A Transport Management System for National Savings Bank

B A S WIMALASENA
2014MIT074
Supervisor: Mr. G.K.A. Dias

A dissertation submitted in partial fulfillment of the requirements for the degree of Master of Information Technology

31st May, 2019

Abstract

In this project, transport Management System was successfully developed for the transport division of the National Savings Bank. The system reduces most of the inefficiencies at this department. For example, assign of drivers and vehicles for transport request is facilitated by this system and is thus easier for the department heads.

As a benefit of this system, management can keep track of all vehicles, drivers, employees, customers and transport schedule details. The customer can be able to make reservations, cancelations and track the vehicle's location. Any changes to the schedule request due to unforeseen circumstances such as accidents, bad weather, driver unavailability is easily done through the system and is automatically communicated to the relevant patients through SMS messages. Since each passenger is given a specific start time, they could come shortly before that time without waiting much time. Once they start the tour, they are put on a monitoring screen to be monitored by the department. Each driver will be getting a SMS message to let the next passenger's details in the FIFO queue and destinations needs to meet. If it is a package delivery, then SMS message contain delivery note number.

The system was implemented using eclipse and MySQL and has been successfully tested and implement by the client.

Declaration

This dissertation is my original work and has not been submitted previously for a degree at this or any other university/institute. To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Student's name: B. A. S. Wimalasena

Signature:

Date: 31st May,2019

Acknowledgements

I would like this opportunity to express my deep sense of gratitude and profound feeling of admiration to my supervisor Mr. Kapila Dias of University of Colombo School of Computing (UCSC), who guided me throughout the project.

The knowledge that I obtained in the lectures conducted at UCSC throughout 2 years became great foundation for me to develop this project. I would like to thank all my UCSC lectures and my friends who helped me in develop and implementing the system.

Special thanks to my client – The Manager, transport department of the National Savings Bank, who trusted me and motivated me towards building a complete solution for them.

Finally, I would like to offer my gratitude to my parents, wife and daughter for encouragement and assistance given to me.

Table of Contents

List of Figures

List of Tables

Abbreviations

NSB	National Savings Bank
ICT	Information and communications technology
MIS	Management Information System
SDLC	Systems Development Life Cycle
OOSE	Object-Oriented Software Engineering
OOPL	Object Oriented Programming Languages
OOP	Object Oriented Programming
JVM	Java Virtual Machine
VMD	Vehicle Management Database
PDL	Program Design Language
UCID	User Centered Interaction Design
RAD	Rapid Application Development
TMD	Transport Management Database

Chapter 1: Introduction

1.1. Client's Background

National Savings Bank is the premier Licensed Specialized Bank in Sri Lanka owned by the Govt. of Sri Lanka. The bank offers various banking solutions to public and has covered the whole island with a network of 255 branches, including 12 postal banking units and 06 loan centers. Head office of the bank located in Colombo and it comprise of several support divisions to do the day to day business.

The client of this project is transport division of the bank who is the facilitator and mostly involves day to day transport activities in the bank. All divisions and near Colombo area branches of the bank can obtain the service directly from the transport division which is in head office while take the required permission from the nominated officers in charged. Outer Colombo areas branches can get the services from the Island wide located regional transport offices while obtain the same permission. The staff of transport division are consisting of a manager, assistant manager, executive officers, staff assistants, drivers, cleaners and mechanics. There are near 80 vehicles in the bank managing by the division which falls into certain categories such as Bank higher management vehicles, Cars, Vans, Lorries and modified marketing vehicles.

1.2. Motivation for the Project

Currently transport division have a manual system to do vehicle reservation, cancellation, collect feedbacks from divisions and branches. They use logbooks, running charts and paper requisition form for manage above tasks. The current system keeps each record of vehicle's running cost, service maintenances, accident repairs, insurances, revenue licenses and emission reports in their logs books. They need to go through each logbook to retrieve data of vehicle details, running reports, periodical renewal deadlines and other details.

There is no driver performance recording system maintained in the current system and no records of driver license details too. With current system there is no information provide to review the completed job and performance evaluation of the drivers.

The following problems have been identified in the existing manual system in the transport division.

- There is no systematic method for booking, cancelling and collect feedbacks from the customers for both divisions and branches.
- No real time vehicle tracking system.
- There is no centralized system to manage booking request to regional office.
- No proper mechanism to store and retrieve vehicles details such as revenue licenses, insurances, emission test reports.
- There is no proper mechanism to store and retrieve drivers details such as permitted license type, number of accidents faced and total driven distance.
- No reminding mechanism to alarm the deadline of revenue licensing, Insuring and emission test.
- No systematic process to calculate running cost, maintainers & other expenses.
- No MIS reports can obtain.
- Inefficient and too much paper works.

By implementing a system, the organization can handle their day today work effectively and efficiently through the automated system. They can organize their work systematically avoiding duplication of work and unnecessary work and procedures.

1.3. Objective

The ultimate objective of this project is to propose an ICT based solution to the client. To meet that following main output have considered.

- Provide a system of transport service booking with an option of authority to approve and decline the request, cancelling and collect feedback.
- Provide a facility to calculate cost and expenses.
- Provide a facility to verify the real time vehicle location.
- Provide a system to both customers and staff with minimum paper works.
- Provide a system to keep records such as drivers, vehicles, pending and completed transport jobs, running expenses.
- Provide a centralized system to the division which allow managing resources effectively and efficiently both head office and regional branches.

- Improve the efficiency of the daily routine processes such as vehicle booking process, driver allocation, vehicle license insurance emission test, generate daily reports.
- Provide MIS data to the management to make effective and efficient decision.

1.4. Assumptions

An assumption is a belief of what have assume to be true in the future. Assumptions are supposed to be true but do not certainly end up being true, however, they may turn out to be false, which can affect the project significantly. They add risks to the project because they may or may not be true. Assumptions play an essential role throughout developing and implementation. Therefore, it is good to collect and identify as many assumptions as can.

Few assumptions have made based on the knowledge, experience or the information available for this project. These are anticipated events or circumstances that are expected to occur during project's life cycle.

- All relevant stakeholders will be given fully support and needful thoughts from onwards the requirement gathering task.
- The transport division will be providing all the resources required by at the implementation of this system.

Chapter 2: Background

2.1. Analysis

Analysis phase is the most significant part of the software development life cycle. It is necessary to get a clear idea, overall big picture about the problem domain and what are the user's requirements and what looks like the final system before developing it.

Requirements gathering is an essential activity in any kind of system developing. Since that is the most difficult, considerable amount of time must be spent for this process. Completely identified requirements lead to accurate decisions which make the analysis phase a successive.

The new system should be benefited to the users of the bank. Analysis would involve,

- Collecting data about the current system / proposed system.
- Find out problems with the current system.
- Establish the problem that the customer needs solving.
- Identify inputs, processing and outputs of the current system / proposed system.
- Identify requirements of the new system.
- Producing a cost benefit analysis.

2.1.1. Requirement Gathering and Fact Finding

Requirement gathering in this stage can be done using various fact-finding techniques, following techniques have been used for the gathering requirements.

2.1.1.1. Interviews

Interviews allow me to get an instant feedback and clarify doubts occurred from scenario-based requirement gathering immediately. Further, during the interview nonverbal communication can be observed and interpreted. Also, users are not restricted for specific structure they can freely express their ideas and indirect observation also can be done during the interview. Interviews done with some selected users including Transport division manager, officers and drivers of the bank.

2.1.1.2. Questionnaires

A questionnaire allows getting a quick response from users of the system and results can be tabulated quickly, also easy to analyst. As many employees busy with their own work and some employees such as drivers not good with that, simple questions are given and asked them to click right.

2.1.1.3. Report Inspection

This involves looking at paperwork for the current system. Refer the documents such as division vehicle inspection reports, forms and the reports that they have prepared manually to the management.

Various transaction reports such as maintenance details and fuel cost, vehicle allocation and vehicle request forms also carefully inspected. This become easy as there was very less user involvement to this, however some employees help me to clarify some unclear documents.

2.1.1.4. Observation

Observation is a technique where activities are analyzed by watching person perform activities. This used to identify certain parts that were missed by users during they were describing on those activities.

Massage passing between other divisions and the Transport division to work as support service in the Bank, were observed for further analysis.

Informal observation was done to avoid uncomfortableness from being watch and to avoid performing odd things.

2.2. Existing system

Currently transport division have no computerized system and they use manual system to do vehicle reservation, cancellation, collect feedbacks from divisions and branches. For that they use logbooks, running charts and paper requisition form for manage above tasks. The current manual system keeps each record of Vehicle’s running cost, service maintenances, accident repairs, insurance, revenue license and emission in their logs books. When they create reports they need to go through each logbook to retrieve data of vehicle details, running reports, periodical renewal deadlines and other details.

Furthermore, there is no driver performance recording system maintained in the current system and no records of driver license details too. With current system there is no information provide to review the completed job and performance evaluation of the drivers.

