and types of support. The two main types of web servers that are commonly used are Apache running on UNIX platforms and IIS running on Windows. The main differences between these web servers are their platform dependencies and the technologies it supports, therefore both must be compared in order to assess the most suitable.

Apache is an open source server software designed for sites of all sizes. Open source brings the added benefit of greater reliability, because the programs source code is constantly under scrutiny, therefore issues like security are always being improved. Apache is platform independent running on UNIX and Windows environments, extensible, and can be installed within 20 minutes. Apache offers supports for most languages including PHP, JavaScript as examined earlier, however, it is not compliant with ASP. Database support for Apache includes MySQL amongst others.

3.5.5 Database Technologies

3.5.5.1 MySQL

This DBMS is very well respected, capable of commercial operations like powering Google's search engine, and also appropriate for much smaller businesses. MySQL is released as an open source software and supports most of the functionality you would come to expect from a commercial Relational DBMS. One major area where MySQL falls short is its lack of support for stored procedures and triggers, however these features are due to be fixed for the next major release, MySQL 5.0. MySQL meshes perfectly with technologies such as Perl, Python, PHP and ASP etc., supporting the use of all server platforms from Microsoft to Apache

3.5.5.2 Evaluation of Database Technologies

Due the explosive growth of the Internet, open source database systems like MySQL have become readily available, and the developers of these systems have provided a large subset of functionality that is provided by the expensive commercial database systems. These open source database systems also run on less-expensive commodity hardware and has proven in many cases easier to develop for applications, as oppose to their commercial counterparts

Although Access provides an easier and more familiar interface for non-technical users, it is expected that the proposed web application will not require the user to directly modify the database. As expressed in section 3.5 the features from a CMS will provide useful functionality, therefore an interface like Access should not be required. Furthermore, it is apparent that Access does not scale as well as MySQL, and the resultant performance issue provides a problematic area in the context of the problem domain. Finally, MySQL both fit in line with the company's tight financial budget, because they are open source, platform Independent and readily available

3.5.5.3 Chosen Technologies

After deliberating, the presentation layer will be programmed using XHTML and CSS, due to their embodiment of current and future web standards. In turn promoting better compatibility and supportability across the vast range of web browsers, as well as aiding with maintainability. As far as the lower layers are concerned, it appears that the benefit of Apache being freeware. Their significant popularity, support, extensibility and availability, provides a strong case over its rivals. Therefore, I have chosen Apache and PHP as the server-side technologies, JavaScript as the client-side technology, while incorporating a MySQL database. Another factor taken into consideration when all these technologies were decided, is my previous experience; as the design and implementation phases will have to be completed within a relatively small time frame of 4 months.

3.6 Systems Analysis

This section will use the previous research and requirements gathering conducted, to apply and analyse them more carefully in the context of the problem domain. This section plays an important part of the development life-cycle, as it enables the developer to grasp a better understanding of what the system is required to do.

3.6.1 Why Do Systems Fail?

As a cause of system failure, poor development practices are one of the most significant. This is due to the complex nature of modern software.

Testing of systems already in operation is also important in being prepared for potential system failures. The most obvious example is with the case of the “YK bug”. There can be problems with testing operational systems however, as the example of the Los Angeles emergency preparedness system shows. Due to an error which occurred while testing around four million gallons of raw sewage was dumped into the Sepulveda Dam area [6].

3.6.2 User Requirements

A good set of user requirements are needed for any project, especially computer system projects, to be successful. This is where many projects fail, in that they do not specify correctly what the system should do. In fact many systems have just been given a deadline for delivery, a budget to spend, and a vague notion of what it should do.

Requirements Definition: The truth is that you do not need a great deal of technical knowledge to specify requirements; in fact it can be a big disadvantage. A requirement for a computer system specifies what you want or desire from a system. For a business in particular this is.

This advantage need not just be a reduction in costs, in fact many systems justified on a reduction in operating costs, fail to deliver as low skilled but relatively cheap staff, have to be replaced by high skilled, and more expensive staff. The advantage can be a reduction in time to process something, which will lead to a reduction in costs, or being able to better use the unique knowledge base belonging to a business.

- **Functional requirements** - are what you want a system to do.
- **Non-functional requirements** - are restrictions on the types of solutions that will meet the functional requirements.
- **Design objectives** - Are the guides to use in selecting a solution [7].

3.6.3 Feasibility Study

This section ultimately determines the viability of the proposed system. The following study is conducted before the design stage commences, and reflects on the research and requirements analysis from the previous sections. The results from this study aid with analyzing and understanding any potential problems, and is used to make the final decision on whether to proceed with the project. This study will be presented in five sections; Technical, Economical, Legal, Operational and Schedule.

3.6.4 Technical

This first area of study assesses whether the chosen technology meets the needs of the proposed system. This issue is fundamental to the success of the project, without appropriately chosen technology the developed system will not be able to fulfil the company's requirements. This issue needs to be considered before development commences, otherwise changing technology at an advanced stage will delay the project and could result in more serious repercussions. As far as the presentation layer is concerned, the use of XHTML and CSS will be employed due

To their compliance with today's web standards. Both technologies used appropriately provide better browser compatibility, which is essential otherwise if your site does not support all web browsers then the company is automatically losing a number of potential visitors and customers. The choice of PHP and MySQL database in terms of open source servers-side technologies,

Security is clearly an issue when data is inputted and outputted via the system. Such systems that operate online are more prone to the risk of being hacked or attacked. Therefore procedures need to be put in place to secure certain information. In response to this, the chosen technologies can be used to implement a login system that requests a valid user name and password; thus protecting data from unauthorized users.

