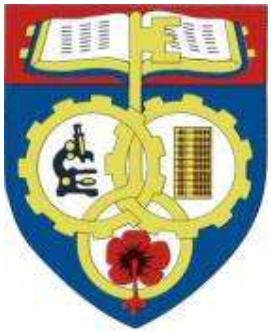
Bus Ticketing System

By

Wang Kang Li



SCHOOL OF ARTS AND SCIENCE

TUNKU ABDUL RAHMAN COLLEGE

KUALA LUMPUR

ACADEMIC YEAR

2010/2011

Bus Ticketing System

By

Wang Kang Li

Supervisor: Kong Hooi Ming

A project report submitted to the School of Arts and Science
in partial fulfillment of the requirement for the
Bachelor of Science, Campbell University, U.S.A,
And
Advanced Diploma in Science.

Division of Computer Science
School of Arts and Science
Tunku Abdul Rahman College
Kuala Lumpur

2010/2011

Copyright 2011 by Tunku Abdul Rahman College.
All rights served. No part of this project documentation may be reproduced, stored in retrieval system, or transmitted in any form or by any means without prior permission of Tunku Abdul Rahman College.

Declaration

The project submitted herewith is a result of my own efforts in totality and in every aspect of the project works. All information that has been obtained from other sources had been fully acknowledged. I understand that any plagiarism, cheating or collusion or any sorts constitutes a breach of College rules and regulations and would be subjected to disciplinary actions.

_____

WANG KANG LI

BUSINESS INFORMATION SYSTEMS

Abstract

The purpose of this project is to fulfil the requirements of Bachelor of Sciences and Advance Diploma in Business Information Systems. This project must be submitted as a Final Year Project in order to graduate.

This system is let the member to make reservation for the bus ticket via online and the administrator can do some back end work such as add, edit, delete and view the information. Customer can make payment via online after they make reservation so that they no need waste the time go for counter make payment.

This system is developed based on two-tier approach with step-by-step from planning, analysis, design, and then implementation. With the Unified Modelling Language (UML) diagram such as use case, activity diagram and sequence diagram is drawn and t. The system flow and the actor are being drawn to understand how the system works does and who the actors that interact with the system are.

The programming language used to develop this project is VB.Net and the tool is Microsoft Visual Studio 2008. Database application is Microsoft SQL Server 2008.

Acknowledgement

From the final year project, I would like to be appreciating Ms Kong Hooi Ming and Ms Tham Sin Seen for contributing their ideas and in-depth knowledge in the field. A warm thank is extended to them for sharing their resources, opinions, knowledge, experience and skills in programming and development methodology, so generously.

I would also like to personally thanks to my partner, Tan Sin Wan. When I am doing the final year project, he offers a lot of information to me in order to finished assignment. She always said any difficult mission is possible to fulfill. She often gives a lot of confidence when I face the problem. Hence, I would like to personally appreciate to her.

After that, I want to thank to our parents because they help us to solve our economy trouble. Then, I would to thanks to all my friends who have provided some opinion. As a result, I can successfully complete the project. It is very higher achievement.

This project also provides us a good opportunity to acquire team work spirit, conduct information search by using various investigation methods. Thus, we can produce reasonably well presented quality report. Completion of this assignment should also let us increase the fundamental knowledge to prepare our next level IS project.

Table of Contents

Declaration ... **iii**

Abstract ... **iv**

Acknowledgement ... **vi**

Chapter 1. System Planning ... **1**

1.1 Introduction ... 2
1.2 Project Aims and Objectives ... 2
1.3 Company Background ... 3
1.4 Project Scope ... 8
1.5 Project Schedule ... 10
1.6 Project Team ... 11
1.7 Outline of approach/Methodology Used ... 11
1.8 Chapter Summary ... 11

Chapter 2. Requirements Analysis ... **12**

2.1 Introduction ... 13
2.2 Fact Gathering ... 13
2.2.1 Research ... 13
2.2.2 Background Research ... 13
2.2.3 Interview ... 14
2.2.4 Observation ... 14
2.3 Feasibility Study ... 15
2.3.1 Technical Feasibility ... 15
2.3.2 Social and Operational Feasibility ... 15
2.3.3 Economic Feasibility ... 17
2.4 Functional Requirements ... 19
2.4.1 Bus Scheduling Module ... 19
2.4.2 Repairing Module ... 20
2.4.3 Reservation Module ... 20
2.4.4 Maintenance Module ... 21
2.4.4.1 Staff Maintenance ... 21
2.4.4.2 Member Maintenance ... 22
2.4.4.3 Bus Maintenance ... 23
2.5 Non-Functional Requirements ... 23
2.6 Software and Hardware Requirement for Development ... 25
2.6.1 Software Requirements ... 25
2.6.2 Hardware Requirements ... 28
2.7 Software and Hardware Requirement for Operational ... 28
2.7.1 Software Requirements ... 28
2.7.2 Hardware Requirements ... 30
2.8 System Architecture Diagram ... 31
2.9 Chapter Summary ... 32

Chapter 3. System Design ... **33**

3.1 Introduction ... 34
3.2 Use Case Diagram for Proposed Bus Ticketing System ... 34
3.3 Activity Diagram for Proposed Bus Ticketing System ... 53

3.4 Sequence Diagram for Proposed Bus Ticketing System 73
3.5 Database Design 90
3.5.1 Data Dictionary 90
3.5.2 Normalization 104
3.6 Class Diagram 107
3.7 Screen Design 108
3.8 Chapter Summary 118

Chapter 4. Programming 119

4.1 Introduction 120
4.2 Programming Language Used 120
4.3 Coding 120
4.4 Chapter Summary 135

Chapter 5. Software Testing 136

5.1 Introduction 137
5.2 Sample Data 137
5.3 Test Case 148
5.4 Chapter Summary 155

Chapter 6. Conclusion 156

6.1 Introduction 157
6.2 Link to Seminar 157
6.3 Justification of Choice Tools 157
6.4 Evaluation againsts the project objectives 158
6.5 Project Management Issues 159
6.6 Future Improvement 159
6.7 Personal Reflection 159
6.8 Summary 160

References 161

Appendices 162

User Guide 164

Chapter 1

System Planning

1. System Planning

1.1 Introduction

This chapter will introduces about company background and overview of the system that developed which is the project aims and objectives, summary modules description, project scope, project schedule, project team and the outline of approach or methodology used.

1.2 Project Aims and Objectives

- **Convenient**

Creating this new system can bring a lot convenient to customer and staff. For example, if customers need to book the bus ticket, they no need go to the bus station to booking or buying the ticket especially the person who are staying very far to bus station. They can direct use this bus services booking via internet; it can bring a lot convenient to the customer who driving to the bus station for buying bus ticket. It also brings convenient to people who are busy, so they can online booking at anytime.

Besides that, this system also brings a lot convenient to staff when they are able to search the information for customer such as search the bus available time and seat. It also enables staff to maintain the data easier.

This system also can bring convenient to the manager such as enable manager to view the report easier and this system can use to summaries the top sale or provide the daily, monthly or yearly report and etc, so manager no need to expand a lot of time to view the hardcopy report .

- **Improve efficiency**

The process time of the current system are time consuming because the current system are using paper work to record down all the information. It is take a lot of time to serve the customer when the staff searching the available time and seat. The purpose of create this system is to improve the efficiency time of process. It can let the staff search the information faster and easier and maintain the data faster and reduce the paperwork.

- **Increase income**

Made reservation using online method can attract more customers because made reservation online can save a lot of time for the customer. If can let the customer feel convenient, so that they will come to our company buy the bus ticket more often when they want go to somewhere around Malaysia so we can earn more income from them.

- **Bus and ticket information**

This system can let the customer know the information about the bus schedule and ticket. Nowadays, online are very common issues to everyone so that checking information using online can save a lot of time to the customer. Customers no need go to the counter to asking about the information of the bus schedule and the schedule.

- **Reduce error data**

The purpose of creating this new system is let the staff to reduce key in wrong data, because the system are using computer to key in the data, there got some validation to check when staff key in the data wrongly and system will provide some selection to let staff direct choose the data. It also reduce the paper work to record all the information, so it bring the benefit for reduce error data.

1.3 Company Background

- **Nature of business**

Kang&Sin Berhad (KSB) is the largest operator of public bus transportation in Malaysia. Kang&Sin Berhad made its debut on Bursa Malaysia on 3 Dec 1999 engages primarily in the bus transportation system such as express bus operations.

It express bus operations have provide the most extensive coverage throughout Peninsular Malaysia, and covering all major cities, towns as well as Singapore. The nationwide network is serviced more than 2000 buses that cover more than 200 routes and generating 800 trips in a day that around 50 million passengers a year. Kang&Sin Berhad has the largest market share with “Kstar” being the leader in the express bus industry in Peninsular Malaysia.

- **Kstar**

Kstar is a Malaysian icon and the leader of the express bus industry. It is one of the most prominent and popular household brand names in Southeast Asia region. It serving more than 200 destinations and covering all major cities and towns with 1000 a day departures across Malaysia and Singapore. Kstar provides safety, enjoyable and affordable travel by using the latest advancement in bus technologies.

Kstar has led the industry by constantly innovating itself in the express bus industry for over 3 decades. After them using the electronic ticketing system, Kstar has made the booking process much faster, accurate and efficient as it enables the passengers to plan their journey, make reservation and purchase express bus tickets at any ticketing counter.

Kstar was also the first to provide insurance coverage for passenger and the baggage, whereby in the case of unfortunate event, the passengers are covered for medical expenses or loss of baggage. In addition to its regularly scheduled passenger service, Kstar also provides some other services such as courier/delivery services, charter packages and bus advertising solutions.

- **Product & services**

  - **Express Bus**

    - ✓ **Kstar**

Figure 1.3.1Seat of the bus

Kstar are further divided according to the bus routes such as Western Region, Eastern Region and Southern Region. For Western Region manages all express routes within this region that stretches from Kuala Lumpur in the south, all the way to Kangar up north. For Eastern Region originating points in Eastern Region are mainly from Kota Bharu, Kuala Terengganu and Kuantan and the Southern Region originating points in Southern Region are mainly from Johor Bahru, Melaka and Seremban.

Kstar fleet availability improvement exercise, all new Kstar buses are under the repair and maintenance contract with the original chassis manufacturer. In terms of safety, all new Kstar buses met the European ECE R66 safety regulations on roof crash standards and equipped with front row seat belts since 2005. Besides that, for passenger's convenience, Kstar tickets can be purchased 60 days ahead of departure time.

For the product innovation, the 2009 version e-ticketing for Kstar is able to place a passenger who prefers to sit beside them. Kstar is the first and only bus operator to introduce adjustable headrest for it newly launched economy express 41 seater single-deck "Club Class" and high capacity economy express 61 seater double-decks "Sky view". It not only can carry more passengers, its "twin-configuration" seats can be separated to create more shoulder room for each and every passenger on board. It also has unique design to consider all safety aspects such as an emergency door on the upper deck that is in-line with the hidden ladder placed at the bottom deck, two escape hatches on the roof and etc.

- Bus Charter Services

  The Kstar has provided rent bus services of travel. Travel & Tours offers safe and reliable service for all your charter needs. From short local charters to all-day to multi-destinations and trips, we are here and ready to serve.

- **Organization Structure**

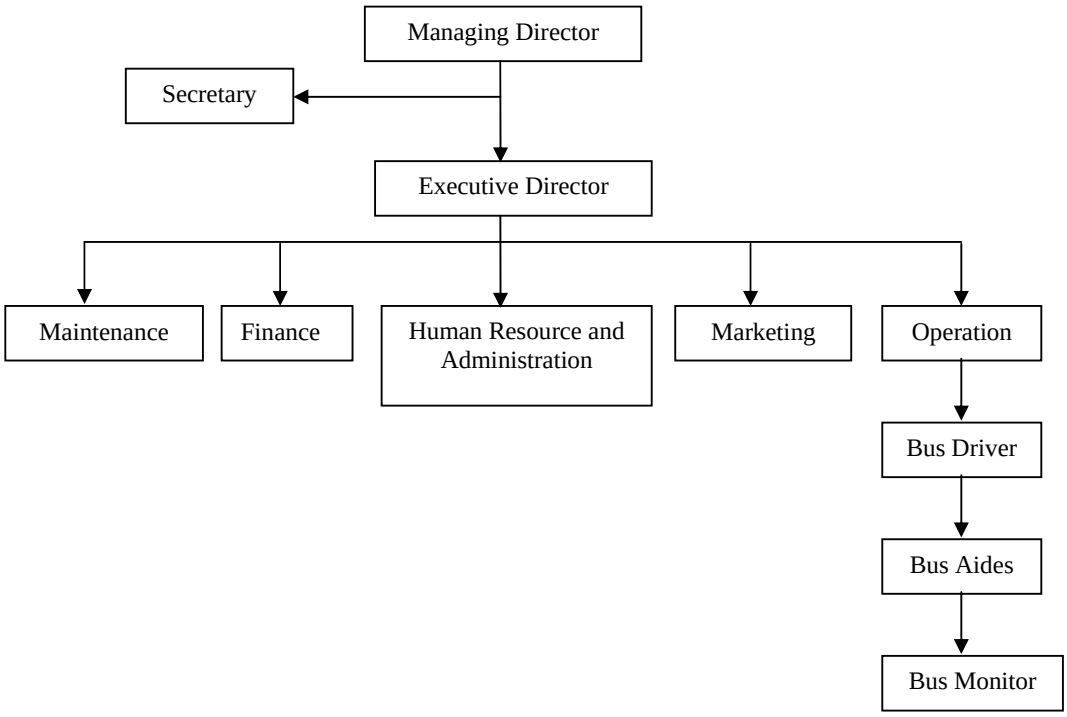


Figure 1.3.2 Organization Chart

- **Current System Used**

Kstar will open from 6.30a.m. to 9.00p.m. Our business will open every day for the customer. Counter for selling ticket will have a staff to write down the destination, time, date on the ticket, has a punch card machine to let the staff punch their card to record the arrive time for working, has a ticket book for record the ticket. When the customer tell the staff where is the destination that he/she want go then the staff will write down on the ticket and make sure with them one more time if correct then will pass the ticket to the customer.