2.3. Design strategies

Before start to implement a system, get a proper idea of the problem domain in the existing system is must. The client has many functions relating to the system each are interrelated. Each of these information systems has a purpose or focus, and each has a life of its own. This “life of its own” concept is called the Systems Development Life Cycle - SDLC. SDLC includes the entire process of planning, designing, developing, testing, deploying and maintaining an information system.

Agile SDLC model is a combination of iterative and incremental model. Small and highly motivated software teams are involved. There should be continues communication between user and the development team. The most important part is delivered first. According to their importance the product is delivered to the user. Delivery is done frequently. Agile methods break the processes into small tasks and present them to the user. The user will do the user acceptance test. If the user requirements are satisfied, then continue. Otherwise the tasks must be repeated. While user involves to the system throughout the SDLC the constant changes could be avoid. Changes are inevitable during the software development.

Business people and developers must work together during the entire life of the project. Meetings must be handled frequently with the development team. Then conveying information with in the development team and they can exchange their views. This process will motivate the development team.

System design is the process of defining the components, modules, interfaces and data for a system to satisfy specified requirements. System development is a process of creating or altering systems along with the processes, practices, models and methodologies.

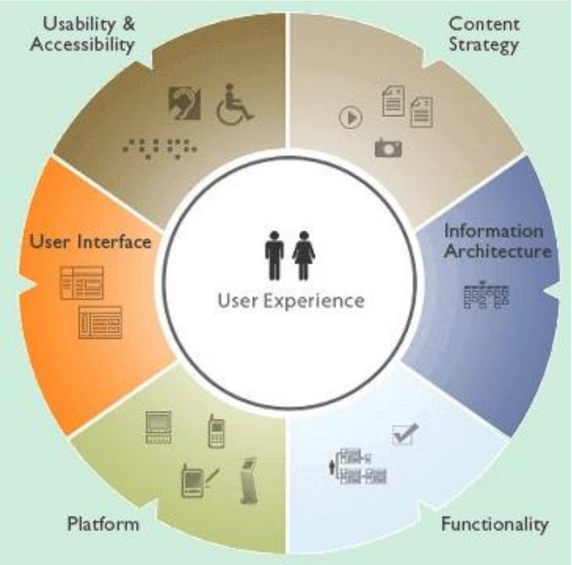

Figure 1: 5 Design Strategies [1]

2.4. System Requirements

The whole system is designed with six separate subsystems. They are Vehicle registration system, Driver registration system, Driver management system, Vehicle management and tracking system, Movement management system and Report generation system. Illustration of the system's intended functions, its surroundings and the relationship between them are provided using use case diagrams. Main use cases represent the functionality provided by the system. Administration, Drivers, System Operator and manager are the main actors of the company.

2.4.1. Functional Requirements

As a solution for the above identified requirements an online web application for the bank with portals to other divisions and transport division to interact is proposed. The online web application will contain following main modules and functionalities.

2.4.1.1. Administrative Module

This module enables the system administrator to add/update/delete records in the database.

- Vehicle Registration – Administrator has the privilege to add/update/delete vehicle details in the system.
- Employee Registration – Employees details including drivers and other internal employees' details are recorded in the system.
- Internal User Management – Administrator assign user authorities for the internal users.

2.4.1.2. Internal Operational Module

Through this module the clerk can perform the following functionalities.

- Customer Management – Customer divisions details are entered to the system by the internal operator.
- Driver profile management – Some Jobs are allocated by the clerk.
- Managing vehicle profiles – Fuel management, licensing and insuring the vehicles.
- Inquire/Manage vehicle bookings – View details of the vehicle reservations.
- Track vehicle locations on the map – Enables identify the location of the vehicle.

2.4.1.3. Internal Management Module

Manager is assigned with to this module.

- Manage Vehicle Rates - Hiring rates and fees are entered though this module.
- Manage offers – Seasonal offers details are introduced.
- MIS Report Generation – Required reports are generated by the manager. Appendix C includes all the system generated reports.

2.4.1.4. Customer Divisions Module

All the divisions in the bank are registered as customers of the system. They can do the following functionalities using the system.

- Manage service reservation/cancelation
- Check history of trips made
- Manage notification – SMS/email alerts are sent to the officers in charge.
- Track currently allocated vehicle's location and status

2.4.1.5. Service Module

These are the other services provides by the vehicle management and transportation system.

- Hire Service
- Check vehicle's and driver's profiles
- Make/cancel reservations
- Give ability to track vehicle in the time of driver is using the vehicle
- View available vehicle to hire and their details

2.4.2. Non-Functional Requirements

2.4.2.1. Security

There should be some security levels for the users of the site. Some users have administrative privileges, some has limited access, some can only input data & modify data and internal operators only can view data. Session management will handle the main security levels.

- Basic Level Security - The users have access to particular pages that can only view the details in the pages except insert or updating features.
- Middle Level Security - The middle level users are permitted to insert and modify details in the database but do not have the administrative authority.
- Administrative Level Security -The administrator of the system can access/insert/modify or delete the data from the database.

2.4.2.2. Reliability

The system should be reliable when it comes the making reservations, which is very important module to the customer divisions. It is also necessary that the updating of the database which is done through the system very accurately.

2.4.2.3. Accuracy

Report generation, data input should be accurate in the system. These reports are given to Account division and management for making future decisions.

2.5. The 3-Tier Architecture of the Transport System

A 3-tier architecture is used for the transport System which is composed of three "tiers" or "layers" of logical computing. This architecture often used in applications as a specific type of client-server system. 3-tier architectures provide many benefits for production and development environments by modularizing the user interface, business logic, and data storage layers. Use of this architecture have given greater flexibility to development teams by allowing them to update a specific part of an application independently of the other parts. This added flexibility can improve overall time-to-market and decrease development cycle times by giving development teams the ability to replace or upgrade independent tiers without affecting the other parts of the system.

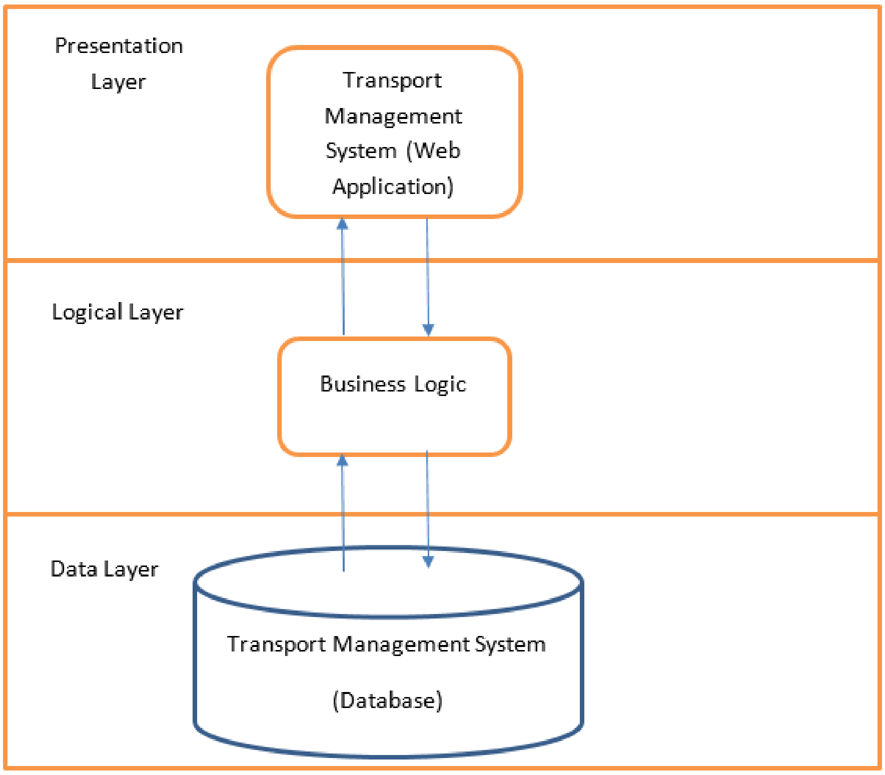
Presentation Layer
Transport Management System (Web Application)
Logical Layer
Business Logic
Data Layer
Transport Management System
(Database)

Figure 2: Illustrates the 3-tier architecture of transport system

2.6. Use Case Diagram

Following use case scenario description refers in ‘Figure 3’ helps to identify the actors and Transport management system processes. The clerk can view information and update details. Manager has the privilege for MIS reports. Customer divisions can make booking though online website of system. Administrator has full authority of the system. Drivers get alerts of the allocation of the vehicles.