3.6.5 Economical

The part of the feasibility report gauges whether the proposed system is affordable, and any incurred expenses justifiable. It is important that the system is cost and benefit effective; otherwise as Heathcoat. Explains, "If the benefits do not outweigh the costs, then it is not worth going ahead with". When applying a cost-benefit analysis on the proposed system, it is evident that any business starting up should have a web site by mandatory.

CHAPTER 4

Design

4.1 Database

A selection of modelling techniques was used to determine the structure of the database and the subsequent structured data. Table schemas have been created to outline the necessary data fields and data types, and in turn aid with fulfilling requirement. An Entity Relational Diagram will then be used to represent the overall structure of the database and Normalization is used to ensure that the structure of the database is accurate, by ensuring that there is no data duplication.

4.1.1 Table Schemas

A main feature of the system is displaying model profiles, such details need to be stored and accessed from a database. Therefore, it is essential that the tables and entities are created correctly.

* The primary keys in the schemas above are in italics and the foreign keys underlined.

BPCAmember (*username* varchar(6), password varchar(6))

ProfilesMen (*man_id* int auto, *username* varchar(6), name varchar(25), height numeric(3,2), chest numeric(2), waist numeric(3,1), weight numeric(3,1), hips numeric(3,1), shoes numeric(3,1), hairColour varchar(12), hairLength varchar(8), eyes varchar(8), sex char(1), password varchar(6))

image_location (*image_id* int auto, image_path char(255), *username* varchar(6))

4.1.2 Entity Relational Diagram

The structure of the stored data as previously shown in the table schemas, coupled with its constraints, can be designed using Entity Relational modelling (ERM). ERM is a graphical representation of entities and their relationships to each other. Each entity in the database has a unique name, definition and type, and these must be chosen carefully.

Choosing suitable names is Important, especially for SQL querying. In addition, each entity should have a unique identifier (primary key), which is vital when a query needs to access information from more than one table. Relationships between entities can be modelled in three ways: one-to-one (1:1); one-to-many (1: M); or many-to-many (M:M).

The following figure 4.1 shows Entity Relational Diagram

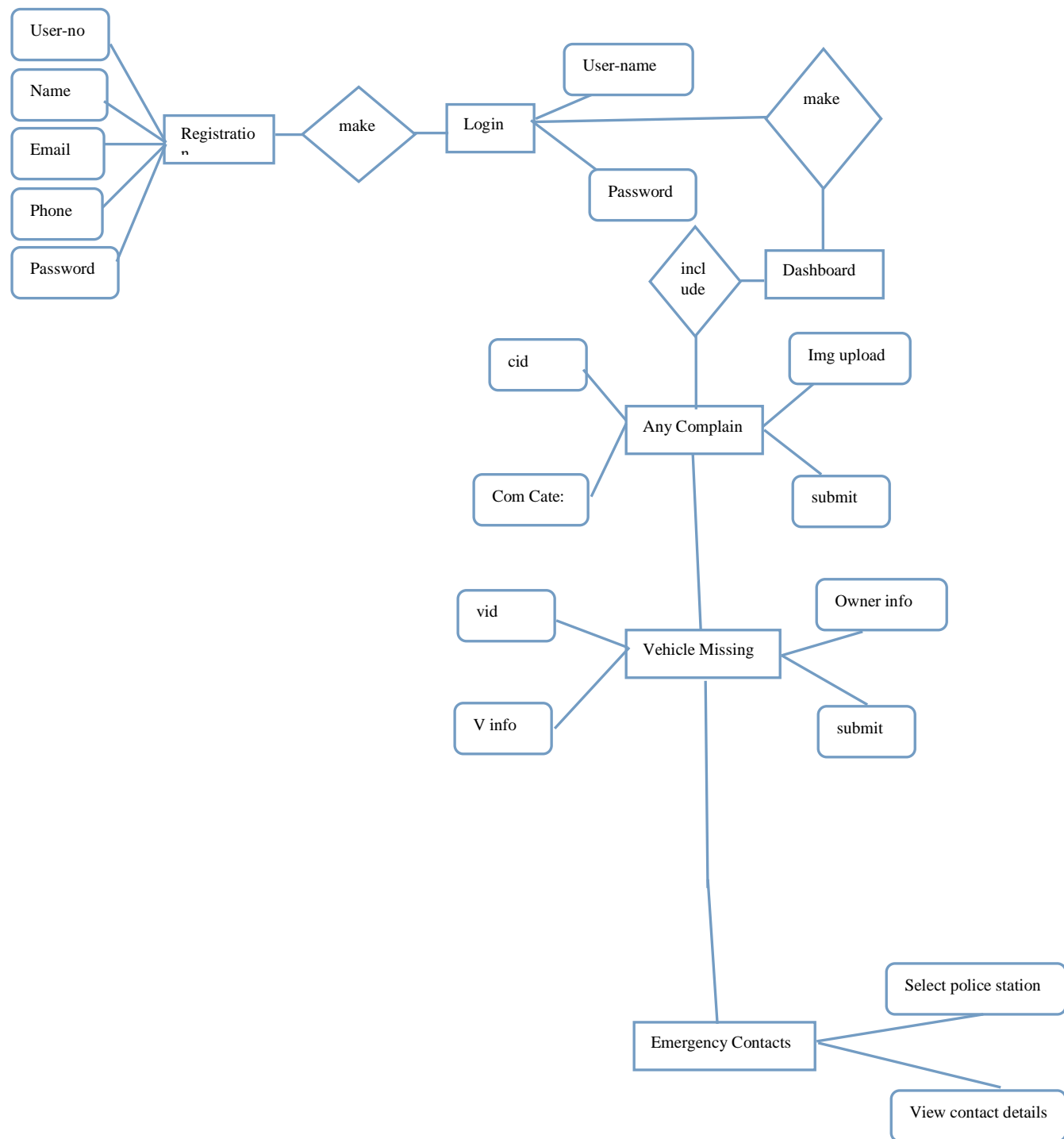


Figure 4.1: Entity Relational Diagram.