- **Business Process**

Kang&Sin Berhad (KSB) is the largest operator of public bus transportation in Malaysia. Their mission is striving to become a conglomerate that is progressive, innovative, profitable and customer oriented and committed national aspirations. Besides that, they have their core value such as customers, human resource, shareholder, suppliers, and sincerity, social responsibility.

Kang&Sin Berhad (KSB) operation express services brand name is Kstar. The nationwide network is serviced by a fleet of more than 1,500 buses that covers more than 250 routes, generating more than 1,000 trips daily and this translates to around 60 million passengers a year. Sometime, there has many customer are come from the foreign country are travel their bus services for travel tourist. KSB usually cooperates with the traveling company and the hotel, so they can be responsible to help their traveling customer to reserve the hotel for the customer and also join the package travel with the traveling company for promotion.

Besides that, they also have some supplier for example they need to always service their bus engine to provide the better service and comfortable to the customer when they are travel bus. So that customer just will continue to travel using their company bus services. This organization has the competitor especially is the others bus services such as Delima Express, Jebat Express and etc.

- **Business Environment**

Kang&Sin Berhad (KSB) is a two level company that have about three counters for the Kstar to sell the ticket to the customer. Inside the counter have one table and one chair for the staff. Those counter is very convenient to the customers because is nearest to the LRT station. Beside the company, have much other company around our company so that those people whose workings at there very convenient if they want go to buy the ticket.

- **Size of operation**

Kang&Sin Berhad (KSB) is the largest operator of public bus transportation in Malaysia. In the company, they has about 60 employee such as 1 managing director, 2 secretary, 2 executive director, 10 person in human resource and administration department, 10 person in finance department, 8 person in marketing department, 27 person in operation department such as 5 person bus monitor, 3 person bus aides and 19 bus driver.

1.4 Project Scope

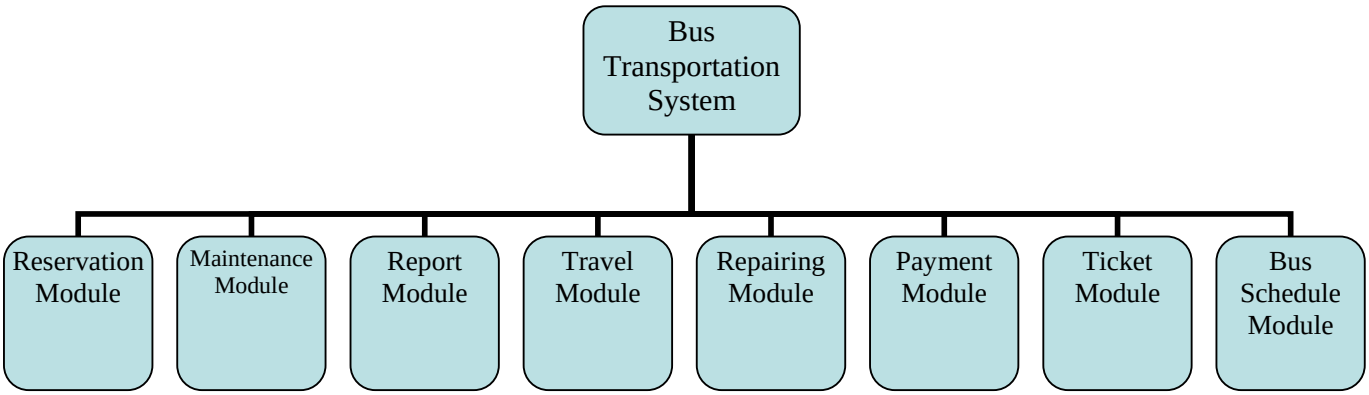


Figure 1.4.1 Main Modules of Bus Transportation System

❖ **Reservation Module**

This reservation module allows the member to make reservation via online to reserve the bus ticket where and when they want to go from one destination to another destination. After reserve the bus ticket then the member must make online payment for the reservation then the detail will store inside the database.

❖ **Maintenance Module**

This module is to help the organization to maintain people and bus information. It used to record all customers and staffs information such as name, IC and address in an organization and record the bus information.

The purpose of using this module is use to maintains the customer, staff and bus information. This module has a functionality of adding a staff and customer's information. After the information was added, if the users need to change the information in the record, this module will let the user to edit and update their new information. This module can let the staff to view the information for checking the records. Besides that, the module can let users to delete the record from a database if the staffs are resigning in the organization and the customer no longer belong the organization.

This maintenance module also can let the user add on the new bus information when company buying the new buses. It also can be modify, view or delete the bus information. The maintenance system also allow user to view the services maintenance for bus to remind the user always checking or make maintenance for bus frequently. This maintenance module also can let the staff, customer to upload the photo so the management can know what are look like for the staff and customer. The staff also can upload the bus photo when the buses go for maintenance so that they can know which bus send for maintenance.

- **Repairing Module**

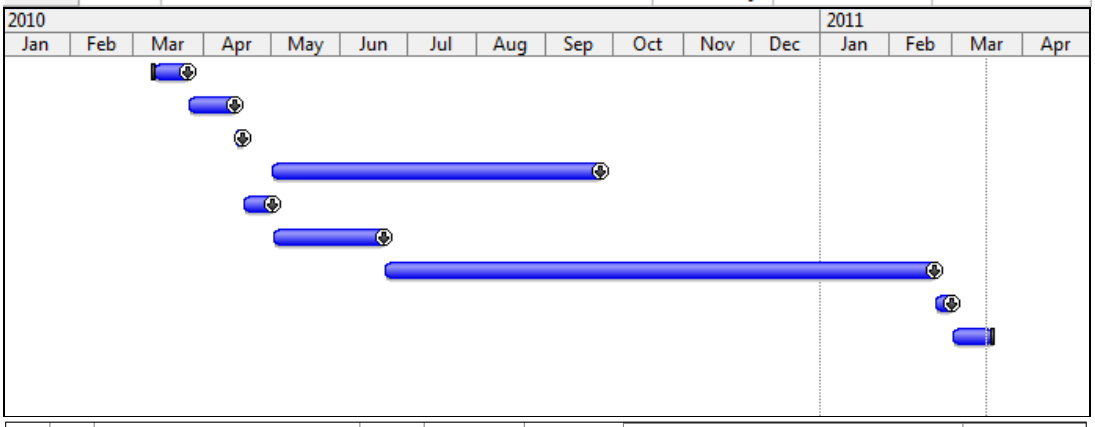
This module is let the administrator to add new service information to the database when have bus send for service so the administrator will choose which service company that they want send the bus go for service. After the bus send back from Service Company so the administrator can edit the service details such as insert the maintenance fees for the particular bus. Administrator also can view the service details to know which bus still under maintenance so they can arrange the schedule according bus number and the status for the bus is ok.

- **Bus Scheduling Module**

This bus scheduling module is use to let the administrator to add the new schedule and the destination for the driver to drive the bus according to the schedule that assign by the administrator. Administrator have the authority to delete and edit the schedule details for the driver if need make necessary changes for the schedule. Driver also can view the schedule according to the time table so that they can follow the schedule to departure from one destination to another destination on time.

1.5 Project Schedule

	Task Name	Duration	Start	Finish
1	Proposal	15 days	Wed 3/10/10	Wed 3/24/10
2	Project Specification	21 days	Thu 3/25/10	Wed 4/14/10
3	Review and Moderation of Project Specification	4 days	Thu 4/15/10	Sun 4/18/10
4	Project Progress Meetings and Deliverables	147 days	Sat 5/1/10	Fri 9/24/10
5	Analysis	13 days	Mon 4/19/10	Sat 5/1/10
6	Design	50 days	Sun 5/2/10	Sun 6/20/10
7	Coding	244 days	Mon 6/21/10	Sun 2/20/11
8	Project Testing	8 days	Mon 2/21/11	Mon 2/28/11
9	Final Documentation	18 days	Tue 3/1/11	Fri 3/18/11



ID	Task Name	Duration	Start	Finish
1	Proposal	15 days	Wed 3/10/10	Wed 3/24/10
2	Project Specification	21 days	Thu 3/25/10	Wed 4/14/10
3	Review and Moderation of Project Specification	4 days	Thu 4/15/10	Sun 4/18/10
4	Project Progress Meetings and Deliverables	147 days	Sat 5/1/10	Fri 9/24/10
5	Analysis	13 days	Mon 4/19/10	Sat 5/1/10
6	Design	50 days	Sun 5/2/10	Sun 6/20/10
7	Coding	244 days	Mon 6/21/10	Sun 2/20/11
8	Project Testing	8 days	Mon 2/21/11	Mon 2/28/11
9	Final Documentation	18 days	Tue 3/1/11	Fri 3/18/11

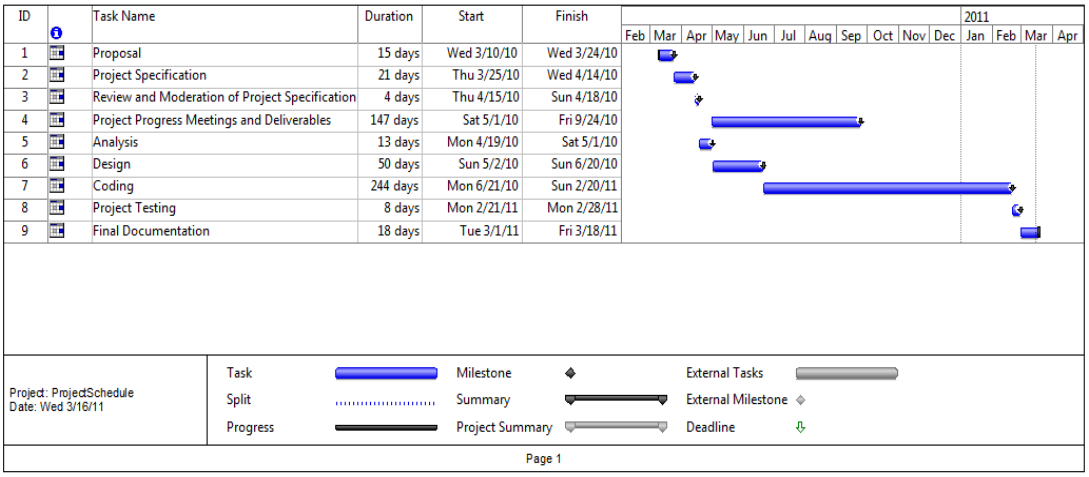


Figure 1.5.1 Gantt chart for develop Bus Transportation System

The figure 1.5.1 above is a Gantt chart that shows the tasks of a project for develop the bus transportation system, it will show the steps of the project or process, their sequence and their duration are known. The chart shows that the 9 task we need to be carry out, we estimate it will take 520 days to finish all the task, start from 3 march 2010 to 18 march 2011. The horizontal bar is show the duration of the tasks that we are going to carry out.

1.6 Project Team

Tan Sin Wan	**Wang Kang Li**
Payment Module	Reservation Module
Travel Module	Repairing Module
Ticket Ordering Module	Bus Scheduling Module
Report Module	Maintenance Module

Figure 1.6.1 Distribution of workload within team

1.7 Outline of approach/Methodology Used

To develop a function-able and good quality system, the software development approach that chosen must be suitable for developing the certain system, it must also provide all the features that needed for developing the system. In this project, which is to develop a bus ticketing system, the object-oriented approach is chosen.

The object-oriented approach with the three tier design, it has the ability of encapsulation, inheritance and polymorphism that provide the facilities of the data hiding and reusability of the code. The objects created in the program can be reused and the internal structure of the objects it hide using encapsulation to protect them from corruption. So it made the system maintenance easier and shortens the development time.