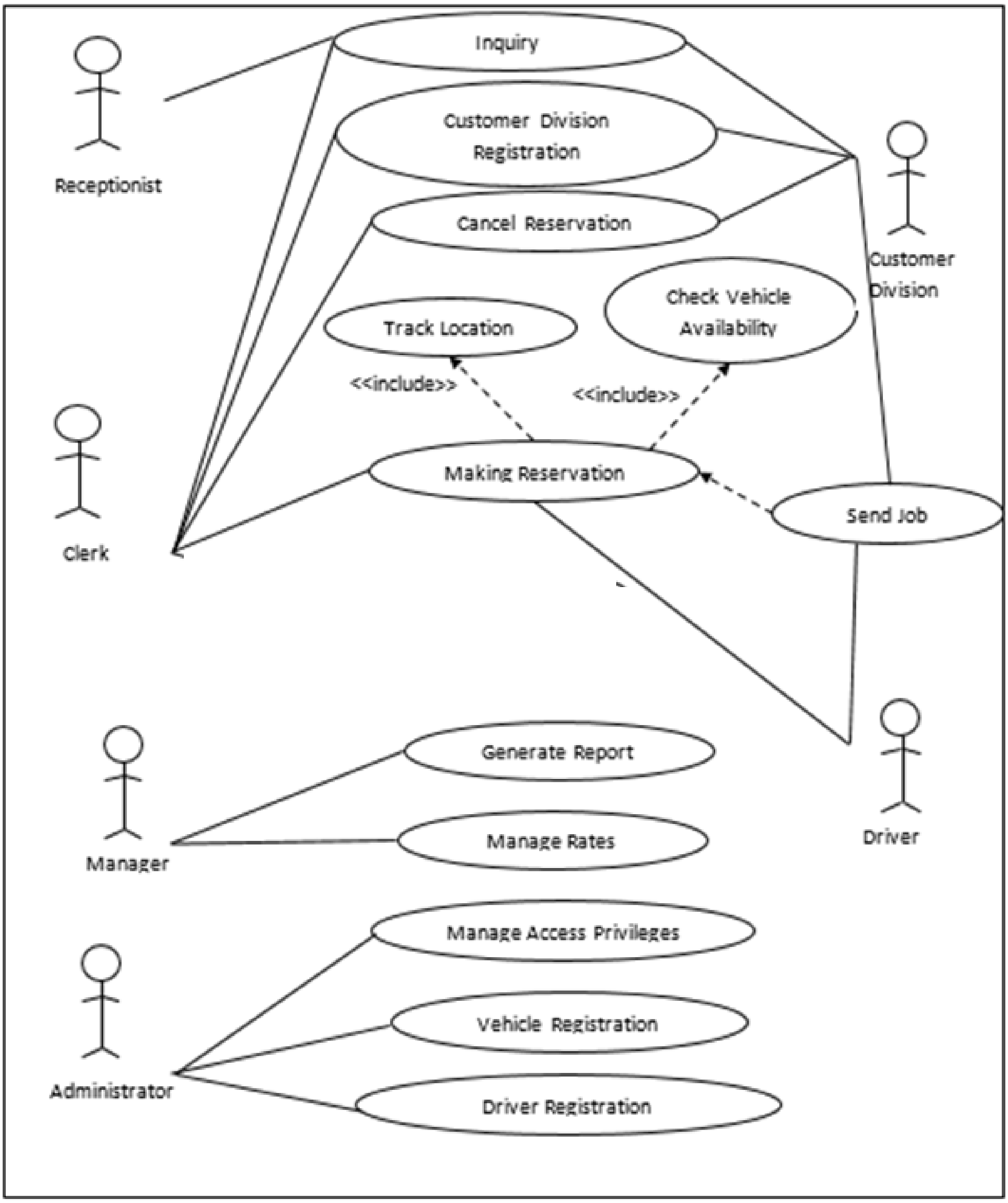

Figure 3: Use Case Diagram

2.7. Use Case Narratives

This shows the use case narratives for Vehicle Registration use case.

Table 1: Use Case Narrative - Vehicle Registration

Use Case Name:	Vehicle Registration	Use-Case Type – Business Requirement
Use Case ID:	VM – 01	
Priority:	High	
Source:	Document	
Primary Business Actor:	Administrator	
Other Participating Actors:		
Other Interested Stakeholders:		
Description:	At present bank's own vehicles are used for transportation. Required vehicle information captured to the system.	
Preconditions:	Logged in user should have admin privileges.	
Trigger:	This use case is initiated when the administrator selects the option to Register Vehicles.	
Typical Course of events	**Actor Action**	**System Response**
	Step 1: Administrator entered vehicle details to the system.	**Step 2**: System verifies the information and added to the database.
		Step 3: Vehicle Registration number is set as the primary key.
	Step 4: Send registration information to the printing Job.	**Step 5:** System formatted the registration information and print registered vehicle report.
Alternate Courses	Vehicle Registration number has set as the Primary Key.	
Post conditions	Vehicle was successfully registered, and system registration information issued.	

Important use case narratives are attached Appendix A.

2.8. Similar systems

There are different type of vehicle transport and Movement management systems around the world. Most of the popular systems were in United States, European countries with East Asian countries by vehicle transport agencies.

Most of the software solutions are developed for general purposes. Some systems have features with limitations. That makes the customer confuse and dissatisfy. This chapter describes existing similar systems with advantages and disadvantages.

2.8.1. Truckers Helper Overview

Truckers Helper is a fleet and trucking management solution for small and medium businesses. It offers such services as billing and invoicing, dispatch management, mileage tracking, and others. The software was designed and launched by Truckers Helper Online headquartered in Florida, United States. [2]

The software is a viable solution for small and medium sized businesses. It can be used in such major industries and domains as retail and wholesale, transport and delivery services, third party logistics, automation and manufacturing, and others. Some notable clients of the software include J&B Broughton Trucking, Overnight Carriers, S&S Trucking, and others.

2.8.2. Dossier Fleet Maintenance Overview

Dossier Fleet Maintenance Software is a fleet and asset management software solution for your business. The software was released by Arsenault Associated headquartered in New Jersey, United States. Various fleet maintenance operations and PM Service schedules are automated by the software, in addition to other services. [3]

The software offers extensive features and is powerful enough to meet the needs of all business sizes. If your company needs an advanced fleet maintenance system that can regulate all the aspects related to equipment, scheduling, and other aspects, Dossier Fleet Maintenance can provide that to you.

Swift implementation of the software makes it a viable solution for you if you want to save time. If your business is in expanding stages, the software will still work for you since it is flexible enough to expand with the evolving business needs.

The software is deployed in a wide range of industries dealing in heavy equipment. Noteworthy clients of the software include Potomac Disposal, Inc., Grand Prairie ISD, Acadian Ambulance, and Krapf Bus Companies.

2.8.3. BrokerPro Overview

BrokerPro is a comprehensive Transportation Management Software solution designed to make life easier for medium to large transportation intermediaries dealing in TL and LTL freight. With BrokerPro, dispatchers work from a single screen where they can easily build loads, assign carriers; enter shipment details, fax or email rate confirmations and post loads to load boards. [4]

2.8.4. A comparison of the most popular Transport Management systems

Comparison of Truckers Helper, Dossier Fleet Maintenance and BrokerPro systems are listed in the table 2:A

Table 2: A comparison of the most popular patient scheduling systems

Features	**Truckers Helper**	**Dossier Fleet Maintenance**	**BrokerPro**
Size of the Company	Small and Medium businesses.	All business sizes.	Medium to large businesses.
Software Modules	An Accounting module which manages income and expenses. Customer and driver records are maintained including safety records and auto-reminders.	The software focuses on efficient equipment and asset management. Include modules for estimating, accounting and management.	Seamless bi-directional integration with QuickBooks for Accounting. Automatically generate customer invoices and carrier settlements. Tools to generate and modify reports detailing active & open orders, assigned & pending loads, aging summaries, sales & load history and performance ratings.
Tracking Vehicles	Offers comprehensive truck maintenance, trailer maintenance, and inventory management services.	Any type of equipment, no matter how heavy, can be tracked with the software and this includes buses, tractors, trailers etc.	Truckload capability lets you split loads, enter multiple customers, stops, consignees, and rate information.
Facilities	The software also offers logbook to keep track of logs and verify driver rule compliance.	Low learning curve and quick implementation of the software allows for minimum training duration of employees, thus saving money and time.	Quickly fax, print or email customer and carrier rate confirmations from one screen. Automatically send status emails as check-calls are received.
Efficiency	One-click data entry system is not available.	Efficiency is also improved with the special one-click data entry system.	A single ``click`` pushes your loads to DAT, Internet Truckstop and Post Everywhere.

Other important capabilities	Billing and invoicing, payroll management, trailer tracking, dispatch management, routing and scheduling, and others	Billing & invoicing, dispatch interface, driver compliance, fuel optimization, IFTA reporting, maintenance scheduling Provide optimum route for transportation. This is made possible by the special route optimization feature.	Safer Watch Safer Watch gives you the ability to simply and quickly qualify, import, and manage carriers.

2.8.5. Review of the Systems

In the above systems, they design the software solutions for common purposes in vehicle transport industry. They have well organized web based automated vehicle transport management system with spare parts order management system. Concept is good but the requirement of NSB Transport division is slightly different than these solutions. Especially they need both back end and front-end solutions for vehicle transportation management.

Chapter 3: Methodology

Current developments are mainly done using Object Oriented Programming Languages (OOPL). Because of its re-usability, easy maintenance and better support from case tools are few that Object Oriented Methodology is necessary. OOSE is an object-oriented system design methodology with all these advantages, therefore move forward with OOSE as we discussed in previous chapter. With this methodology and concerning application as web application we have few popular programming language options to go forward; those are Java, PHP and HTML.