4.1.3 Integrity Constraints

The previous ER diagram shows how the entities (tables) in the database relate to each other, utilizing primary and foreign keys to form these relationships; as expressed in the database schema. Every table (record) in the database maintains a primary key, which is used to uniquely identify that record to any other record in the database. The use of a primary key is essential for preserving the integrity of the information stored in the database, because it is quite common for table fields to share the same value e.g. 'Name'. The primary keys in each table are assigned a numerical identification, which is generated by using 'auto number' to ensure that no record has the same ID. To form relationships between two tables, the joining table must contain the primary key of the other table, which is known as the foreign key. The use of such keys enforces entity integrity and the way references can be made to link information together is an essential part of database normalization.

4.1.4 Normalization

This is a design technique by which relational database tables are structured in such a way as to make them less vulnerable to certain types of logical inconsistencies and anomalies. This is a vital design consideration, because for instance if the field 'username' was changed in bpcamembers, then username would have to be changed manually for every time it occurs in other tables in the database. If the update was not completed consistently throughout the database, then this would result in redundant data. Normalization will also allow the database to be more scalable, improving the speed of queries and subsequent data retrieval. Third normal form (3NF) is a type of normalization which ensures that a new relation is created for each functional dependency found in the database, this technique will be applied to the database in question.

4.2 Server-side Considerations

4.2.1 Server-side Security

Server-side security in relation to CGI-scripts is a main concern with such techniques. There are numerous tools which can be used to scan code for typical vulnerabilities. However, for the specific use of python for writing CGI-scripts RATS auditing tool will be used during the implementation phase to test the code for such security flaws, to eliminate buffer overruns and SQL injections etc.

4.2.2 Performance & Maintainability

When designing the CGI-scripts to perform the desired functionality, there are maintainability and performance issues to consider. Therefore in order to resolve these issues the script file sizes must be kept as small as possible, this can be achieved by having a main python function file that all the other CGI scripts access. This distinct separation also aids with maintainability as the smaller CGI files are more manageable, and any added functions can be stored in one place, and not repeated.

4.3 Presentation Layer

4.3.1 Web Interface

An important aspect to consider is setting the dimensions of the web page. A common error made in web design is providing an interface that is too big to fit in most user screens. This results in the user having to waste time scrolling the page in order to perform tasks, which may become tedious and off putting. Ultimately it is impossible to choose a screen resolution that fits all, therefore common sense dictates that the page size should accommodate the largest possible number of users. According to The Counter 51% of surveyed users view pages at monitor Resolution 800x600, with 35% using 1024x768.

The design is very straight forward, easy for users to understand and yet will provide a modern simplistic look. This design will be incorporated for every page found

throughout the web site, supporting consistency and orientation the navigation menu is situated underneath the banner and will remain there throughout the website; formatted to support the color and text scheme. Supply the template used for all the other web pages with the ‘main content’ as the only aspect that will change.

The following figure 4.2 shows User Home Page

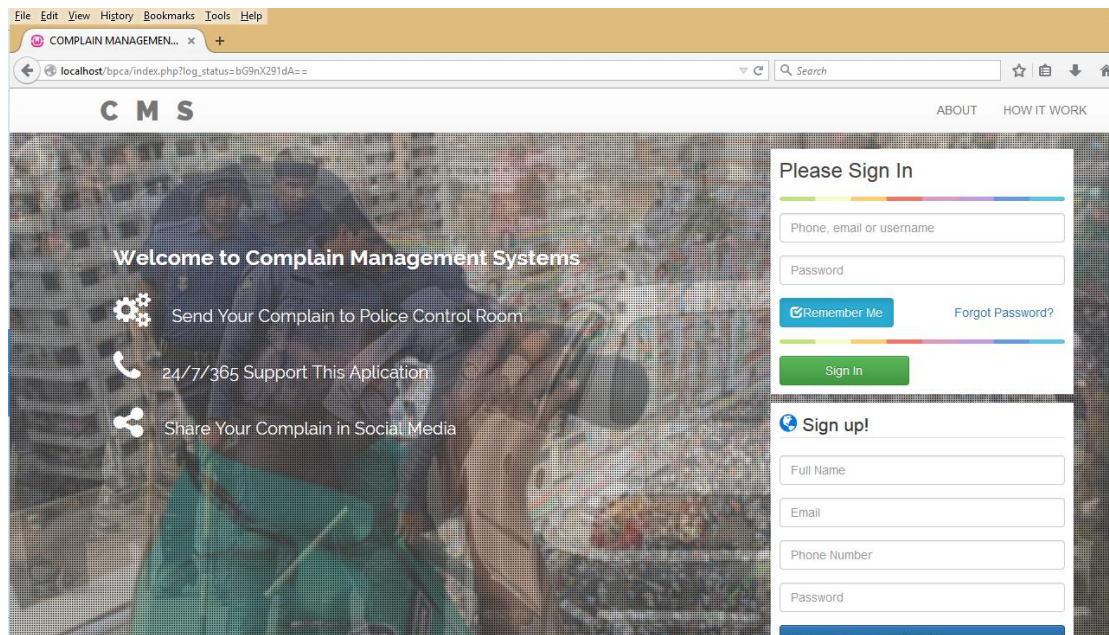


Figure 4.2: User Home Page

4.3.2 Navigation

To achieve this, the sites navigation must be clear, consistent and the text links should always be appropriate labelled and concise. The main navigation menu should also avoid dynamic menus created using technologies such as flash and JavaScript etc., because they pose supportability and compatibility problems with older browsers.