Object-oriented approaches also reduce the complexity of the system development. This is the reason why object-oriented approach is used in developing this project.

1.8 Chapter Summary

As a conclusion, the preliminary investigation of the system is needed to carry out to assess whether the new proposed system is feasible. The project objective need to state clearly to make sure the project can successfully produce.

Chapter 2

Requirements Analysis

2 Requirements Analysis

2.1 Introduction

This chapter will discuss about the requirement analysis which included Fact Finding, development environment, operational environment, software and hardware requirement, functional requirement. It will list out the minimum requirements required for the project to have. Fact finding will discuss about the information gathered to develop the system.

2.2 Fact Gathering

2.2.1 Research

Research is one of the fact-finding methods that can used to get more information regarding the company and the user so that the developer can produce a system that can fulfill the user requirement and the needs of develop a new system.

Those popular bus companies' website that researched by me such as Transnational, Konsortium and many more bus company's website that can let me to get more information regarding how a bus company website works. This is to make sure the requirement and the quality of the system is met. Having that website that can let me to refer for develop this system is very good opportunity to let me to learn more from different point of view so that we can produce a better system for the user used.

2.2.2 Background Research

Background research is also another method of fact-finding gathering the information about the company's nature of business, product and service, organization chart, and the current system of used.

By having this technique so those system analysts and the developer can use those information that gather to develop the system that based on the user requirement and fulfill the user requirement for the user. So this background research is an important starting point for the developer and the system analyst to build good system for the user that is based on the user requirements.

2.2.3 Interview

Interview is another method of fact-finding to get the information and gather the information during the system analysis phases to develop the system because the system will develop based on the user requirement.

Interview provides a better and faster response compared with others fact-finding methods because can direct get the information from the user so we can know what are the requirement from the user.

Hence want to get the first hand and more accurate information regarding the system so I ask my sister's friend Mr. Goh who is currently worked at the Mayang Sari Sdn Bhd so I can know more information from different view. I had asked him some question as below:

1. Does your company have used any system?
2. Do you think the current system user friendly for the user?
3. Do you think this current system easy to use for the user?
4. What do you think about user interface of current system?
5. What do you like and dislike about current user interface?
6. Does this is current system often error occur?
7. What suggestion do you have for improving current system?

2.2.4 Observation

Every time when we want back to the hometown so we need go to the counter buy the bus ticket to back home. While we buy the ticket from the counter then we will observe their system to understand more on how to use the bus ticketing system and still can saw have some of the counter still using paper work that is using pen to write down the ticket information for the consumer.

Through the observation, we are more understood the different task was carrying out by different process, and the sequence of the process was important also to develop a good system for the user. Examples of the bus ticketing system that we observe are Konsortium Berhad, KKKL Berhad, and Transnational Berhad.

2.3 Feasibility Study

2.3.1 Technical Feasibility

Is it those current equipment and the existing technology can be fulfill the requirement of the new system?

Since this new system is an online and offline system so we needed to setup the facility of the computer including server for this system so we need add computer, some necessary of the hardware and software for running this new system is needed.

Those technical staff is needed for responsibility to this new system is to take care of the server and the database for this new bus ticketing system. Technical staff is the basic requirement for any system to make sure the system will not encounter any problem and if have any problem occur then will solve the problem immediately so that will affect the business operation of the company. This proposed system is currently made for the convenient of the company to the customer and the staff of the company. This new system will be more compatible with other system because we using ASP.Net to develop this new system.

2.3.2 Social and Operational Feasibility

How well the new bus ticketing system fits into the project objective of the company? Will this new system provide a better solution to the bus company and increase the performance of the company?

This new system is operationally feasible because it can be used effectively after it has been developed. The current system is inefficiently that amount of the customer is limited because they need waste time just go for the counter buy the ticket and a lot of tasks have to perform manually and using pen to write down the ticket information so the customer need wait the staff to process the ticket. This problem needs to be solved by using this new system so that the possibility of perform the daily operation can be more efficiency and effectively to the customer and also the customers will be increased for the daily operation.

This entire new bus ticketing system involved an online and the offline system. Therefore, some of the changing that will happen to the most of the current staffs, so the company will arrange some simple training to them in order to operate the new bus ticketing system very well and provide more convenient for the customer in order to have faith with the company.

The new bus ticketing system is relatively easy to use for the all user. This system is design in a simple way and meaningful for every user to used it. In other words, those staffs in the company no need go for any specialized training; they just need have some simple training from the company and spend a few times try to use it. After that, they already can know how to use the new bus ticketing system and use it smoothly to increase the profit and the performance of the company.

The purpose of develop this new bus ticketing system is to bring more benefit for the customers to feel more convenient as well as the staff during the business operation of the company. So that during the planning phase of this new system, gather the staff and the customer's opinion is important because need develop a new system that need based on the user requirement to improve the performance and increase the service for customer.

2.3.3 Economic Feasibility

Costs and Benefits Summary

COST BENEFIT SUMMARY						
PURCHASE VERICAL SOFTWARE PACKAGE						
	QTY	YEAR0	YEAR1	YEAR2	YEAR3	YEAR4
COST (RM) :						
Hardware:						
-Printer(All-in-one)	1	500				
-Server – Network Document Server	1	2,300				
- Framework Standard Server	1	55,000				
- Computer (Hardware & operating system ready)	5	10,000				
- Communication device – Security Network Monitory Device	1	10,340				
-Ether network High-speed cable modem	1	400				
-Cable		200				
Software:						
- Operating system for framework Standard server	1	6,000				
- Application software – security/Anti virus software	5	750				
- Office Software	5	2,050				
- communication support software	5	2,400				
- Furniture		2,000				
- Power Supply – Power Socket		500				
- Electric Wire		500				
- Consultant Time to install, configure and test		5,000				
- Consultant Training for users		3,500				
- Technology Support			400	400	400	400
- Maintenance, Backup and update		700	800	800	800	800
TOTAL COST (RM) :		101,140	1,200	1,200	1,200	1,200
BENEFIT (RM) :						
-Eliminate the headache and inefficiency of Paper-based system.		4,515	4,515	4,515	4,515	4,515
- Reduce time to place orders & eliminate error by using B2B punch out, integrated orders, & confirmations		5,300	5,300	5,300	5,300	5,300
- Reduce the purchasing agents place		20,000	20,000	20,000	20,000	20,000
- Reducing invoice disputes and improving invoice matching		3,000	3,000	3,000	3,000	3,000
TOTAL BENEFIT (RM) :		32,815	32,815	32,815	32,815	32,815

Figure 2.3.3.1Bus Ticketing System Cost and Benefit Summary

Techniques of Evaluating

Payback Analysis

	Costs (RM)	Cumulative Costs	Benefits (RM)	Cumulative Benefits
Year0	101,140	101,140	32,815	32,815
Year1	1,200	102,340	32,815	65,630
Year2	1,200	103,540	32,815	98,445
Year3	1,200	104,740	32,815	131,260
Year4	1,200	105,940	32,815	164,075

Figure 2.3.3.2 Bus Ticketing System Cumulative Costs and Benefits Calculation

For the Bus Company, by the end of year 3, the cumulative benefits are RM 131,260 which for exceed the cumulative cost are RM 104,740. Therefore the payback period is established during the year 3.

Return on Investment

ROI = (Total Benefits – Total Costs)/Total Costs

Total Benefits = 164,075
Total Costs = 105,940
Net = 58,135
ROI = 54.88%

Net Present Values Analysis

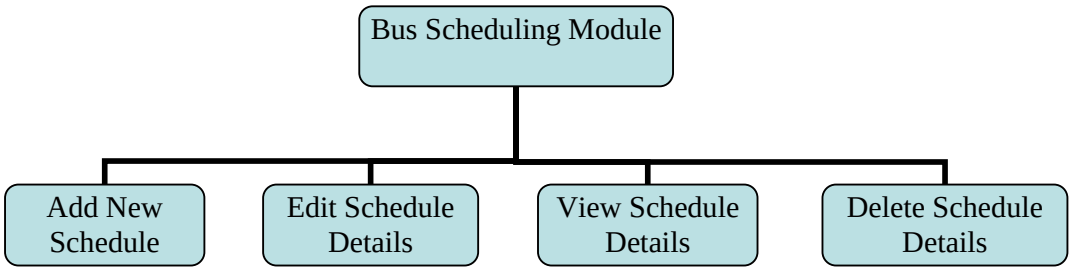
	Year0	Year1	Year2	Year3	Year4	Total
Benefits (RM)	32,815	32,815	32,815	32,815	32,815	
Present Value Factor (10%)	1.000	0.909	0.826	0.751	0.638	
Present Value	32,815	29,828.8	27,105.2	24,644.1	22,412.6	136,805.7
Costs (RM)	101,140	1,200	1,200	1,200	1,200	
Present Value Factor (10%)	1.000	0.909	0.826	0.751	0.638	
Present Value	101,140	1,090.8	991.2	901.2	819.6	104,942.8
Net Profit Value						31,862.9

Figure 2.3.3.3 Net Present Value Analysis for Bus Ticketing System

The net present value take into account of the timing of the costs and benefits, the value is adjusted by the discount rate which is 10% that provides a common yardstick and recognizes the time value of the money. After discounting all the costs and benefits, subtract the sum of the discounted costs from the sum of the discounted benefits, the net present value of the investment in Bus Ticketing System after 4 year is RM 31862.90.

2.4 Functional Requirements

2.4.1 Bus Scheduling Module



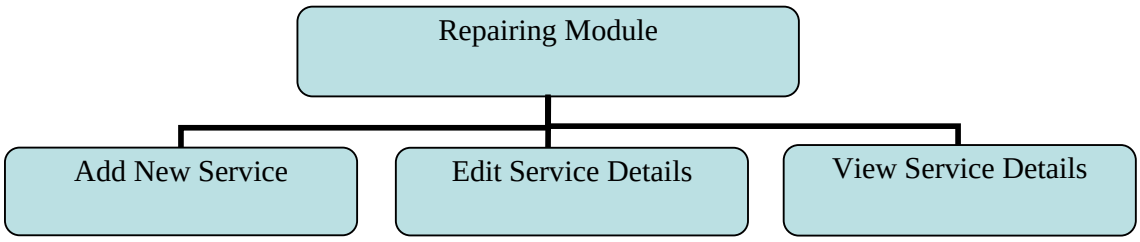
- Add New Schedule
  - ✓ The system has a function that allow administrator to add the new schedule and the destination to the driver.
  - ✓ The system will check on the same time, same day and same driver that already have the schedule or not, if yes then the system will not allow the administrator to add the new schedule for the particular driver on the same time, day and place.
  - ✓ If not then the administrator can assign the new schedule for the particular driver.

- Edit Schedule Details
  - ✓ The system has a function that allows the administrator to edit the schedule that already assigned. If have any happen occur so that the administrator need to reassign the schedule for the driver.

- View Schedule Details
  - ✓ The system has a function that allow the driver to view the schedule that assign by the administrator according to their name so the driver can know when and what time they should driver the bus from one destination to another destination.

- Delete Schedule Details
  - ✓ The system has a function that allows the administrator to delete a schedule for the particular driver if they feel that the schedule not suitable for this particular driver on that time.

2.4.2 Repairing Module

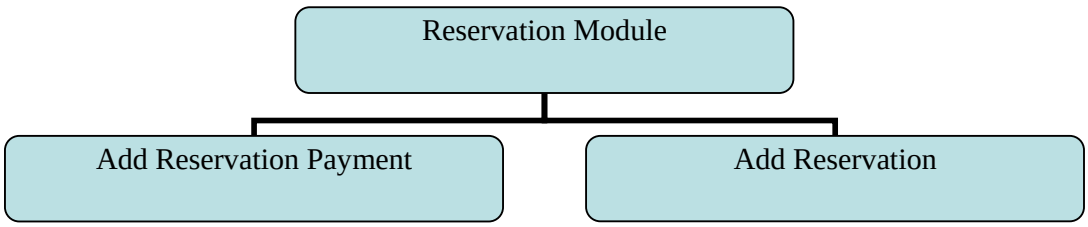


- Add New Service
  - ✓ The system has a function that allow administrator add new service for the bus if have bus send for service and the details will insert into the database.

- Edit Service Details
  - ✓ The system has a function to allow the administrator to edit the service details such as update the maintenance fees for the bus service when the bus send back from Service Company.

- View Service Details
  - ✓ The system has a function to allow the administrator to view the service details such as which bus still under maintenance.