Apart from that, the main advantages of Object-Oriented Programming (OOP), besides being able to create more complex software, and develop answers without changing the questions, are numerous. OOP facilitates creating reusable code, which can save a lot of work. Information can be hidden to safeguard data from improper access. *Polymorphism* lets the programmer to create multiple definitions for operators and functions, with the programming context determining which definition is used and *Inheritance* lets the programmers derive new classes from older ones. When using OOP, the programmer can focus on representing concepts, instead of concentrating on tasks.

When it comes to Java it is a very well defined OOPL, which can be run in any machine that has the Java Virtual Machine (JVM) and Java is an open source language which has real object-oriented concepts in use. As this is a web project can use Java NetBeans IDE to develop well structured, secure web applications.

During the implementation process Eclipse Java EE – Kepler IDE used to build the code and web interface which were the outcome of the design process. Database which is designed using the class diagram was created using MySQL software.

3.1. Implementation model of OOSE

After deriving the database design from the class diagram, implementation of the database is done using MySQL. The database is password protected and database administrator has ability to restrict its usages as desire.

This project uses MVC (Model, View, Controller) Architecture to structure the system. Model contains all the logic and interactions with database level and these modules are used collaboratively between many web pages through the controllers to enhance reusability.

‘*VehicleCommon*’ java file and ‘*VehicleServlet*’ java file contains coding to act as the main model and main controllerin the vehicle transport and managements system. Other *‘jsp files’*contain the coding regarding to view the system. Main JSP file of the system is ‘*index.jsp’*.

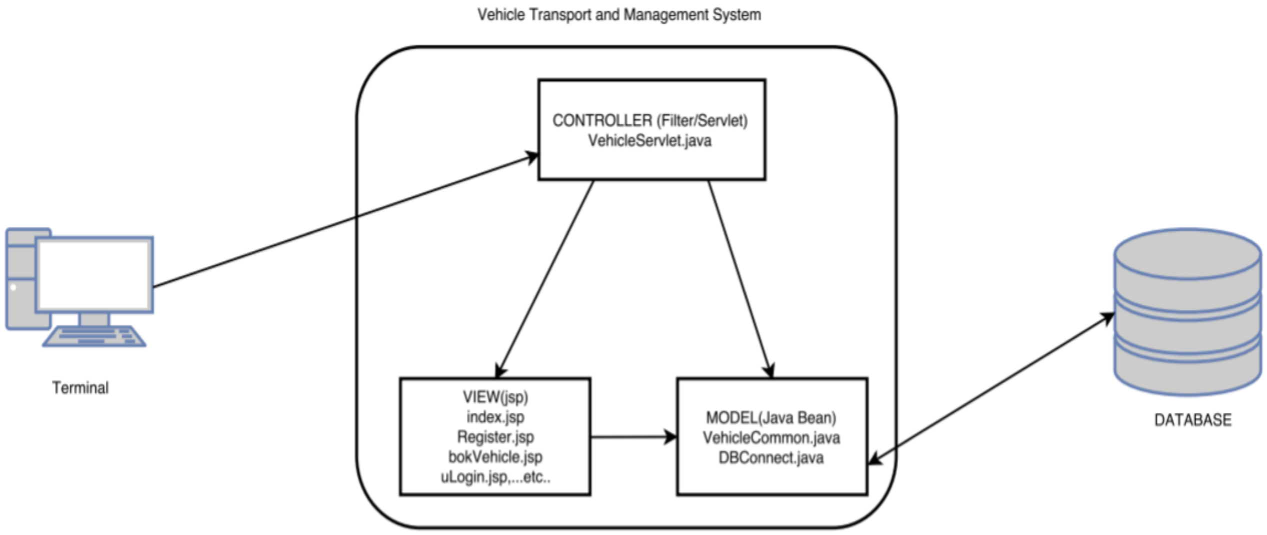

Figure 4: Module Structure of the System

3.2. Database Design

Database design is an important place in designing a system. During this phase care should be taken to avoid redundancy of information storing into a database, since it leads to wastage of memory space.

The attributes of the classes are used in database design and normalized tables are taken into the implementation of the database. Transport Management Database (TMD) is designed using MySQL database design software. Technical details of few main tables are further elaborate in Appendix A.

Table 3: DB table structure for table admin_user

Column	Type	Null	Description
emp_id	varchar(10)	No	Employee ID
username	varchar(30)	No	User Name
userpass	varchar(30)	No	Password
user_catg	varchar(10)	No	User Category
user_add	date	No	User added date to system
user_chg	date	Yes	User details change date

The table ‘admin_user’ consists with the users of the staff-members who have privileges to logging to the system. Their logging information is saved in the logged_admin_users table.

Table 4 : DB table structure for table logged_admin_users

Column	Type	Null	Description
emp_id	int(10)	No	Employee ID
uname	varchar(30)	No	User Name
login	Date/Time	No	Login Details
logout	Date/Time	Yes	Logout Details

3.3. Class Diagram

This represents the static view of the system refers in 'Figure 5', as it shows interaction between classes in the system. Also, classes are the blueprint for objects. Ultimately the attributes of the classes are used to create database design and normalized tables are taken in to final implementation of database.

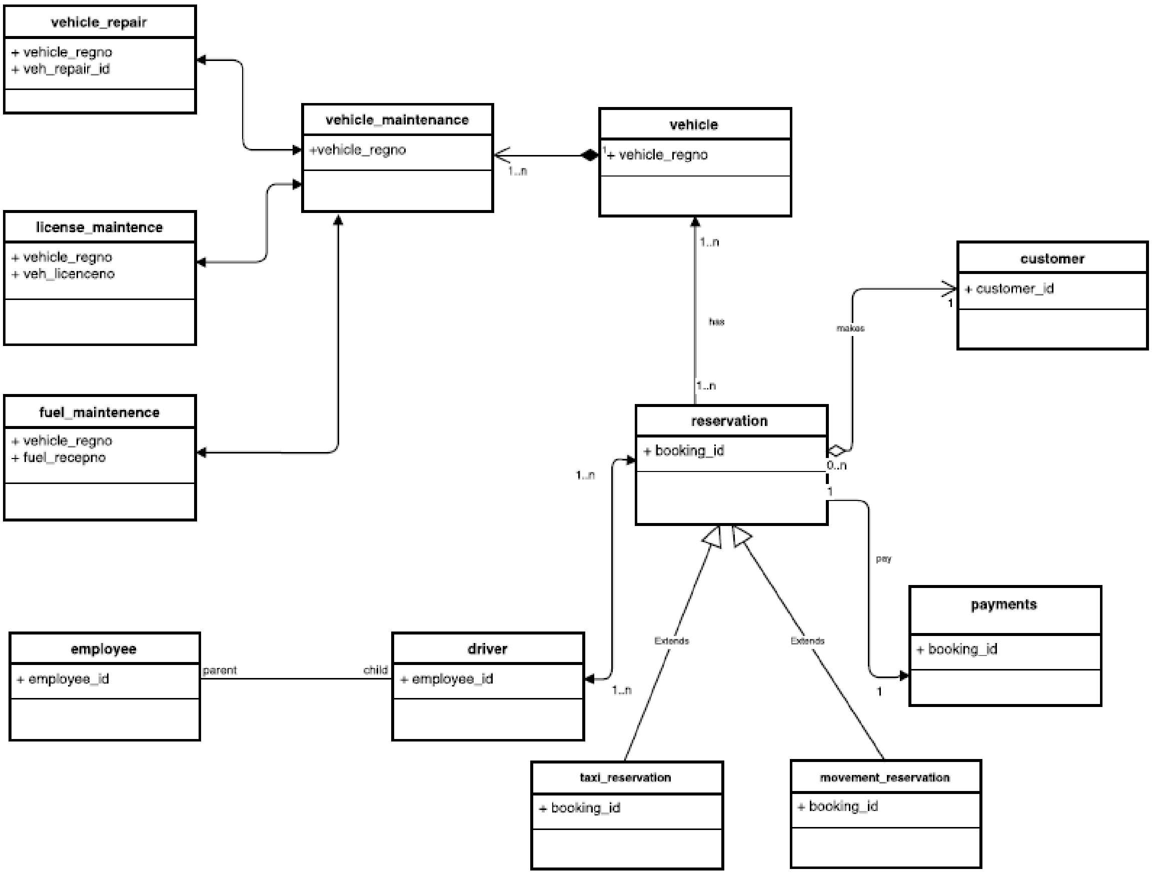

Figure 5: Class diagram of the system

3.4. Sequence Diagram

Sequence diagram shows how objects communicate by passing messages. Sequence diagrams consist of blocks, which are directly used for program design. Program Design Language (PDL) was used for block designs. Block designs are later combined to create the necessary methods in the classes. User Registration sequence diagram is shown below. Some important diagrams are available in Appendix A.