The following figure 4.3 shows User Main Menu



Figure 4.3: User Main Menu

4.3.3 Colors

The color scheme used throughout a web site is very important and has to be carefully chosen green for fertility, nature and earth; and blue for truth, nature and peace etc. high contrast is a very important design issue i.e. using light colors against dark colors, for maximum distinction between the layers of content.

In following with a consistent design, the color scheme chosen will be used throughout the site and will be determined by the color scheme incorporated in the banner design, taking into consideration different color shades and levels of contrast. Therefore the hexadecimal values are represented as red (#FF0000), grey (#DEDEF0), black (#000000) and white (#FFFFFF); and will form the schema for the web site.

4.4 System use-case diagram

A use-case diagram was created to detail which use-cases were in the scope of the initial prototype. It was decided that the prototype would implement basic ‘view route’, ‘create route’, ‘edit route’ and ‘search/filter routes’ use cases.

The following figure 4.4 shows: use-case diagram.

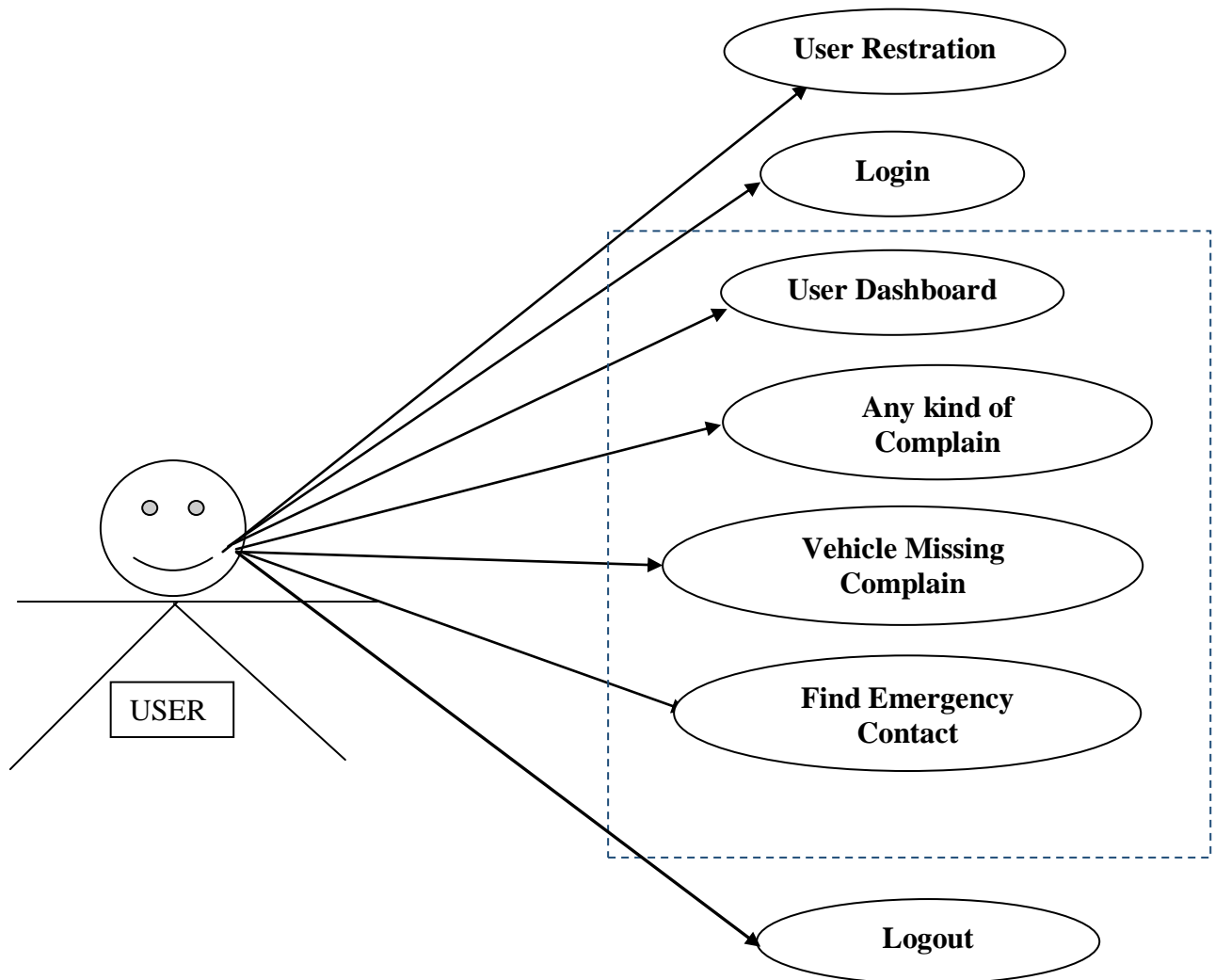


Figure 4.4.: use-case diagram.

4.4.1 Secure Area

Concerns the implementation of the third requirement. The main design issue considered here is the ability for privileged users to access the secured area, namely the owners, so they can add new model clients to their database; supporting system maintenance. Access is provided by entering a username and password in the login screen on the home page once access is gained, a link will be provided to another page containing a form with text fields that can be edited with client information. Once the form is completed, the data can be directly inserted into the database by pressing the 'submit' button, providing the information entered does not jeopardize the database's integrity. Integrity checks will validate the text fields on the client and server-side before they are submitted to the database.