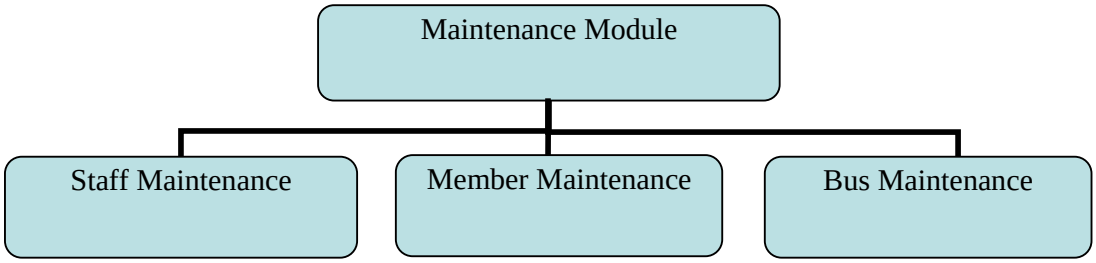
2.4.3 Reservation Module



- Add Reservation
  - ✓ The system has a function to allow the member to add reservation via online. When the member click the next button then will go to online payment form to make payment.

- Add Reservation Payment
  - ✓ The system has a function that allows the member to add reservation payment via online.
  - ✓ When the member click the next button then will go to the online payment form and the member will key in all the information in the payment form then the system will check the credit card no and the credit card expired date whether is valid or not .

2.4.4 Maintenance Module



2.4.4.1 Staff Maintenance

- Add New Staff
  - ✓ The system has a function to allow the administrator to add new staff for the company.
  - ✓ The system will check the staff IC whether already exist in the database or not, if the IC for the staff already exist then the administrator cannot add the new staff and the details to the company database. If the IC does not exist in the database so that the administrator can add the new staff and the details in to the company database.

- Edit Staff Details
  - ✓ The system has a function to allow the administrator to edit the staff details for the company if those staff have change the personal information so they will inform the administrator and the administrator will help them to update the details for them so their details will become updated.

- View Staff Details
  - ✓ The system has a function to allow the administrator to view the staff details in the company database.
  - ✓ Administrator can search their details based on the staff ID and staff Name. After select the staff ID from the system then click the search button then the details for that particular staff ID will list out in the form to let the administrator view.

- Delete Staff
  - ✓ The system has a function that allows the administrator to delete the staff in the company database if the staff resigns from company and no longer belongs to the company.

2.4.4.2 Member Maintenance

- Add New Member
  - ✓ The system has a function that allows the administrator to add new member for the company.
  - ✓ The system will check the member IC to know whether this member already exist in the company database or not, if not exist then the system will allow to add the new member for the company. If already exist then the system will not allow anyone to add the new member for the company.

- Edit Member Details
  - ✓ The system has a function to allow the administrator to edit the member details that already in the database.
  - ✓ When the members need to make any changes for the personal details then they can inform the administrator to help them edit the details that they wish to change.

- View Member Details
  - ✓ The system has a function to allow the administrator to view the member details that already inside the company database so that they can know the total amount of the member for the company.

- Delete Member
  - ✓ The system has a function to allow the administrator to delete the member if the member wishes to withdraw the membership from the company so they can inform the administrator.

2.4.4.3 Bus Maintenance

- Add New Bus
  - ✓ The system has a function that allows the administrator to add the new bus for the company if the company buys a new bus for company.

- Edit Bus Details
  - ✓ The system has a function that allow the administrator to edit the bus details such as the driver name, engine model, transmission model, color and so on.

- View Bus Details
  - ✓ The system has a function that allows the administrator to view the bus details inside the company database so that they can know have how many bus belong to the company.

  - ✓ Delete Bus

    The system has a function that allow the administrator to delete the bus inside the database if the bus already used many years and cannot used anymore for the company to carry the passengers.

2.5 Non-Functional Requirements

- **Convenient**
  - ✓ Creating this new system can bring a lot convenient to customer and staff. For example, if customers need to book the bus ticket, they no need go to the bus station to booking or buying the ticket especially the person who are staying very far to bus station. They can direct use this bus services booking via internet; it can

bring a lot convenient to the customer who driving to the bus station for buying bus ticket. It also brings convenient to people who are busy, so they can online booking at anytime.

- ✓ Besides that, this system also brings a lot convenient to staff when they are able to search the information for customer such as search the bus available time and seat. It also enables staff to maintain the data easier.

- ✓ This system also can bring convenient to the manager such as enable manager to view the report easier and this system can use to summaries the top sale or provide the daily, monthly or yearly report and etc, so manager no need to expand a lot of time to view the hardcopy report .

❖ **Improve efficiency**

- ✓ The process time of the current system are time consuming because the current system are using paper work to record down all the information. It is take a lot of time to serve the customer when the staff searching the available time and seat. The purpose of create this system is to improve the efficiency time of process. It can let the staff search the information faster and easier and maintain the data faster and reduce the paperwork.

❖ **Increase income**

- ✓ Made reservation using online method can attract more customers because made reservation online can save a lot of time for the customer. If can let the customer feel convenient, so that they will come to our company buy the bus ticket more often when they want go to somewhere around Malaysia so we can earn more income from them.

❖ **Bus and ticket information**

- ✓ This system can let the customer know the information about the bus schedule and ticket. Nowadays, online are very common issues to everyone so that checking information using online can save a lot of time to the customer. Customers no need

go to the counter to asking about the information of the bus schedule and the schedule.

- **Reduce error data**
  - ✓ The purpose of creating this new system is let the staff to reduce key in wrong data, because the system are using computer to key in the data, there got some validation to check when staff key in the data wrongly and system will provide some selection to let staff direct choose the data. It also reduce the paper work to record all the information, so it bring the benefit for reduce error data.

2.6 Software and Hardware Requirement for Development

2.6.1 Software Requirement

- **Microsoft SQL Server 2008**

  Microsoft SQL Server 2008 is a relational model database server produced by Microsoft. Its primary query languages are T-SQL and ANSI SQL. SQL Server 2008 was released (RTM) on August 6, 2008 and aims to make data management self-tuning, self organizing and self maintaining with the development of SQL Server 2008 Always On technologies, to provide near-zero downtime. SQL Server 2008 also includes support for structured and semi-structured data, including digital media formats for pictures, audio, video and other multimedia data. In current versions, such multimedia data can be stored as BLOBs (binary large objects), but they are generic bitstreams. Intrinsic awareness of multimedia data will allow specialized functions to be performed on them. SQL Server 2008 can be a data storage backend for different varieties of data: XML, email, time/calendar, file, document, spatial, etc as well as perform search, query, analysis, sharing, and synchronization across all data types.

- **Microsoft Office Word**

  Microsoft office word is very important software to us because we need do documentation for our project. We can using this software to key in the word and also can add in some features such as draw the organization chart, table, and the diagram.

❖ **Microsoft Visual Studio 2008**

Visual basic is a common language that used by everyone because this language very easy to learn and understand the coding. Visual Basic (VB) 2008 is the third-generation event-driven programming language and integrated development environment (IDE) from Microsoft for its COM programming model. VB is also considered a relatively easy to learn and use programming language, because of its graphical development features and basic heritage.

Visual Basic 2008 was derived from basic and enables the rapid application development (RAD) of graphical applications, access to databases using data access objects, and creation of ActiveX controls and objects. Scripting languages such as VBA and VBScript are syntactically similar to Visual Basic, but perform differently.

❖ **Web Browser**

The web browser such as Internet Explorer, Safari, Firefox, Opera, Google chrome, Maxthon, Netscape and so on are required for retrieving, presenting, and traversing information resources on the World Wide Web. Although browsers are primarily intended to access the World Wide Web, they can also be used to access information provided by web servers in private networks or files in file systems. Some browsers can be also used to save information resources to file systems.

Users need it to view our company web pages, image video or other piece of content. Hyperlinks present in resources enable users to easily navigate their browsers to related resources. Without the web browser, users cannot browse our restaurant web pages to make order. Most web browsers can display a list of web pages that the user has bookmarked so that the user can quickly return to them.

❖ **IBM Software (Rational Software Architect)**

IBM Rational Software Architect, (RSA) made by IBM's Rational Software division, is a comprehensive modeling and development environment that uses the Unified Modeling Language (UML) for designing architecture for C++ and Java 2 Enterprise Edition (J2EE) applications and web services. Rational Software Architect is built on

the Eclipse open-source software framework and includes capabilities focused on architectural code analysis, C++, and model-driven development (MDD) with the UML for creating resilient applications and web services.

- **Notepad**

Notepad is a basic text editor that users can use to create simple documents. The most common user for Notepad is to view or edit text (.txt) files.

- **Paint**

Paint is a basic picture editor that users can use to create simple picture. The most common use for Paint is to edit the picture files such .jpg, .png.

- **.Net Framework 3.5**

The .Net Framework is an integral windows component for building and running the next generation of software applications and Web services. The .Net Framework Supports over 20 different programming languages so that makes it easier than ever before to build, deploy, and administer secure, robust, and high-performing applications and manages much of the plumbing involved in developing software, enabling developers to focus on the core business logic code. Besides that, the .Net Framework is composed of the common language runtime and a unified set of class libraries.

- **Printer driver**

In computers, a printer driver or a print processor is a piece of software that converts the data to be printed to the form specific to a printer. The purpose of printer drivers is to allow applications to do printing without being aware of the technical details of each printer model.

2.6.2 Hardware Requirement

- **Printer**

  Printer is a hardware that can produce the hard copy of the documentation which store in the electronic form. We need using printer is because we want print out the form of reservation to the staff and counter want print out the ticket for the customer when they come to collect their ticket. We also need print out the report that we already some analysis to the high level management.

- **Laptop**

  Laptop is a personal computer designed for mobile use and small and lights enough to sit on a person's lap while in use. Laptop can bring a lot of convenient to us such as can bring to anywhere and easy so we need using the laptop to do our project to fulfill the requirement and complete it.

- **Stationery**

  Stationery like ink, paper, pen and so on that were need use because we using printer so need ink for refill when the cartridge already out of ink. This all is the necessary thing that we need to do and use.

2.7 Software and Hardware Requirement for Operational

2.7.1 Software Requirement

- **Printer driver**

  In computers, a printer driver or a print processor is a piece of software that converts the data to be printed to the form specific to a printer. The purpose of printer drivers is to allow applications to do printing without being aware of the technical details of each printer model.

- **Operating system**

  An operating system (OS) is a software program that enables the computer hardware to communicate and operate with the computer software. The operating systems that

can use are Windows (Windows 98, XP, Vista, 7), Macintosh OS X, the many versions of Linux and Unix, i5/OS (IBM iSeries) and z/OS (IBM zSeries mainframes) and others. It is responsible for management and coordination of processes and allocation and sharing of hardware resources and can let the computer hardware to communicate and operate with the software.

- **Web Browser**

The web browser such as Internet Explorer, Safari, Firefox, Opera, Google chrome, Maxthon, Netscape and so on are required for retrieving, presenting, and traversing information resources on the World Wide Web. Although browsers are primarily intended to access the World Wide Web, they can also be used to access information provided by web servers in private networks or files in file systems. Some browsers can be also used to save information resources to file systems.

Users need it to view our company web pages, image video or other piece of content. Hyperlinks present in resources enable users to easily navigate their browsers to related resources. Without the web browser, users cannot browse our restaurant web pages to make order. Most web browsers can display a list of web pages that the user has bookmarked so that the user can quickly return to them.

- **DBMS**

A **Database Management System** (**DBMS**) is a set of computer software programs that is designed as the means of managing all databases that are currently installed on a system hard drive or network. It controls the creation, maintenance, and the use of the database with computer as a platform or of an organization and its end users. Users must required DBMS to store or record the data or very transaction like history in their computer.

This system is can employees accessed and easily to search for records with certain qualities. For this database it have store for more types database that is bus database, customers database and ticket order database. A database of customers can be gathered to determine which customers should be contacted for promotion.

In addition, a database of ticket order it is to record all the order taking by customer and up to dates to following the record and is record or save the order information, which is already sold out.

2.7.2 Hardware Requirement

- **Printer**

Printer is an output device that produces the hard copy of documentation. We want using this hardware because we want staff print out the ticket for the customer when the customer come to the counter collect the ticket. Until end of month, the staff also want do some analysis to produce the report for the high level management so the high level management can do so improvement for the performance so the staff need using printer to print out the report.

- **Laptop**

Laptop is a personal computer designed for mobile use and small and lights enough to sit on a person's lap while in use. Laptop can bring a lot of convenient to us such as can bring to anywhere and easy so we need using the laptop to do our project to fulfill the requirement and complete it.