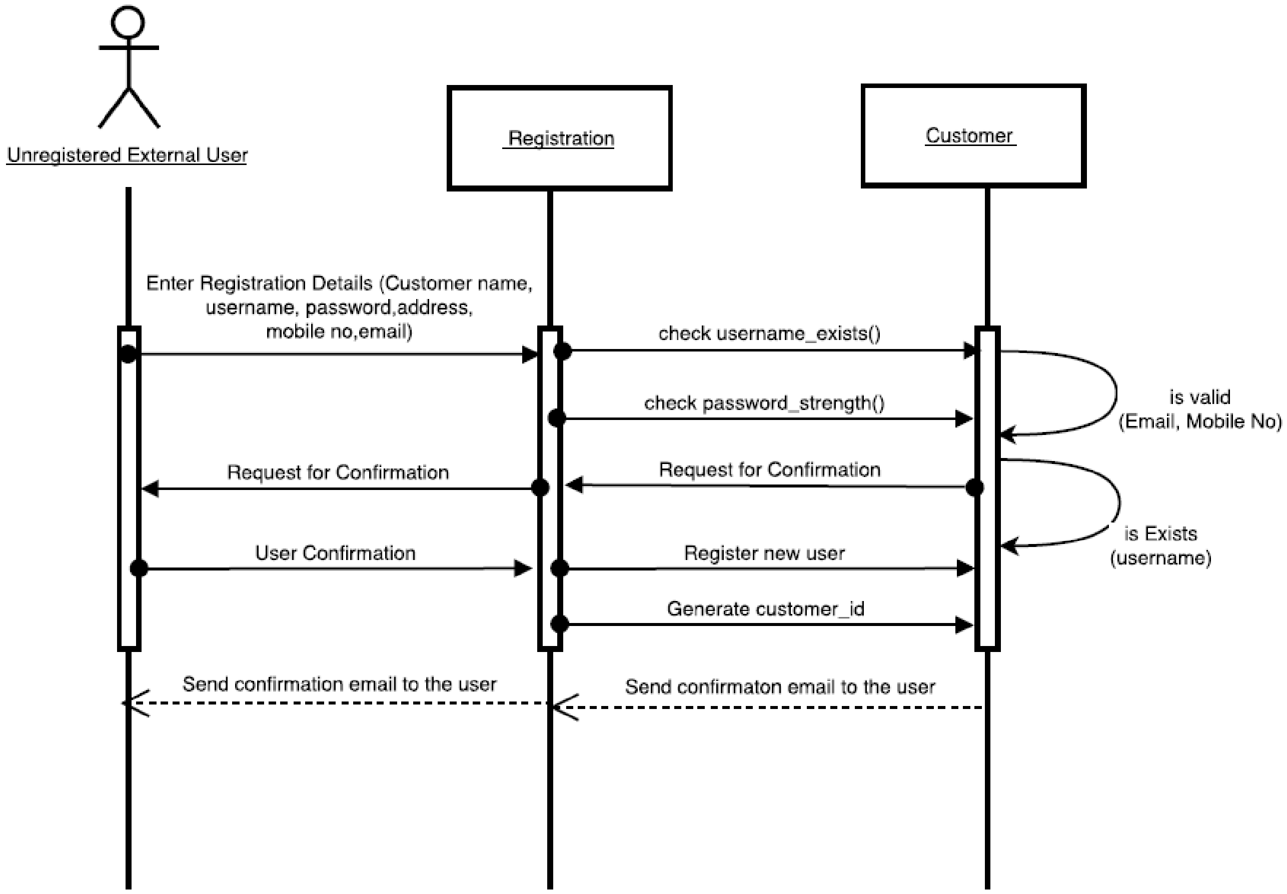
Unregistered External User
Registration
Customer
Enter Registration Details (Customer name, username, password,address, mobile no,email)
check username_exists()
check password_strength()
is valid (Email, Mobile No)
Request for Confirmation
Request for Confirmation
is Exists (username)
User Confirmation
Register new user
Generate customer_id
Send confirmation email to the user
Send confirmaton email to the user

Figure 6: User Registration Sequence Diagram

3.5. Database Implementation

The free and open source relational database management system My SQL Workbench has used in this project to implement the TMD.

‘*DBConnect*’java class file consist of coding which can be reused to connect to the TMD.

3.6. Frontend Implantation

For the front-end implementation of web-based user interfaces used HTML and JavaScript as JQuery Framework with Bootstrap to implement better dynamic responsive web pages design in NetBeans IDE 8.2

3.7. User Interface Design

To design the user interface web pages the concept of Logical User Centred Interaction Design (LUCID) Methodology by Dr Charles Kreitzberg is used to provide user the best user friendliness from the system. [5]

As the prototype used in requirement gathering it was easy to gather the user interface requirements. The screens were design by HTML and Java Scripts.

Since most of the internal users are interact with the system full day, it is necessary for going on light colour, but components need to be easily viewable to them. There it is decided to use substance themes for this user interface designs. After the implementation of the interface, the training and help through the system is done.

But for the external users, interfaces are designed with more attractive features and combining bright colours.

Java script files were used to design the users' interface. Cascading style sheets with Bootstrap features were used to improve the graphical view the user interface parameters such as forms, buttons, menus, text boxes, fonts.

Chapter 4: Implementation

4.1 Introduction

During the implementation process Eclipse Java EE used to build the code and web interface which were the outcome of the design process. Database which is designed using the class diagram wad created using MySQL software.

4.2 Technological Overview of the Implementation

4.2.1. Implementation Language

Current developments are mainly done using Object Oriented Programming Languages. Because of its re-usability, easy maintenance, better support from case tools, etc. are few that Object Oriented Methodology in necessary. Object Oriented Software Engineering (OOSE) is an object-oriented system design methodology with all these advantages, therefore I move forward with OOSE as we discussed in previous chapter. With this methodology and concerning application as web application we have few popular programming language options to go forward those are Java, PHP and HTML.

Apart from that, the main advantages of OOP, besides being able to create more complex software, and develop answers without changing the questions, are numerous. OOP facilitates creating reusable code, which can save a lot of work. Information can be hidden to safeguard data from improper access. Polymorphism lets the programmer to create multiple definitions for operators and functions, with the programming context determining which definition is used, also Inheritance lets the programmer derive new classes from older ones. When using OOP, the programmer can focus on representing concepts, instead of concentrating on tasks.

When it comes to Java it is a very well-defined object-oriented programming language, which can be run to any machine that has the Java Virtual Machine (JVM) and Java is an open source language which has real object-oriented concepts in use. As this is a web project, we can use Java NetBeans IDE to develop well structured, secure web applications.

4.2.2 Implementation Database

The free and open source relational database management system MySQL with the GUI tool phpMyAdmin was used in this project to implement the vehicle management database.

‘DBConnect’java class file consist of consist of coding which can be reused to connect to the VMD database.

```
package net.vehicle;
import java.sql.*;
import java.io.IOException;
import java.util.*;

public class DBConnect {
    public static Connection prepareConn() {
        //System.out.println ("In prepareConn()…..");
        Connection con=null;
        try {
        Class.forName("com.mysql.jdbc.Driver");
    con=DriverManager.getConnection("jdbc:mysql://localhost:3306/vdb","root","");
        }
        catch (Exception ex) {
            System.out.println ("Exception in preapreConn()…"+ex);
        }
        return con;
    }//prepareConn
    public static void closeConn(Connection con) throws SQLException,IOException {
        //System.out.println ("In closeConn()…..");
        if(con!=null)
            con.close();
    }//closeConn
    public static int updateQuery(String sql) {
        int update=0;
        try{
            Connection con=prepareConn();
            Statement stmt=con.createStatement();
            update=stmt.executeUpdate(sql);
            closeConn(con);
        }catch(Exception e){
            System.out.println ("Exception in updateQuery()..."+e);
        }
        return update;
        }//updateQuery
    //-----------------------------------------------------------------------
public static ArrayList doUserLogin(String sql) {
        ResultSet rs=null;
        ArrayList one=new ArrayList();
        try {
                Connection con=prepareConn();
                Statement stmt=con.createStatement();
                rs=stmt.executeQuery(sql);
                while(rs.next()){
                    one.add(rs.getInt(1));
                    one.add(rs.getString(2));
                }
                closeConn(con);
            }
            catch (Exception ex) {
                System.out.println ("Exception in executeQuery()...."+ex);
            }
        return one;
    }   //-----------------------------------------------------------------------
}//class
```

4.2.3 Frontend Implantation

For the front-end implementation of web based user interfaces I used HTML and JavaScript as JQuery Framework with Bootstrap to implement better dynamic responsive web pages design.

4.3 Implementation model of OOSE

After deriving the database design from the class diagram, implementation of the database is done using MySQL. The database is password protected and database administrator has ability to restrict its usages as desire.