CHAPTER 5

Implementation

5.1 Database

The table schemas as proposed in the design section were mostly implemented, maintaining the chosen tables and data types. However, the original use of the 'image_location' table to reference the locations of the model images, for retrieval and presentation in the 'photo gallery' was dropped. This was due to technical problems which will be evaluated in further detail the database was populated using test data, which was more than adequate for setting the system up.

5.2 Server-side Considerations

5.2.1 Security

Security risks are evident when using CGI-scripts to process information between the client and server. Such security checks are paramount, especially as a login system will be implemented providing access to a secure area, which will contain personal information and other features

5.2.2 Performance and Maintainability

Determine methods for resolving performance and maintainability issues. These suggestions have been implemented into the system, by using a main function file called BPCA. Which contains all the functions required for the CGI files to operate.

5.3.1 Layout

The layout of the interface was constructed using divisions, which is the preferred XHTML choice to using tables in HTML. A graphical representation of how these divisions are organized can be seen in more detail in appendix N, where code snippets will be supplied to show how they were implemented.

5.3.2 Navigation

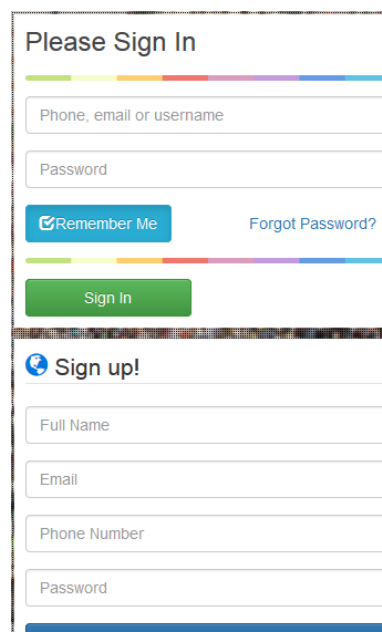
The navigation menu was constructed using divisions and each menu link is clearly separated. When the user hovers their mouse cursor over the link the area changes to a darker shade of red. Also white text was used on a red background to provide a sufficient level of contrast the navigation bar, and the use of CSS enables the developer to reuse the same formatting style by simply calling the id reference tag, found in the CSS file. In addition, the navigation bar is structured so hovering over certain links activates a drop down menu with more options.

5.4 Secure Area

5.4.1 Login

The secure area used by members is accessed via the login box on the homepage. In order to access the secure area, a user must acquire a valid username and password supplied by BPCA, upon confirming their application to be a model. This feature was implemented using a PHP CGI-script which queries the database using SQL. If the information supplied is incorrect then the script redirects the user to another page, informing the user about their failure to login

The following figure 5.1 shows Login Area.



The image shows a web form for user authentication. It is divided into two main sections: 'Please Sign In' and 'Sign up!'. The 'Please Sign In' section includes a text input for 'Phone, email or username', a password input, a 'Remember Me' checkbox, a 'Forgot Password?' link, and a green 'Sign In' button. The 'Sign up!' section includes a globe icon, a 'Full Name' input, an 'Email' input, a 'Phone Number' input, and a 'Password' input. The form is styled with a white background and a blue border.

Figure 5.1: Login Area

5.4.2 User Area/ Dashboard

The following figure 5.2 shows User Dashboard.