2.8 System Architecture Diagram

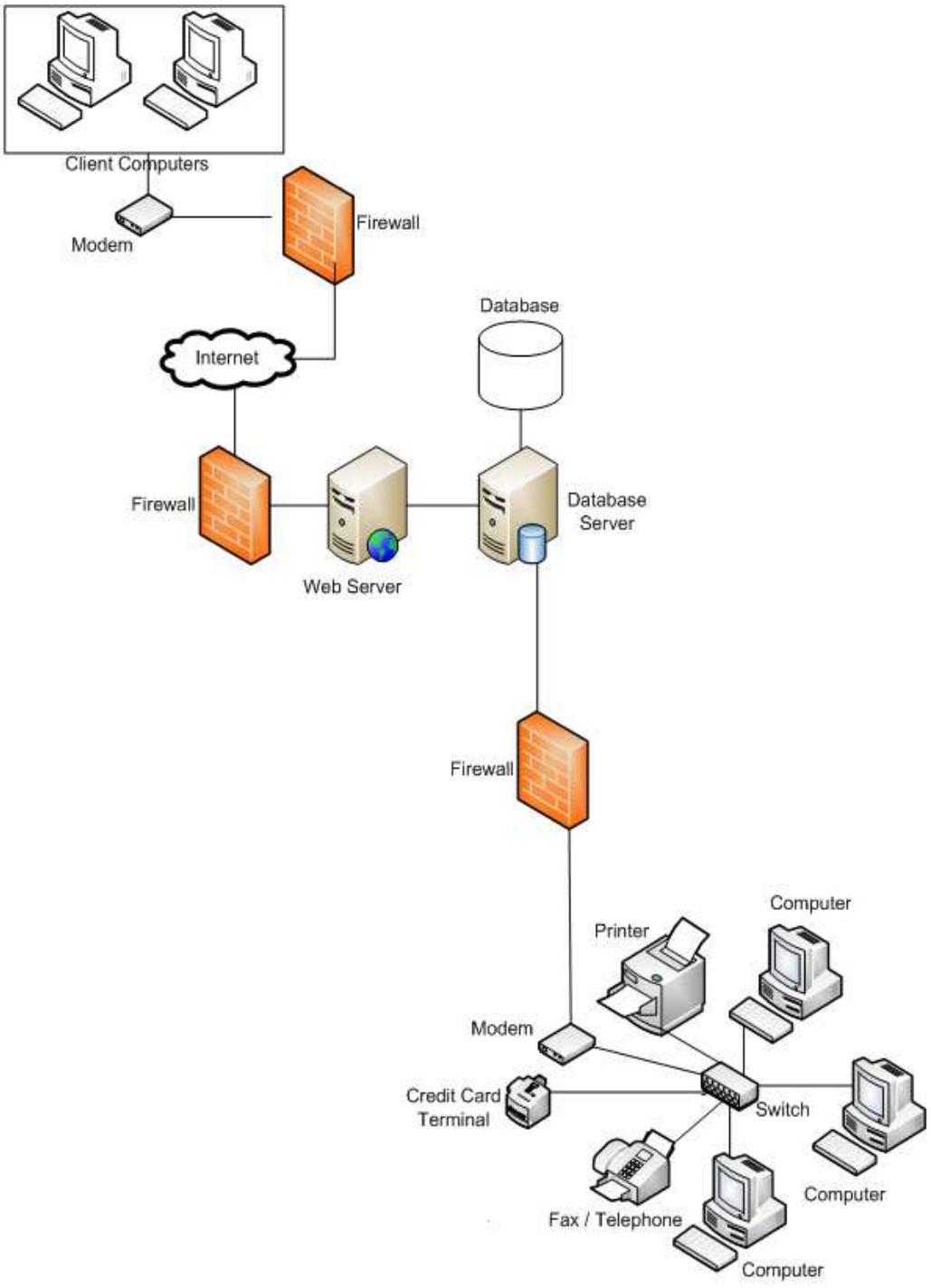


Figure 2.7.1 System Architecture Design

2.9 Chapter Summary

This chapter has listed out the minimum requirement need to build this project and those methods that used to gather the information from the different way. This is to avoid any unnecessary compatibility issues.

After go through the feasibility study, we are strongly recommend to the management so that they can invest and develop a new bus ticketing system because this new system will bring benefits to the company as well as the customers. After perform all the costs and benefits analysis and calculation such as payback analysis, Return on Investment (ROI) and the Net Present Value (NPV), so that we can conclude that the new system will bring the profits and benefits to the company.

Functional requirements are listed out the functions that is provided in this project and who are the person going to carry out the function.

Chapter 3

System Design

3 System Design

3.1 Introduction

In this chapter will discuss about the design of the system which is the use case, activity, and sequence diagram, class diagram, database design, data specification for each tables, user interface design and how the interface act to perform the functions.

3.2 Use Case Diagram for Proposed Bus Ticketing System

Overview Use Case Diagram

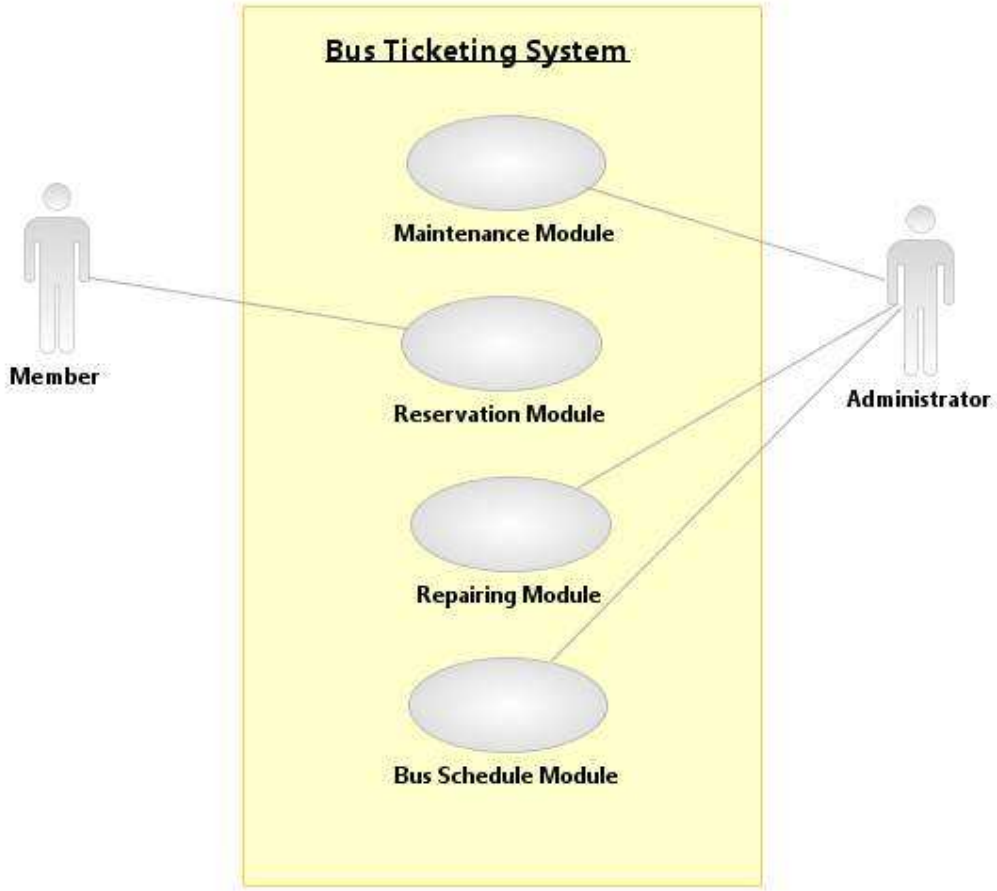


Figure 3.2.1 Overview Use Case Diagram for Bus Ticketing System

Use Case Diagram for Maintenance Module

Staff Maintenance

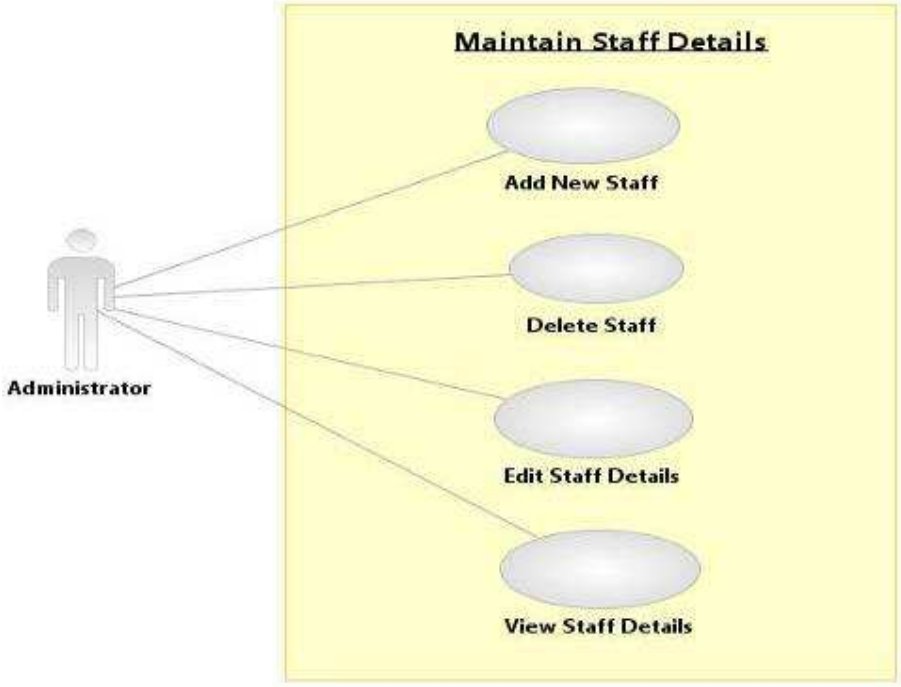


Figure 3.2.2 Use Case Diagram for Staff Maintenance

Use Case Description

Use Case Name: Add New Staff	
Brief Description: This use case is allow the administrator to add new staff for company	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Maintenance Menu
2. Click “Update Staff Information” Button	
	3. Display Staff Maintenance Form
4. Click “Add” Button	
5. Key In Staff IC	
6. Click “Check” Button	
	7. Verify Staff IC

8. Enter Staff Login Details	
9. Click “Check” Button	
	10. Verify Staff Login Details
11. Enter Staff Details	
12. Click “Save” Button	
	13. Verify Staff Details
	14. Save record into database
	15. Show Message Box “Record Added”
16. Click “Ok” Button	
Alternative Flows:	
A-7 Step 5: If the staff IC exist in the database - System prompt “Invalid Applicant”	
A-10 Step 8: If staff login details not correct - System prompt “Please key in the correct data”	
A-13 Step 11: If the staff details does not fill in and the wrong data - System prompt “Please enter correct data in the fields”	

Use Case Name: Edit Staff Details	
Brief Description: This use case is allow the administrator to edit the staff details	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Maintenance Menu
2. Click “Update Staff Information” Button	
	3. Display Staff Maintenance Form
4. Select Staff ID	
5. Click “Search” Button	
	6. Display Staff Details
7. Click “Edit” Button	
8. Change Staff Details	
9. Click “Save” Button	
	10. Verify Staff Details

	11. Save record into database
	12. Show Message Box “Record Updated”
13. Click “Ok” Button	
Alternative Flows:	
A-10 Step 8: If have empty field and the data not correct - System prompt “The field cannot leave it blank and key in the correct data”	

Use Case Name: Delete Staff	
Brief Description: This use case is allow the administrator to delete those staff already resign	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Maintenance Menu
2. Click “Update Staff Information” Button	
	3. Display Staff Maintenance Form
4. Select Staff ID	
5. Click “Search” Button	
	6. Display Staff Details
7. Click “Delete” Button	
	8. Show Message Box “Confirm Delete Staff”
9. Click “Ok” Button	
	10. Delete Record from database
	11. Show Message Box “Record Deleted”
12. Click “Ok” Button	
Alternative Flows:	
A-9 Step 10: If user click cancel button - Record remain in database	

Use Case Name: View Staff Details
Brief Description: This use case is let the administrator to view the staff details
Actor: Administrator

Main Flow:	
Actor Action	**System Response**
	1. Display Maintenance Menu
2. Click “Update Staff Information” Button	
	3. Display Staff Maintenance Form
4. Select Staff ID	
5. Click “Search” Button	
	6. Display Staff Details

Member Maintenance

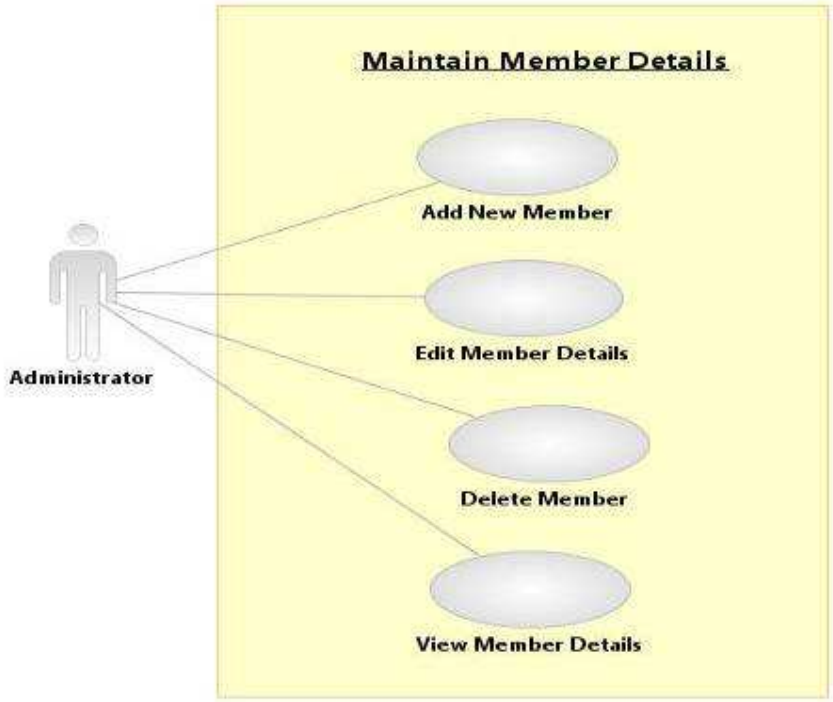


Figure 3.2.3 Use Case Diagram for Member Maintenance

Use Case Description

Use Case Name: Add New Member	
Brief Description: This use case is allow the administrator to add new member for company	
Actor: Administrator	
Main Flow:	