As this project uses MVC (Model, View, Controller) Architecture to structure the system. Model contains all the logic and interactions with database level and these modules are used collaboratively between many web pages through the controllers to enhance reusability. ‘VehicleCommon’ java file and ‘VehicleServlet’ java file contains coding to act as the main model and main controller in the vehicle managements system. Other jsp files contain the coding regarding to view the system. Main JSP file of the system is ‘index.jsp’.

```
<html>
<head>
<title>TMS</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
<link href="include/admin.css" rel="stylesheet" type="text/css">
<link href="include/menu.css" rel="stylesheet" type="text/css">
<link href="include/main.css" rel="stylesheet" type="text/css">
<link rel="STYLESHEET" type="text/css" href="include/calendar.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
<script language="JavaScript" type="text/javascript" src="include/simplecalendar.js"></script>

  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

<script>
function CloseSession(){

        //if(confirm('Want to Close this session - index ?')) {
         //  window.location.href='do?MOD=MTN&ACT=clsSess';
        //}
}
</script>

<style type="text/css">
<!--
html,body{
   text-align: center;
   background-image: url(images/pattern.gif);
}
.style1 {
          font-size: 16px;
          font-weight: bold;
          color: #000066;
}
.white{
color:#FFFFFF;
}
.style2 {color: #EED9A8}
-->
</style>
</head>
<body onbeforeunload="CloseSession()">
<div class="">
<table width="1000" border="0" align="center" cellpadding="0" cellspacing="1" class="graybox">
  <tr>
    <td colspan="2">
            <%@ include file="include/header.jsp" %>
         </td>
 </tr>
 <tr>
  <td colspan="2"  bgcolor="#000066"  style="height:25px;"><span class="style1">
   <marquee truespeed="truespeed">
         <span class="style2">Your transport solution is our duty.... </span>
   </marquee></span></td>
 </tr>
 <tr>
  <td width="20%" valign="top" bgcolor="#EED9A8" class="navArea"><p>&nbsp;</p>
   <%@ include  file="include/umenu.jsp" %>
   <p>&nbsp;</p>
   <p>&nbsp;</p>
   <p>&nbsp;</p>
   <p>&nbsp;</p>
   <p>&nbsp;</p>
   <p>&nbsp;</p>
   <p>&nbsp;</p>
   <p>&nbsp;</p>
  </td>
  <td width="80%" valign="top" class="contentArea">
    <table width="100%" border="0" cellspacing="0" cellpadding="20">
    <tr>
     <td>
<%
String ename = (String)session.getAttribute("uname");
System.out.print(ename);
String conPage=(String)session.getAttribute("content_page");
System.out.print("conPage ==> "+conPage);
if(conPage!=null){
%>
<jsp:include page="<%=conPage%>"/>
<%
}
else {
%>
<jsp:include page="welcome.jsp"/>
<%
}
%></td>
    </tr>
   </table></td>
 </tr>
</table></div>
<p>&nbsp;</p>
<div class="white">
<p align="center">Copyright 2016 - 2019  - <b>National Savings Bank.</b> - All Rights Reserved <br>
<p align="center">Head Office, Savings House, 255, Galle Road, Colombo 3, Sri Lanka Tel: 011-
2573008-15,  Fax: 011-2379333,   E-mail: manager.transport@nsb.lk<br>
</div>
</body>
</html>
```

4.4. User Interface Design

To design the user interface web pages the concept of LUCID(Logical User Cantered Methodology) by Kreitzberg is used to provide user the best user friendliness from the system.

As the prototype used in requirement gathering it is easily gathering the user interface requirements. The screens were design by HTML and paper prototyping.

Since most of the internal users are interact with the system full day, it is necessary for going on light colour, but components are easily viewable them there it is decided to use substance themes for this user interface designs. After the implementation of the interface, the training and help through the system is done.

But for the external users, interfaces are designed with more attractive features and combining bright colours.

Java script files were used to design the users interface. Cascading style sheets with Bootstrap features were used to improve the graphical view the user interface parameters such as forms, buttons, menus, text boxes, fonts and etc.

Sample user interfaces of the system are in the Appendix B.

Chapter 5: Evaluation

Major objective of the testing phase is to produce an error free system to the client. Testing is an activity that helps in finding out bugs/defects/errors in a software system under development, to provide a bug free and reliable system/solution to the customer.

Testing is broadly deployed in every phase of the software development. Each method is tested with white box testing and black box testing. Thereafter, classes are tested in combination. Then group of classes are tested as cluster testing. Finally, the full system is checked for integration. User validation and verification with the document is done eventually.

5.1. Objectives of Testing

System testing was initially done by us since we have access to all the relevant database tables and user interfaces which lead to use live data. Later system was installed in client environment for testing. Considering functionality of the system, system was tested in three stages.

- Component Testing – Each individual data table and user interfaces has tested for the functionality.
- Integrated systems testing – Component tested as integrated system for the interdependencies of the component.
- Network testing – System is testing in local area network as well as over the internet for client server communication.

Software testing is the crucial element of software assurance and represents the ultimate review of specification, design and code generation.

The objectives are:

- Test the system to check out and verify whether the functional requirements are satisfied.
- Identify the bugs or defects in the implementation so that they can be debugged and rectified.
- Identifying the way, the system responds to different scenarios and the robustness and helpfulness for erroneous situations.

Based on the results of testing solution can be further improved in processing performance, which might have been overlooked in previous stages. Further testing provides the usability of the system, based on which enhancements could be carried out.

Testing always ensures the following:

- System comes to bug free position.
- Separate procedures/function performance and check whether those are working properly.
- System quality kept to the expected level.
- Check whether the system provide exact user requirements.

5.2. Test Model of OOSE

At the beginning for each class it is checked whether relevant methods and operations are captured, further check whether all attributes related to the objects are identified. These attributes specify various stages of class can undergo, also whereas the operations provide the relevant interface to communicate with it. Message passing between object lets them to communicate to perform task successfully. Therefore, it is necessary to perform that.

5.3. Unit Testing

As the system require higher degree of accuracy each method is tested using white box testing and black box testing. As example model class functions are tested with mock data to in and out to in the controller to identify functions are working at expected level.

Further unit level test cases are written to verify black box testing. Below is an example for it to verify login functions. Test Plan and Test Cases are further elaborated in Appendix C.

Table 5: Test Case - Login to the system as customer (other divisions and branches)

Test case No	01
Test Case Type	Unit Testing
Test Case Name	Login to the system as customer
Test Case Description	Registered user logged in to the system

Test Data	
Valid	Invalid
Valid Username Valid Password	Invalid Username Invalid password
Expected Output	
Valid	Invalid
User should navigate to user portal of user.	Inform login combination invalid message. Should show link to the registration page.
Result	Pass

5.4. Integration Testing

After ensuring the accuracy of the methods, model classes are tested overall. It is identified whether the classes contain all relevant methods and attributes.

For example, the model class "*Reservation*" has several methods and delegates which will be called during triggers from events. Some methods are called during runtime when the form loads. If the methods and delegates give the expected output without an error, when we run the class using controller classes to ensures its accuracy.

5.5. System Testing

All classes are tested. Therefore, it is necessary to check whether all classes work together by-passing messages between models as well as controllers and views.

Further, to make sure the accuracy of the system, a complete set of accurate data is taken from the users. The relevant customer details, driver/vehicle details, and vehicle allocation and transaction details are taken to be entered. Further, these details are manually processed by some users. Ultimately, details are compared to check the accuracy. Until the system provides the accurate results this was carried on thrice. However ultimately it was successful.

5.6. Acceptance Testing

This was done together with clients. Selected staff from the client was given with created test data to perform activities. Their feedback on the system and their exception levels were recorded for improvements. Level of user satisfaction was the key factor considered and as below mentioned usability testing was carried out through the project to ensure what is creating is what user wants. Sample testing feedback form is included in the Appendix C.

5.7. Regression Testing

Finally, the system scenarios test cases are written and some of those static test cases are automated using selenium java client version to save time in regression testing process as continuously integrate new changes to the code to make sure nothing new broken from those. Results of testing are included in Appendix C.

5.8. Usability Testing

Usability testing is a necessity to test as that reduces the highest risk impact to the system to user acceptance. In early stages of the project UML diagrams and thereafter prototype web interfaces are used to evaluate and get feedbacks on user experience to enhance the system while developing.

The developed web interfaces were shown to the clients and the functionalities described. The interfaces were tested to ensure that only the required information is available. Then all the JSP pages were put through – Performance testing, where most of the pages were tested separately for the performance of the individual pages. The pages that cannot perform individually (depend on other pages and java classes) were tested for performance after the individual performance testing of the other JSP pages were done. When the interfaces were shown to the client, at some occasions they came up with some more requirements and suggestions. The possible actions to adopt them in the system were also made.

Under performance testing, 'load testing' and 'volume testing' was done with the help of the organization staff. HTTP load tool was used to determine how the server handles the workload. Further after complete development evaluate on usability by getting feedbacks on parallel run period by letting them experience the system hands on.

5.9. Evaluation of the system

Evaluation can be considered as systematic acquisition and assessment of information used to provide feedback about developed system. Therefore, the results of the system evaluation can be considered as key indicators in accessing the degree of success associated with the development process.

The evaluators were selected carefully to evaluate all the related operations of the system. Selected staff of the bank and the officers were the evaluators of this system. The system was tested by using the dummy data on offline. Further the site is planned to be evaluated or modified according to the feedbacks of users once it is sited.

The evaluation criteria were selected for evaluation and some of the main criteria used for assessment as follows. Test plan summary is included in the Appendix C.