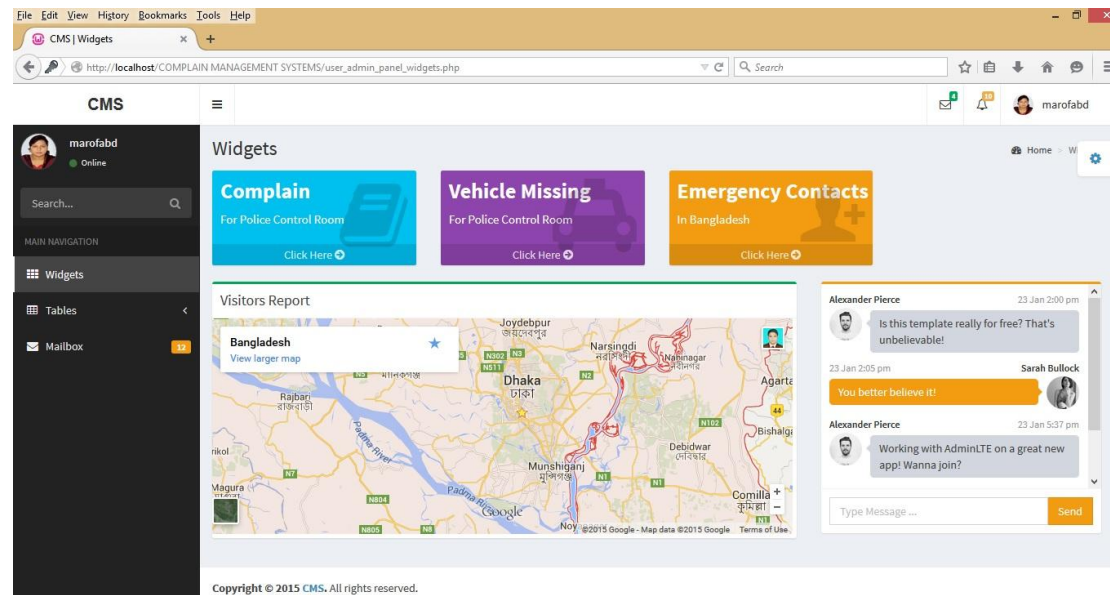
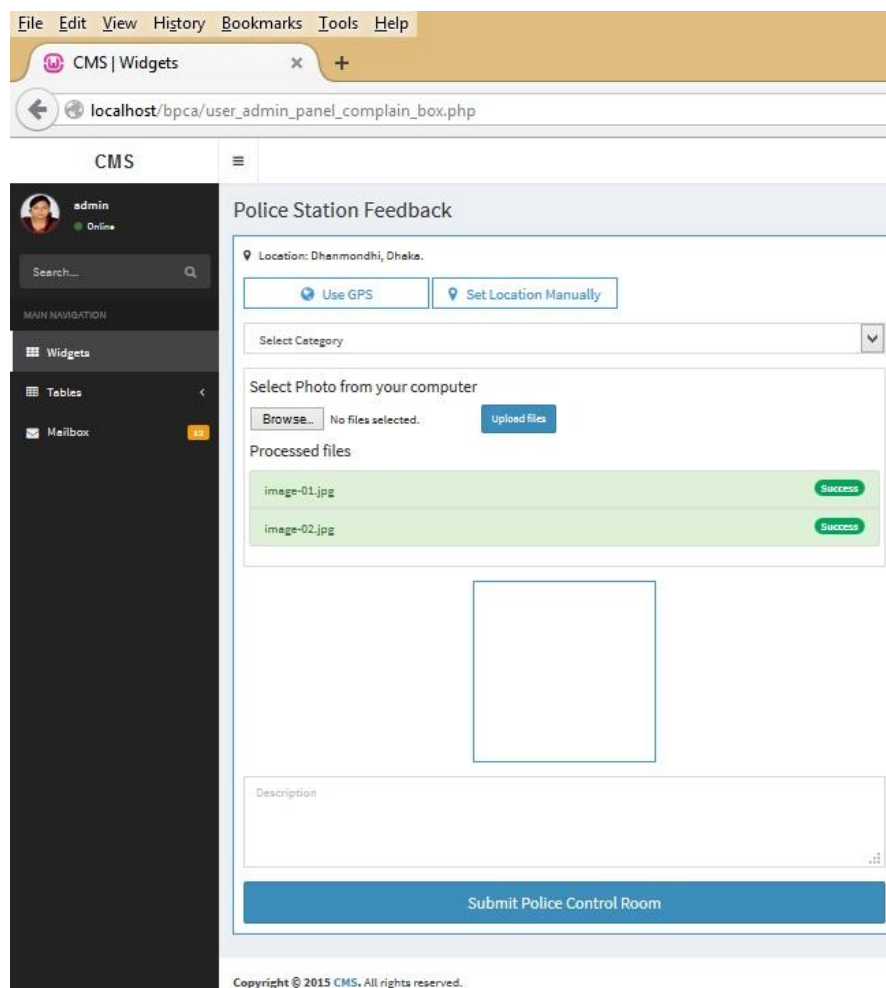


Figure 5.2: User Dashboard

5.4.3 Complain For Harassment

Another additional feature in this software which will be able to receive *Any Kinds of Complaints* happening in roadside or anywhere else around the country. For example, a woman or a girl can instantly complain while being harassed or being teased on road. The procedure is as same as the first one. Victim has to choose the option from the dashboard.

The following figure 5.3 shows Complaints for Harassment.



The screenshot shows a web browser window with the address bar displaying 'localhost/bpca/user_admin_panel_complain_box.php'. The page is titled 'Police Station Feedback'. On the left, there is a dark sidebar with a user profile for 'admin' (Online) and a main navigation menu with options: Widgets, Tables, and Mailbox. The main content area contains the following fields and buttons:

- Location: Dhanmondhi, Dhaka.
- Buttons: Use GPS, Set Location Manually
- Select Category (dropdown menu)
- Select Photo from your computer section with a 'Browse...' button and 'No files selected.' text.
- Processed files list showing 'image-01.jpg' and 'image-02.jpg', each with a 'Success' status.
- A large empty rectangular box for a photo or video.
- Description (text input field)
- Submit Police Control Room (blue button)