Actor Action	**System Response**
	1. Display Maintenance Menu
2. Click "Update Member Information" Button	
	3. Display Member Maintenance Form
4. Click "Add" Button	
5. Key In Member IC	
6. Click "Check" Button	
	7. Verify Member IC
8. Enter Member Login Details	
9. Click "Check" Button	
	10. Verify Member Login Details
11. Enter Member Details	
12. Click "Save" Button	
	13. Verify Member Details
	14. Save record into database
	15. Show Message Box "Record Added"
16. Click "Ok" Button	
Alternative Flows:	
A-7 Step 5: If the member IC exist in the database - System prompt "Invalid Applicant"	
A-10 Step 8: If member login details not correct - System prompt "Please key in the correct data"	
A-13 Step 11: If the member details does not fill in and the wrong data - System prompt "Please enter correct data in the fields"	

Use Case Name: Edit Member Details	
Brief Description: This use case is allow the administrator to edit the member details	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Maintenance Menu
2. Click "Update Member Information"	

Button	
	3. Display Member Maintenance Form
4. Select Member ID	
5. Click “Search” Button	
	6. Display Member Details
7. Click “Edit” Button	
8. Change Member Details	
9. Click “Save” Button	
	10. Verify Member Details
	11. Save record into database
	12. Show Message Box “Record Updated”
13. Click “Ok” Button	
Alternative Flows:	
A-10 Step 8: If the data incorrect and blank data - System prompt “ Please key in the correct data and data cannot leave it blank”	

Use Case Name: Delete Member	
Brief Description: This use case allow the administrator to delete member	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Maintenance Menu
2. Click “Update Member Information” Button	
	3. Display Member Maintenance Form
4. Select Member ID	
5. Click “Search” Button	
	6. Display Member Details
7. Click “Delete” Button	
	8. Show Message Box “Confirm Delete Member”
9. Click “Ok” Button	

	10. Delete Record from database
	11. Show Message Box “Record Deleted”
12. Click “Ok” Button	
Alternative Flows:	
A-9 Step 10: If the user click cancel button - Record remain in the database	

Use Case Name: View Member Details	
Brief Description: This use case is allow the administrator to view the member details	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Maintenance Menu
2. Click “Update Member Information” Button	
	3. Display Member Maintenance Form
4. Select Member ID	
5. Click “Search” Button	
	6. Display Member Details

Bus Maintenance

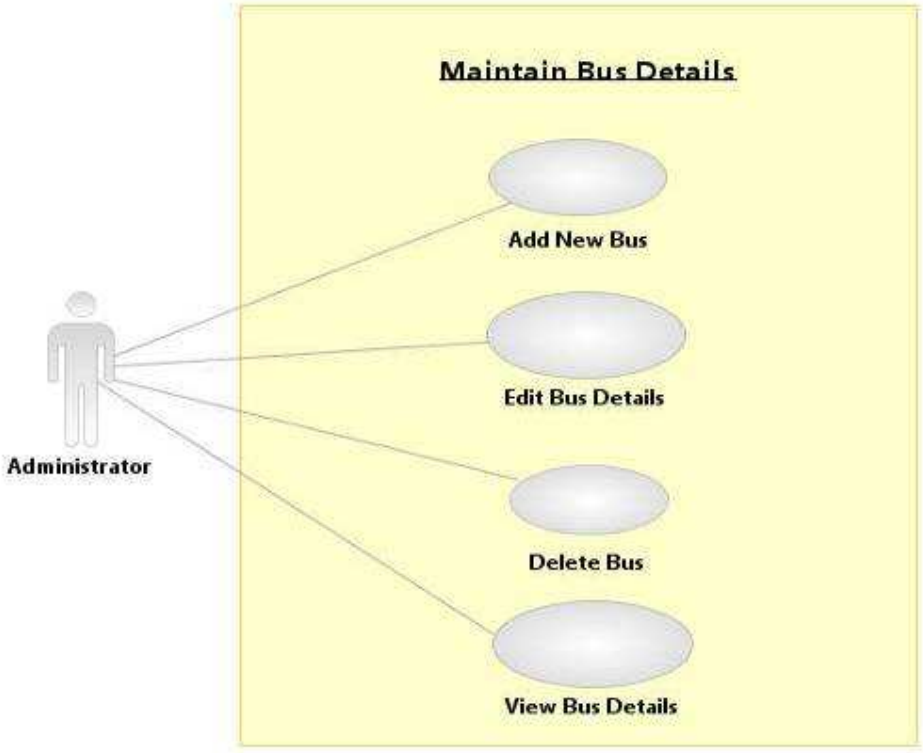


Figure 3.2.4 Use Case Diagram for Bus Maintenance

Use Case Description

Use Case Name: Add New Bus	
Brief Description: This use case allow the administrator to add the new bus for company	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Maintenance Menu
2. Click “Update Bus Information” Button	
	3. Display Bus Maintenance Form
4. Click “Add” Button	
5. Key in Bus Number	
6. Click “Check” Button	
	7. Verify bus number

8. Enter Bus Details	
9. Click “Save” Button	
	10. Verify Bus Details
	11. Save record into database
	12. Show Message Box “Record Added”
13. Click “Ok” Button	
Alternative Flows:	
A-7 Step 5: If the bus number exist in the database - System prompt “Invalid Bus Number”	
A-10 Step 8: If bus details not correct and have blank data - System prompt “Please key in the correct data and fill in the blank data”	

Use Case Name: Edit Bus Details	
Brief Description: This use case allow administrator to edit bus details	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Maintenance Menu
2. Click “Update Bus Information” Button	
	3. Display Bus Maintenance Form
4. Select Bus ID	
5. Click “Search” Button	
	6. Display Bus Details
7. Click “Edit” Button	
8. Change Bus Details	
9. Click “Save” Button	
	10. Verify Bus Details
	11. Save record into database
	12. Show Message Box “Record Updated”
13. Click “Ok” Button	
Alternative Flows:	
A-10 Step 8: If bus details incorrect and have blank data	

- System prompt “Please key in the correct data and fill in the blank data

Use Case Name: Delete Bus	
Brief Description: This use case allow the administrator to delete bus for company	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Maintenance Menu
2. Click “Update Bus Information” Button	
	3. Display Bus Maintenance Form
4. Select Bus ID	
5. Click “Search” Button	
	6. Display Bus Details
7. Click “Delete” Button	
	8. Show Message Box “Confirm Delete Bus”
9. Click “Ok” Button	
	10. Delete Record from database
	11. Show Message Box “Record Deleted”
12. Click “Ok” Button	
Alternative Flows:	
A-9 Step 10: If the user click cancel button - Record remain in the database	

Use Case Name: View Bus Details	
Brief Description: This use case allow the administrator to view the bus details	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Maintenance Menu
2. Click “Update Bus Information” Button	
	3. Display Bus Maintenance Form

4. Select Bus ID	
5. Click "Search" Button	
	6. Display Bus Details

Use Case Diagram for Reservation Module

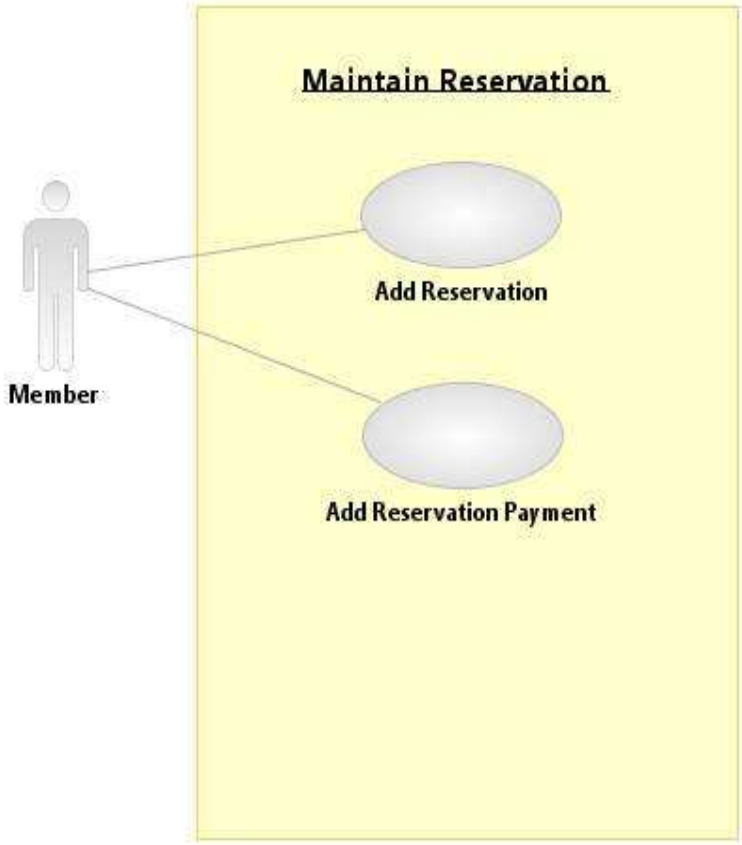


Figure 3.2.5 Use Case Diagram for Reservation

Use Case Description

Use Case Name: Add Reservation, Add Reservation Payment	
Brief Description: This use case allow the member to add reservation and reservation payment	
Actor: Member	
Main Flow:	
Actor Action	**System Response**

	1. Display Reservation Form
2. Select Member ID	
3. Select Departure Destination	
	4. Display Arrival Destination
5. Select Arrival Destination	
6. Select Date	
7. Click “Search” Button	
	8. Verify Schedule Details
	9. Display Schedule Details
10. Select a time	
11. Click “Next” Button	
	12. Display Seat
13. Select Seat Number	
14. Click “Next” Button	
	15. Calculate Total Amount
	16. Display Payment Form
17. Key in Payment Details	
18. Click “Paid” Button	
	19. Verify Payment Details
	20. Save record into database
	21. Show Message “Record Added”
	22. Display Reservation Confirmation
Alternative Flows:	
A-8 Step 3: If the destination do not have schedule - System prompt “This Destination do not any Schedule”	
A-19 Step 17: If the payment details incorrect and have blank value - System prompt “Please key in the correct data and fill in the blank value”	

Use Case Diagram for Bus Scheduling Module

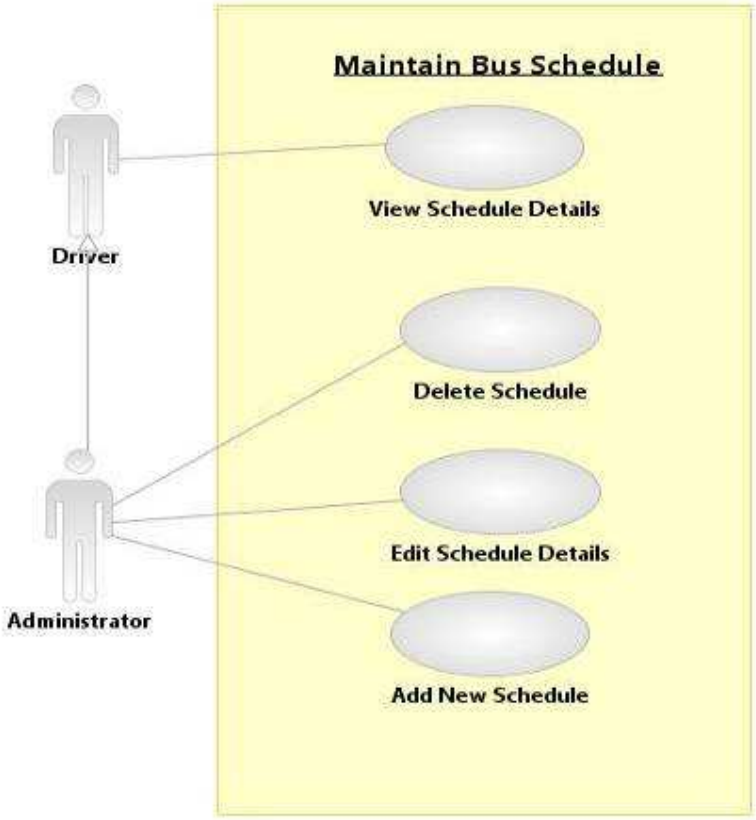


Figure 3.2.6 Use Case Diagram for Scheduling

Use Case Description

Use Case Name: View Schedule Details	
Brief Description: This use case allow the driver to view their schedule	
Actor: Driver	
Main Flow:	
Actor Action	**System Response**
	1. Display Scheduling Menu
2. Click "View Schedule Details" Button	
	3. Display View Schedule Form
4. Select Driver Name	
5. Click "Search" Button	