- Overall functionality of the system
- Usability and practicality of the system for use in day today activities.
- Performance and speed
- Overall assessment of the system
- Website information contents assessment, usability of the site, attractiveness.

The evaluation was carried out mainly based on the feedback obtained from the evaluators during and after the system demonstration.

5.10. Test Plan

The Test Plan describes the scope, approach, resources and schedule of intended test activities. The system has been tested under following criteria. The plan is attached to Appendix C.

- Function Testing
- Usability Testing
- Content Testing
- Interface Checking
- Compatibility Testing

5.11. Evaluation Result

Most of the evaluators were satisfied with the functions offered by the system and they were agreed this can be used as a final website.

The following section lists out the main areas that gave more satisfaction and the improvement are which were pinpointed by the evaluators.

5.12. Evaluator’s Satisfactory Points

- User-wise access controls and functionalities
- Use of combo boxes to minimize the typing effort and rejects the input data
- Related information for a selected or given criteria will be automatically displayed
- Restriction of invalid characters during the data input
- Overall functionality of the system
- Meaningful warnings and errors to users of the system

5.13. Proposed Improvements

- Certain printed statistics such printed reports details
- Integration more payments methods
- Introducing own payment wallet

Chapter 6: Conclusion

Critical appraisal is the process of systematically examining developed software to judge its value and relevance. Purpose of this chapter is to argue about the developed system. Web application was developed to transform the current manual system of National Savings Bank Transport Division. The significant importance of this system is its networked and centralized architecture, allowing several client divisions to connect and access the same data of this system to reduce ambiguity and increase availability of the information. Clients and Drivers are usually located all around the country, which needs to communicate with the office to get jobs and with customer to find where they are. Critical analysis of the developed system will be discussed next.

6.1. Evaluation of the Project

At the end of the project, all the required functional, nonfunctional requirements were completed while adhering to constraints of the project. Also, client requirements that were given while system development was also satisfactory implemented.

We were delighted with the response received from the bank on implementation of the project on time. They were very happy with the easy to use, attractive and web interfaces which the staff could easily learn and business model that generates new revenues to the company.

In the beginning of the project some requirements and processes were bit more complex and accessibility to the business data should be limited according to each employee’s task without blocking tasks he needs to do, this lead to do requirement gathering process in more detail way which helps us to identify every business process very clearly.

There centralized database was chosen, because of some data should be able to viewable separately as well as in combination to provide all necessary information to make quick decisions without having need to go through many interfaces. Therefore, centralized database was necessity.

The project has been done as client server system by using HTML and Java Script as a front-end and Java and MySQL Server as the back-end. Moreover, OOSE has been selected as the system design. Reasons for choosing these options were clearly defined with comparisons with other comparatives are clearly described in the introduction chapter at Designing and Methodology chapters of the project report.

Whole project and report were done by the writer. Many books and internet had to refer to enhance the knowledge. The knowledge obtained from Rapid Application Development module was very helpful. Program Design and Programming lectures conducted by UCSC to develop system using Java language and NetBeans IDE Framework were used in the system development. We had no previous knowledge in MySQL server however content we got from Database Systems subject help us to overcome those problems with database.

Interface designing, and database designing were done by using diagrams drawn from the information gathered, and then those database tables were further developed by normalizing before using into coding.

Finally, it would be pleasuring to inform that this system was able to solve all the problems informed by the Transport division of the bank. Moreover, the system will lead to improve efficiency & productivity of the employees of transport division as well as save money and time with improving business quality and security as a banking support service.

6.2. Project Management

While the developer was able to complete the project close to originally planned time but for several sections such as development and implementation sections needed much more time than the proposed time duration at the beginning. This was purely due to the under estimation if time needed to learn new technologies used in the project. Several new features were needed to be carefully examined and studied to implement them in the developed system. The detailed analysis took much time to verify the actual requirements of the training process.

The developer had realized that many new skills set to be acquired or unfamiliar technologies introduced in a project could greatly affect the overall project schedule. Hence, if the developer has to re-draw this or draw another project schedule, this can be taken as an important learning point for the overall project planning.

6.3. Lessons Learnt

From the beginning of the project it was a great source of learning and experience since, no previous experience in business environments and no previous knowledge in how NSB transport division’s services operate in depth, so this became challenge.

However, by going through the project life cycle, learnt in-depth processes of the business environment and learnt accounting from many resources handle all day to day accounting on transportation division which is different from traditional banking operations.

Therefore, that knowledge becomes great assert to create this system and any further systems in that domain.

Further, handling client was become a great experience, as learning how to understand client's expectation, how to make them convince on the project progress without using technical stuff and how to get process in their mind into use. Therefore, that becomes good experience on working with client which learnt through this project.

Finally, as there is a strict time schedule as there is only for months to complete the project, we learnt lot about time management during the project to cover up deadlines with other studies. Also, from this project we got a skill of overcoming stress during the project which can apply to whole life.

6.4. Future Enhancements

Even though the main objective of the system is to satisfy client 100% is impossible, due to client was needed further enhancements of the project at the period of implantation by using fully functioned mobile based solution to make customers (divisions and branches) and driver more interact with the division to avoid conflicts and for quality service.

Therefore, we have agreed to enhance project by enhancing existing mobile application which used to track user locations and display

Moreover, the program can be further modified in order to meet the security measures that arise in the further, which will enable the company to have high data security with many payment methods as requested by transport manager.

Finally, Backup system can also be expanded to deal with more problematic situations enabling the bank to restore fast in problems that may arise suddenly or unexpectedly.

Bibliography

[1] IBM, "5 Web Design Strategies that Can Hurt the User Experience," 2016. [Online]. Available: https://www.interaction-design.org/literature/article/5-web-design-strategies-that-can-hurt-the-user-experience..

[2] itqlick, "itqlick," 2019. [Online]. Available: https://www.itqlick.com/truckers-helper.

[3] dossier, "dossier," 2014. [Online]. Available: http://dossiersystemsinc.com/.

[4] ISS Infinity, "ISS Infinity," 2019. [Online]. Available: http://www.e-infinitysoftware.com/products/broker_pro/.

[5] Cognetics Corporation, "The LUCID Design Framework," 1998. [Online]. Available: https://www.cs.ucy.ac.cy/courses/EPL435/papers/lucid-overview.pdf.

Appendix A– System and Technical Documentation

Table 6: DB table structure for table Reservation

Column	Type	Null	Description
b_id	int(10)	No	Booking ID Code
u_id	int(10)	No	User ID
u_name	varchar(40)	No	User Name
v_name	varchar(50)	No	Vehicle Registration Number
d_id	varchar(50)	No	Driver ID
sdate	varchar(30)	No	Start Date
edate	varchar(30)	No	End Date
nod	int(10)	No	Number of Days
km	varbinary(10)	No	Distance in km
no_of_seats	int(10)	No	No of Seats
status	varchar(20)	No	Status
b_date	varchar(40)	No	Booking Date

Table 7: DB table structure for table driver

Column	Type	Null	Description
emp_id	int(10)	No	Employee ID
Emp_code	varchar(3)	No	Employee Category
f_name	varchar(40)	No	Employee First Name
l_name	varchar(40)	No	Employee Last Name
address_1	varchar(100)	No	Address Line 1
address_2	varchar(40)	No	Address Line 2
city	varchar(30)	No	City
mob	varchar(15)	No	Mobile Number
lic_no	varchar(20)	No	Licence Number
exp	varchar(20)	No	Expiry Date
experiance	int(4)	No	Experience
note	varchar(200)	No	Notes

Table 8: DB table structure for table repair

Column	Type	Null	Description
r_id	int(10)	No	Repair ID
v_name	varchar(50)	No	Vehicle Registration No
g_name	varchar(50)	No	Garage Name
d_id	varchar(50)	No	Driver ID
date	varchar(40)	No	Repairing Date
odometer	varchar(20)	No	Odometer Value
cost	int(10)	No	Cost
note	varchar(200)	No	Remarks

Table 9: DB table structure for table users

Column	Type	Null	Description
c_id	int(10)	No	Customer ID
custname	varchar(100)	No	Customer Name
uname	varchar(50)	No	User Name
pass	varchar(50)	No	Password
add1	varchar(200)	No	Address Line 1
add2	varchar(40)	No	Address Line 2
city	varchar(30)	No	City
mobile	varchar(15)	No	Mobile No
email	varchar(30)	No	email Address

Table 10: Use Case Narrative - Customer Registration

Use Case Name:	Driver /Employee Registration	Use-Case Type – Business Requirement
Use Case ID:	VM – 02	
Priority:	High	
Source:	Document	
Primary Business Actor:	Administrator	
Other Participating Actors:		
Other Interested Stakeholders:	Drivers and Employees	
Description:	All employees including drivers must be registered with their personal details. They provide their information to the administrator.	
Preconditions:	Logged in user should have admin privileges.	
Trigger:	This use case is initiated when the administrator selects the option to register details of driver/employee.	
Typical Course of events	**Actor Action**	**System Response**
	Step1: Administrator enters driver/employees' details to system.	**Step 2**: System verifies the information and added to the database.
		Step 3: Auto generated employee registration number according to their job category.
	Step 4: Send registration information to the printing Job.	**Step 5:** System formatted the registration information and print registered Driver/ Employees report.
Alternate Courses	Employee ID is generated after the driver/employee information entered.	
Post conditions	Driver/employee successfully registered, and Employee ID issued.	