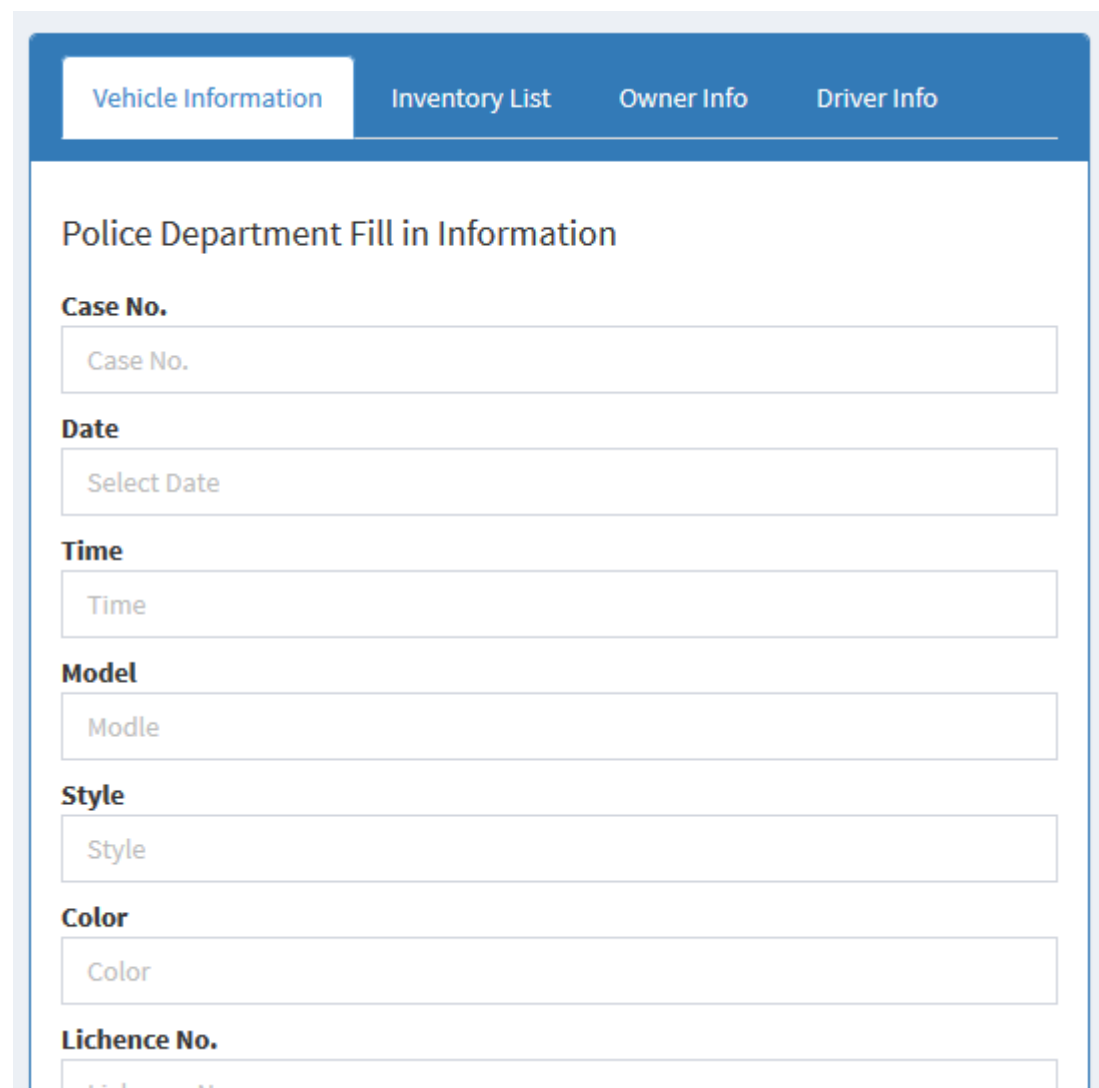
At the bottom, a copyright notice reads: 'Copyright © 2015 CMS. All rights reserved.'

Figure 5.3: Complaints for Harassment

5.4.4 Vehicle Missing Complain Box

Vehicle Missing Complains will help people find out their missing vehicles which might got stolen unfortunately. For that, user will have to put their user name and password. After that, a dashboard will be opened up. There will be options which he/she needs to choose and can make their complains over there by writing the cars number, detail and place of being stolen.

The following figure 5.4 shows Vehicle Missing Complains box.



The image shows a web application interface for reporting a missing vehicle. At the top, there is a blue navigation bar with four tabs: "Vehicle Information" (which is selected and highlighted), "Inventory List", "Owner Info", and "Driver Info". Below the navigation bar, the main content area is titled "Police Department Fill in Information". This section contains several input fields, each with a label and a placeholder text: "Case No." with placeholder "Case No.", "Date" with placeholder "Select Date", "Time" with placeholder "Time", "Model" with placeholder "Modle", "Style" with placeholder "Style", "Color" with placeholder "Color", and "Lichence No." with placeholder "Lichence No.". The form is designed with a clean, modern look using a light blue and white color scheme.

Figure 5.4: Vehicle Missing Complains box

CHAPTER 6

Testing

6.1 Usability Inspection

A usability inspection is an evaluation based on general design principles or a specific list of guidelines, which is usually carried out by the developer. Nielsen's heuristic evaluation for interface design used to develop the system, will be reintroduced as a checklist to ensure they have been successfully met. However this method of testing is exclusively for the developer, therefore unit testing requiring user interaction will be implemented to ensure the developer has not over seen anything. Overall the usability inspection returned positive results, as determined by Nielsen's heuristic evaluation. All ten heuristics were successfully met as determined by the checklist evaluation. However, it was evident that 'user control and freedom' as part of the checklist was not fully supported. It was determined that in the secure area of the site, certain links returning the user to where they came from, was not full evident.

6.2 Unit Testing

This method of testing involves a thorough examination of the units that make up the prototype, ultimately evaluating the usability and efficiency of the systems interface. The method employed for this testing is the KLM framework, a simplified adaptation. This technique is effective for determining how well the user task routines have been implemented, considering the cognitive and physical processes; with the aim of determining the unit testing carried out for this project, tested the routines required for accomplishing tasks set by the developer. In order to test all the functionality the developer supplied a user with a list of tasks, and filled in a table recording the results. Such tasks included navigating to different areas of the site and interacting with the system e.g. adding clients to the database.

6.3 Browser Testing

Therefore it was important that the web site supported as many browsers as possible, because many ecommerce sites lose business due to a lack of support from all browsers, displaying their site in the way it was not developed for. In order to achieve a high level of browser compatibility and presentation consistency, methods recommended by the W3C such as use of XHTML, CSS and JavaScript were implemented. Furthermore, to test how browser compatible the system was, the site was viewed in several browsers e.g. Microsoft Internet Explorer, Firefox, Mozilla, and Opera etc. shows a good level consistency across the different browsers; overall the layout, navigation bar and images and etc. transferred well across the different platforms, with the exception of a few text fields which drifted slightly.

CHAPTER 7

Conclusion

7.1 Conclusion & Future Work

Today is the age of modern science and Information technology, which is critical to the development of more effective operational and management process. To serve customers well, companies need to be proficient in many areas, reduce cycle times, faster development of new business strategies improved customers services, and increase knowledge sharing and learning. IT is a critical but most powerful resource for accomplishing all the goals. Right now, we are initiating this software as web-based. In near future, we aim to make it as a mobile app-based software- which will become much easier for people to respond quickly on the spot.