	6. Display Schedule Details

Use Case Name: Add New Schedule	
Brief Description: This use case is let administrator to add the new schedule for the driver	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Scheduling Menu
2. Click “Assign Schedule” Button	
	3. Display Scheduling Form
4. Click “Add” Button	
5. Key in New Destination	
6. Click “Check” Button	
	7. Verify New Destination
8. Enter Destination Details	
9. Enter Schedule Details	
10. Click “Save” Button	
	11. Verify Schedule and Destination Details
	12. Save record into database
	13. Show Message Box “Record Added”
14. Click “Ok” Button	
Alternative Flows:	
A-7 Step 5: If the destination exist in the database - System prompt “Invalid Destination”	
A-11 Step 8: If destination details and schedule details incorrect - System prompt “Please key in the correct data”	
A-11 Step 9: If the schedule details exist in the database - System prompt “This Schedule already exist in the database”	

Use Case Name: Edit Schedule Details
Brief Description: This use case is let administrator to edit the schedule details
Actor: Administrator

Main Flow:	
Actor Action	**System Response**
	1. Display Scheduling Menu
2. Click “Assign Schedule” Button	
	3. Display Scheduling Form
4. Select Package ID	
5. Click “Search” Button	
	6. Display Schedule Details
7. Click “Edit” Button	
8. Change Schedule Details	
9. Click “Save” Button	
	10. Verify Schedule Details
	11. Save record into database
	12. Show Message Box “Record Updated”
13. Click “Ok” Button	
Alternative Flows:	
A-10 Step 8: If the data incorrect and have blank data - System prompt “Please key in the correct data and fill in the blank data”	

Use Case Name: Delete Schedule	
Brief Description: This use case is let administrator to delete the schedule for driver	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Scheduling Menu
2. Click “Assign Schedule” Button	
	3. Display Scheduling Form
4. Select Package ID	
5. Click “Search” Button	
	6. Display Schedule Details
7. Click “Delete” Button	
	8. Show Message Box “Confirm Delete

	Schedule"
9. Click "Ok" Button	
	10. Delete Record from database
	11. Show Message Box "Record Deleted"
12. Click "Ok" Button	
Alternative Flows:	
A-9 Step 10: If the user click cancel button - Record remain in the database	

Use Case Diagram for Repairing Module

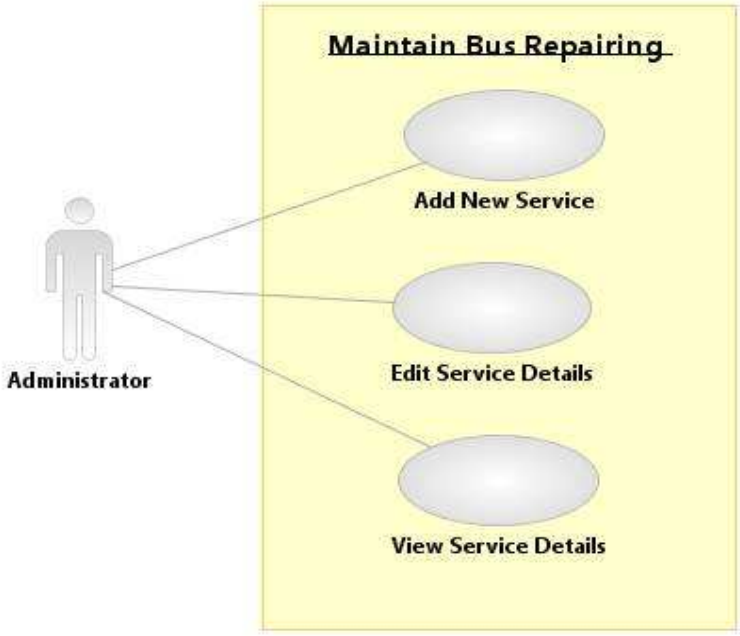


Figure 3.2.7 Use Case Diagram for Repairing

Use Case Description

Use Case Name: Add New Service	
Brief Description: This use case is let the administrator to add new service for the bus	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**

	1. Display Maintenance Menu
2. Click “Update Bus Information” Button	
	3. Display Bus Maintenance Form
4. Select Bus ID	
5. Click “Search” Button	
	6. Display Bus Details
7. Click “Send For Service” Button	
	8. Verify Bus Details
	9. Display Service Information Form
10. Select a date	
11. Select a Company Name	
	12. Display Service Company Details
13. Click “Save” Button	
	14. Verify Service Details
	15. Save record into database
	16. Show Message Box “Record Added”
17. Click “Ok” Button	
Alternative Flows:	
A-8 Step 4: If the bus details do not have driver name - System prompt “Only those bus that have driver name just can send for service”	
A-14 Step 10: If the data incorrect or have blank data - System prompt “Please key in the correct data and fill in the blank data”	

Use Case Name: Edit Service Details	
Brief Description: This use case is let the administrator to edit the service details	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Repairing Menu
2. Click “Update Service Details” Button	
	3. Display Service Information Form
4. Select Service ID	

5. Click “Search” Button	
	6. Display Service Details
7. Click “Edit” Button	
8. Change Service Details	
9. Click “Save” Button	
	10. Verify Service Details
	11. Save record into database
	12. Show Message Box “Record Updated”
13. Click “Ok” Button	
Alternative Flows:	
A-10 Step 8: If the data incorrect or have blank data - System prompt “Please key in the correct data and fill in the blank data”	

Use Case Name: View Service Details	
Brief Description:	
Actor: Administrator	
Main Flow:	
Actor Action	**System Response**
	1. Display Repairing Menu
2. Click “Update Service Details” Button	
	3. Display Service Information Form
4. Select Service ID	
5. Click “Search” Button	
	6. Display Service Details