Table 11: Use Case Narrative - Making Reservation

Use Case Name:	Making Reservations	Use-Case Type – Business Requirement
Use Case ID:	VM – 04	
Priority:	High	
Source:	Online Request / Conversation	
Primary Business Actor:	Customer	
Other Participating Actors:	Clerk, Driver	
Other Interested Stakeholders:	Customer – customer will receive the SMS after making the reservation. Receptionist – Reservation information need to generate reports.	
Description:	Customer login to the system and add hire details to the system. SMS containing the reservation details will send to the driver after making the reservation.	
Preconditions:	Customer should be registered user of the system. Vehicles should be available for allocation.	
Trigger:	This use case is initiated when making the reservation.	
Typical Course of events	Actor Action	System Response
	Step 1: Customer will look for the vehicle type available at the time of hire.	Step 2: System will check whether the customer is registered.
		Step 3: Track location and check the availability of vehicle.
	Step 4: Customer will enter the hire details.	Step 5: System will add hire details for the registered customer.
		Step 6: Reservation ID is generated by the system.
		Step 7: System will send SMS to the customer and allocated driver.
Alternate Courses	Driver and vehicle are allocated to the reservation. Job Alert is sent to the driver.	
Post conditions		

Table 12: Use Case Narrative - Cancel Reservation

Use Case Name:	Cancel Reservation	Use-Case Type – Business Requirement
Use Case ID:	VM – 05	
Priority:	High	
Source:	Online Request/ Document	
Primary Business Actor:	Clerk	
Other Participating Actors:	Customer	
Other Interested Stakeholders:	Driver	
Description:	Customer login into the system and request to cancel the reservation. Or customer can request to the organization's clerk to cancel the reservation by giving information.	
Preconditions:	Cancellation can be allowed 24 hours before the departure time.	
Trigger:	This use case is initiated when canceling the reservation.	
Typical Course of events	Actor Action	System Response
	Step 1: Search by reservation ID	Step 2: System display search result
	Step 3: User click cancel reservation.	Step 4: system check the cancellation possibility. if possible -> proceed otherwise -> show error message "cannot proceed cancel reservation"
	Step 5: Clerk print Cancellation report when required.	
Alternate Courses	Number of cancelations per customer is recorded.	
Post conditions		

Table 13: Use Case Narrative - Manage Access Privileges

Use Case Name:	Manage Access Privileges	Use-Case Type – Business Requirement
Use Case ID:	VM – 06	
Priority:	Moderate	
Source:	Document	
Primary Business Actor:	Administrator	
Other Participating Actors:	Manager, Clerk, Receptionist	
Other Interested Stakeholders:		
Description:	Administrator can be able to insert, update, delete and view details of the system. Administrator assigns privileges to the other 3 user levels. (Manager, clerk and receptionist).	
Preconditions:		
Trigger:		
Typical Course of events	Actor Action	System Response
	Step 1: Administrator will choose the modules allowed for the user according to his/her job category.	Step 2: System will asks the modules to be select for the particular user.
	Step 3: Administrator will select insert, update or delete or view system information.	Step 4: System will update tables according to the changes has done by administrator.
Alternate Courses	The login page and accessibility changed according to the privileges given to the job category.	
Post conditions		

Appendix B – MIS Reports

Figure 7: Vehicle Details Report

Vehicle Details Report -2019-05-31

National Savings Bank – Transport Division

Vehicle Reg. No.	Vehicle Type	Make	Fuel Type	Vehicle Value	Insuring Company	Insurer Name

Prepared By: ……………………………..

Checked By: …………………………….

Figure 8: Driver details report

Driver Details Report As at ………………..

National Savings Bank – Transport Division

Employee Number	Driver Name	NIC Number	City	Mobile Number	License Number	Licence Expiry Date	Assigned Vehicle

Prepared By: ……………………………..

Checked By: …………………………….

Figure 9: Vehicle schedule report

Vehicle Schedule Report As at…………….

National Savings Bank – Transport Division

For the

Period

From:

To:

Vehicle Reg.Number	Driver Name	Booking ID	Start Date	End Date	Travelled Km	Received Amount	Status

Prepared By: ……………………………..

Checked By: …………………………….

Appendix C – Test Results

Table 14: Function Testing - Check all links

ID	Description of Test	Test	Result
1	Whether outgoing links from all the pages from the specific domain are tested?	Yes	Ok
2	Whether all internal links are tested?	Yes	Ok
3	Whether links are jumping on the same pages?	Yes	Ok
4	Whether there are any orphan pages?	Yes	Ok
5	Whether any broken links are in all above-mentioned links?	Yes	Ok

Table 15: Test for all pages

ID	Description of Test	Test	Result
1	Whether all validations on each field are checked?	Yes	Ok
2	Whether default values of the field are checked?	Yes	Ok
3	Whether invalid inputs to the fields in the forms are checked?	Yes	Ok

Table 16: Database Testing

ID	Description of Test	Test	Result
1	Whether data consistency is existed?	Yes	Ok
2	Whether data integrity and no error are in the data?	Yes	Ok
3	Whether database queries are executed correctly?	Yes	Ok
4	Whether data is retrieved correctly?	Yes	Ok
5	Whether data is updated correctly	Yes	Ok

Table 17: Usability Testing - Navigation

ID	Description of Test	Test	Result
1	Whether web sites are easy to use?	Yes	Ok
2	Whether instructions are provided clearly, correct meaning and satisfy the purpose?	Yes	Ok
3	Whether main menu is provided in each page and consistent?	Yes	Ok
4	Content Testing	Yes	Ok
5	Whether contents are logical, meaningful, understand easily, free from spelling and grammatical mistakes?	Yes	Ok
6	Whether dark color and unsuitable fonts are used?	Yes	Ok
7	Whether all the anchor links are worked properly?	Yes	Ok
8	Whether images are placed properly with proper size?	Yes	Ok

Table 18: Interface Checking

ID	Description of Test	Test	Result
1	Whether all the interactions between servers are executed properly and errors are handled properly?	Yes	Ok
2	If database or web server returns any error message for any query by application server, whether application server catches and displays these messages approximately to users?	Yes	Ok

Table 19: Compatibility Testing - Browser compatibility

ID	Description of Test	Test	Result
1	Whether web application is tested on internet explorer, Firefox, Google chrome, Opera, Safari browsers with different versions?	Yes	Ok

Table 20: Login to the systemas customer

Test case No	01
Test Case Type	Unit Testing
Test Case Name	Login to the system as customer
Test Case Description	Registered customer logged in to the system
Test Data	
Valid	Invalid
Valid Username Valid Password	Invalid Username Invalid password
Expected Output	
Valid	Invalid
User should navigate to user portal of particular user.	Inform login combination invalid message. Should show link to the registration page.
Result	Pass

Table 21: User attempt to register the systemas a new customer

Test case No	02
Test Case Type	Unit Testing
Test Case Name	User tries to register with the system as a customer.
Test Case Description	Validate the username, password strength, Mobile number and email address.
Test Data	
Valid	Invalid
Valid new Username Valid Strong password Valid Mobile Number Valid Email address	Existing Username Weak Password Invalid Mobile Number Invalid Email address
Expected Output	
Valid	Invalid
User is registered successfully, and customer ID is sent to the given email address and mobile number.	Invalid data fields are informed with a message.
Result	Pass

Table 22: Unregistered User attempt to make a reservation

Test case No	03
Test Case Type	System Testing
Test Case Name	Unregistered user tries to make a reservation
Test Case Description	Movement management and taxi reservation service is only available to the registered customers of the company.
Test Data	
Valid	Invalid
Expected Output	
Valid	Invalid
User should redirect to the login page.	
Result	Pass

Table 23 : Registered customer attempt to make a reservation

Test case No	04
Test Case Type	System Testing
Test Case Name	Registered customer tries to make a reservation
Test Case Description	After successful vehicle reservation notification should goes to the appointed driver and customer.
Test Data	
Valid	Invalid
Make valid reservation and confirm it.	Decline to confirm reservation.
Expected Output	
Valid	Invalid
Reservation details can be viewed by the customer and internal users. Job confirmation message should be sent to customer and appointed driver.	The reservation data should be wiped out and no message sent to the customer or driver.
Result	Pass

Table 24: Internal user attempt to access unauthorized content user ID

Test case No	05
Test Case Type	Security Testing
Test Case Name	Internal user tries to access unauthorized content for him.
Test Case Description	System data should be properly secured to prevent unauthorized access.
Test Data	
Valid	Invalid
Clerk/Receptionist Level user tries to access to view rates or vehicle or driver information.	Clerk/Receptionist Level user tries to access to modify/delete rates or vehicle or driver information.
Expected Output	
Valid	Invalid
Rates, vehicle and driver information are displayed for inquiry.	Should show warning message and record in activity log.
Result	Pass