7.2 Achievement

During our project work, we have gone through with immense learning of software planning, Coding, Designing web page, software testing and many more. In addition, we learnt how to run through a server. So, we have successfully achieved the concluding result of our project.

7.3 Limitations

- In Bangladesh, GPS tracking is not possible in web-based software.
- Lack of internet availability everywhere.
- General people's does not know-how use on technology.

APPENDIX

The figure shows UML Class Diagram.

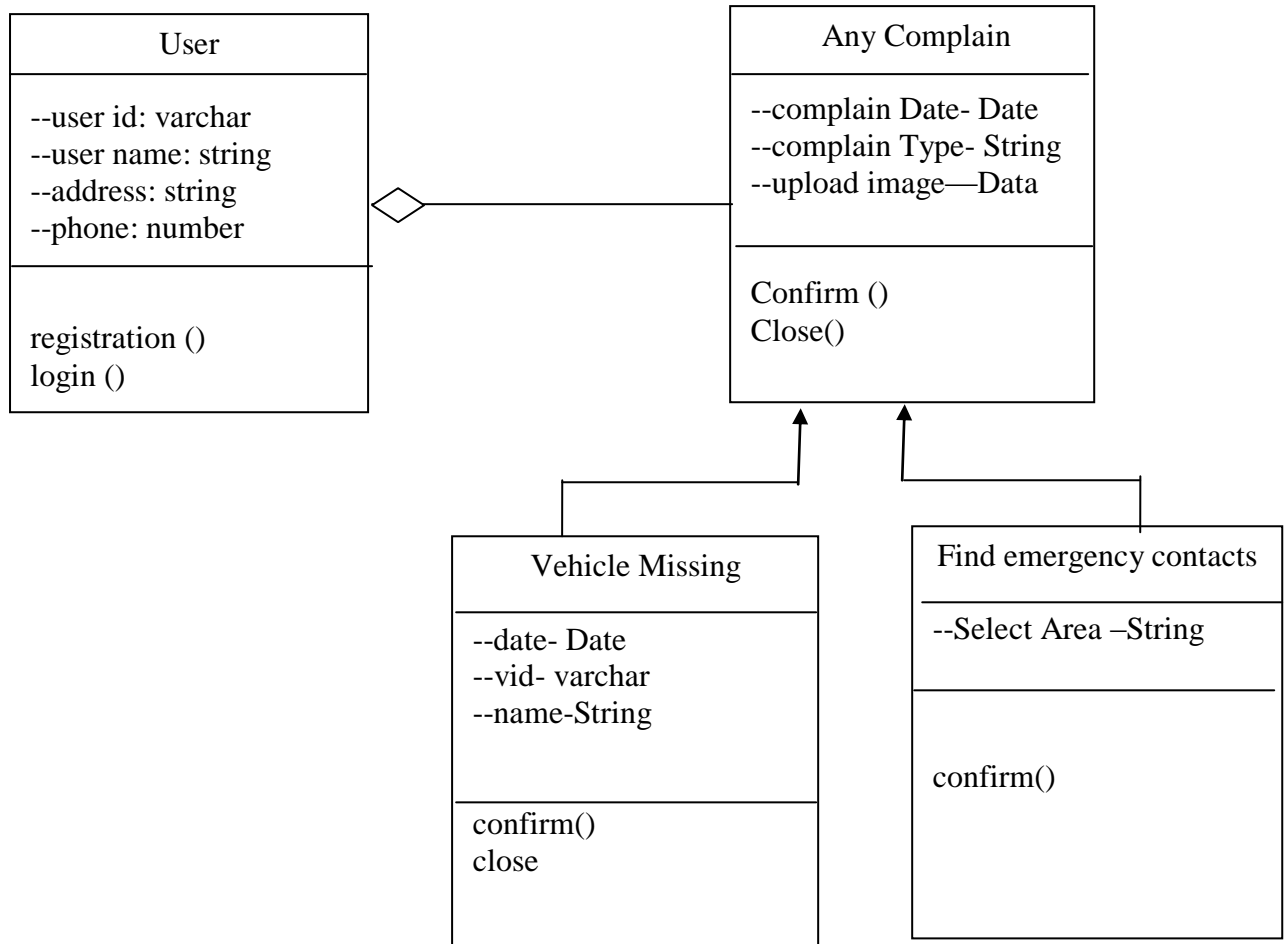


Figure: UML Class Diagram.

References

- [1] Methodology, <http://www.comp.leeds.ac.uk/fyproj/previous-titles/bsc2006.html>, Last Accessed on: 04.08.2015
- [2] The Waterfall Model, <http://www.comp.leeds.ac.uk/fyproj/previous-titles/bsc2006.html>, Last Accessed on: 04.08.2015
- [3] Development Methodology Diagrams, <https://www.google.com.bd/search?q=methodology+diagram&biw=1366&bih=634&tbm=isch&tbo=u&source=univ&sa=X&sqi=2&ved=0CBsQsARqFQoTCMeEya-9oMcCFcEajgod41UKMw#imgsrc=-JX7q24VITINEM%3A>, Last Accessed on: 8.8.2015
- [4] Structured System Analysis and Design Method, <http://www.comp.leeds.ac.uk/fyproj/previous-titles/bsc2006.html>, Last Accessed on: 4.8.2015
- [5] Web standards, <http://www.adobe.com/inspire/2012/04/what-are-web-standards.html>, Last Accessed on: 04.08.2015
- [6] Why do system fail?, <http://www.benmeadowcroft.com/reports/systemfailure/> , Last Accessed on: 05.08.2015
- [7] User Requirements, <http://www.coleyconsulting.co.uk/require.htm>, Last Accessed on: 05.08.2015