3.3 Activity Diagram for Proposed Bus Ticketing System

Activity Diagram for Maintenance Module

Staff Maintenance

Add New Staff

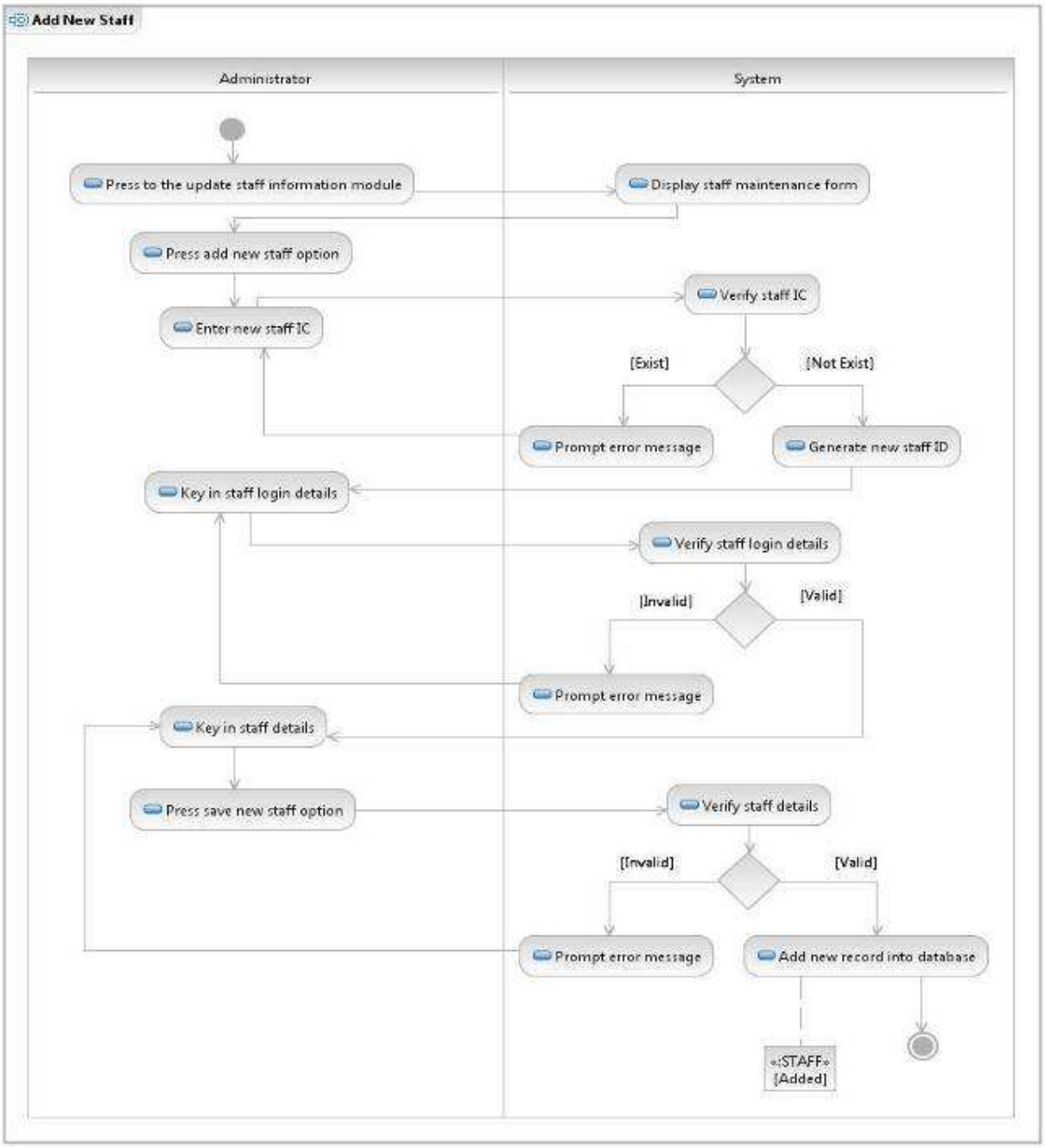


Figure 3.3.1 Activity Diagram for Add New Staff

Edit Staff Details

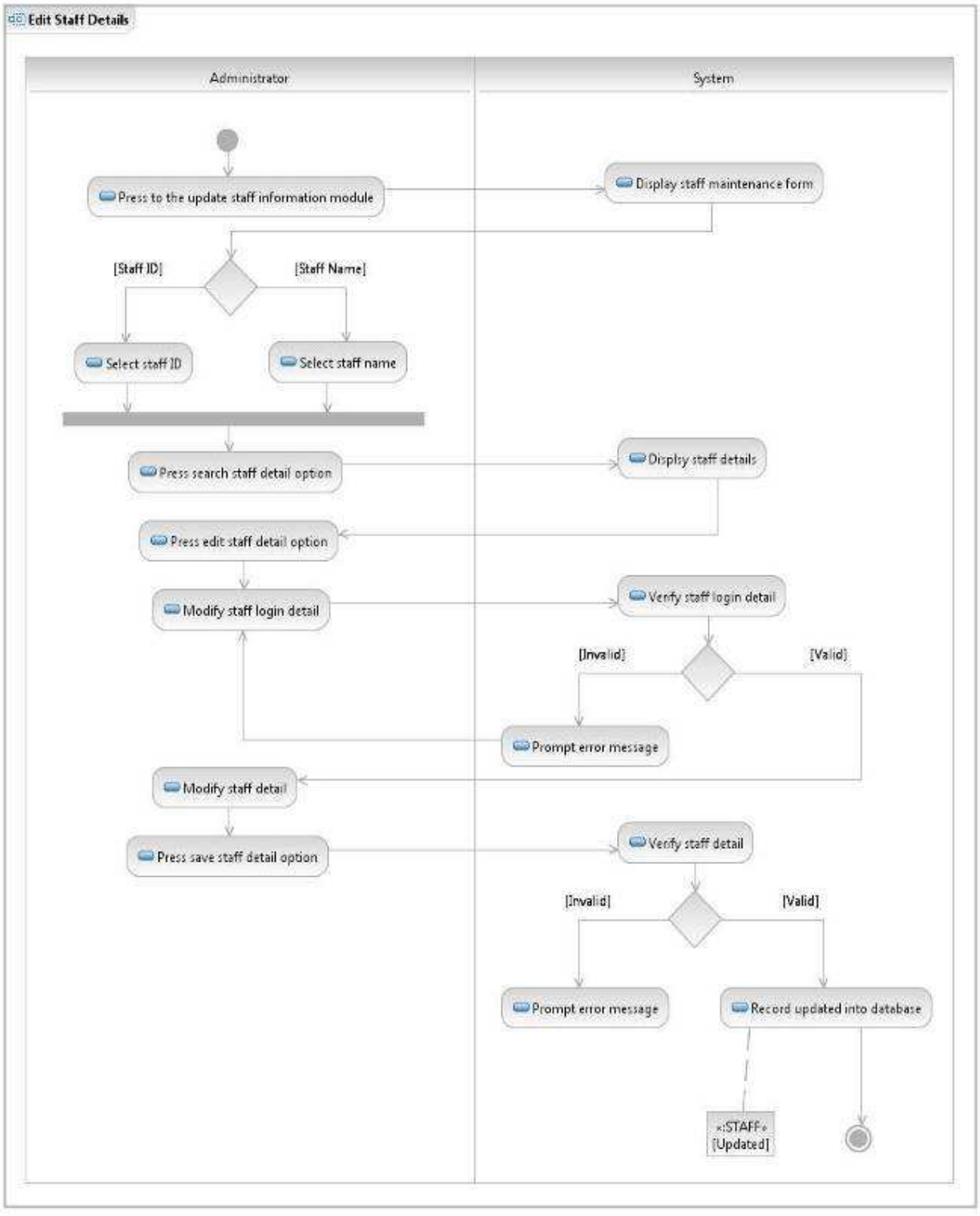


Figure 3.3.2 Activity Diagram for Edit Staff Details

Delete Staff

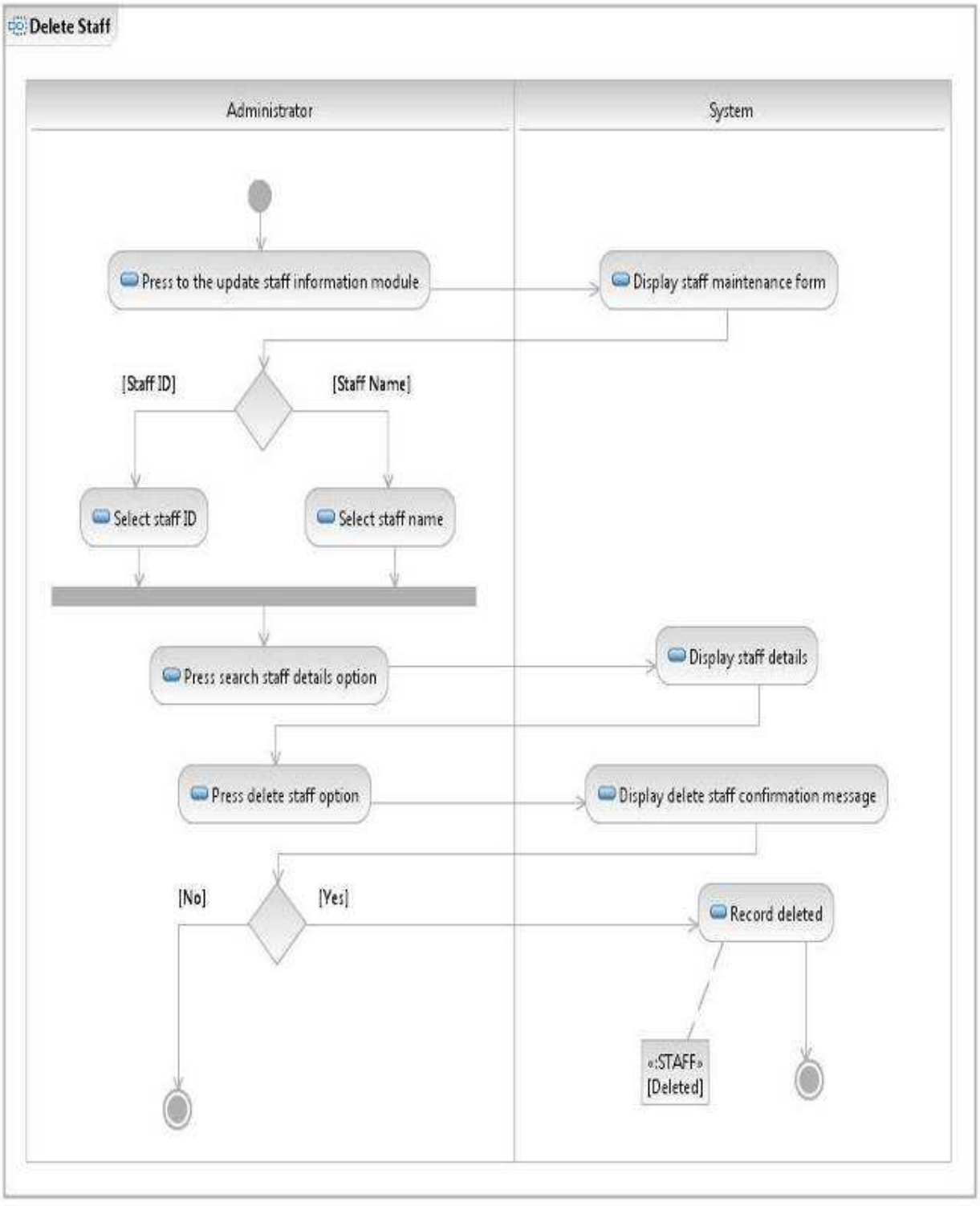


Figure 3.3.3 Activity Diagram for Delete Staff

View Staff Details

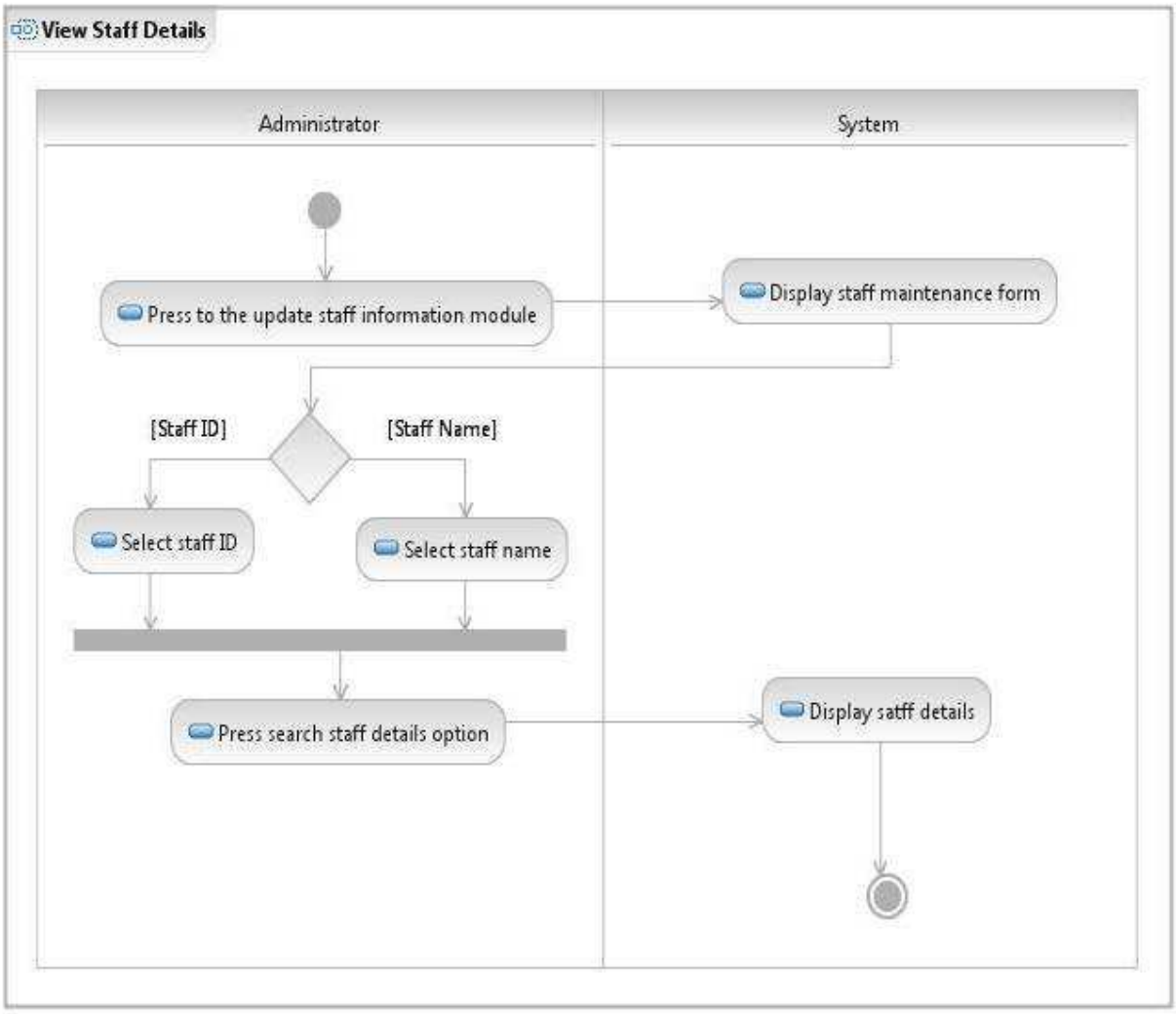


Figure 3.3.4 Activity Diagram for View Staff Details

Member Maintenance

Add New Member

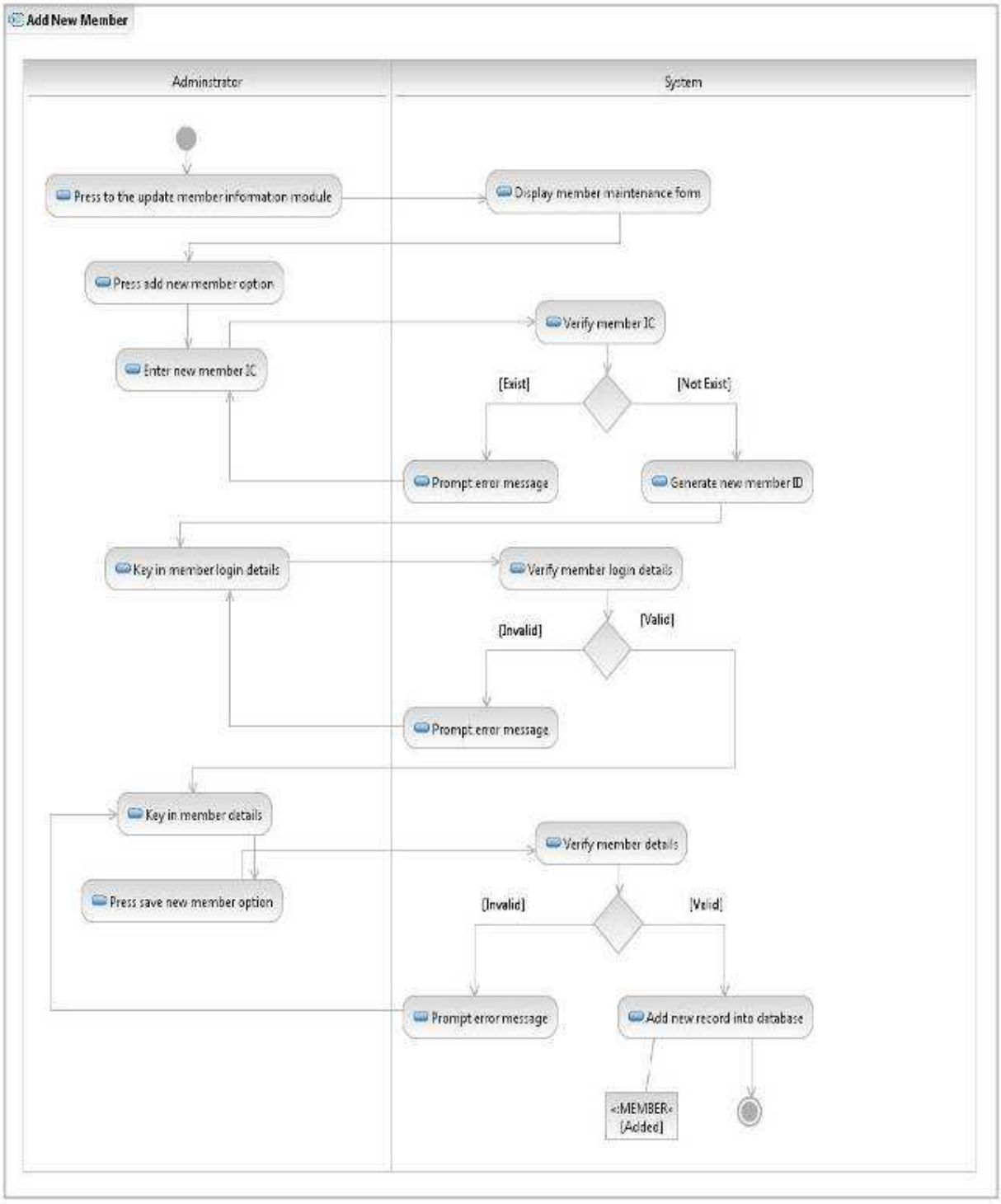


Figure 3.3.5 Activity Diagram for Add New Member

Edit Member Details

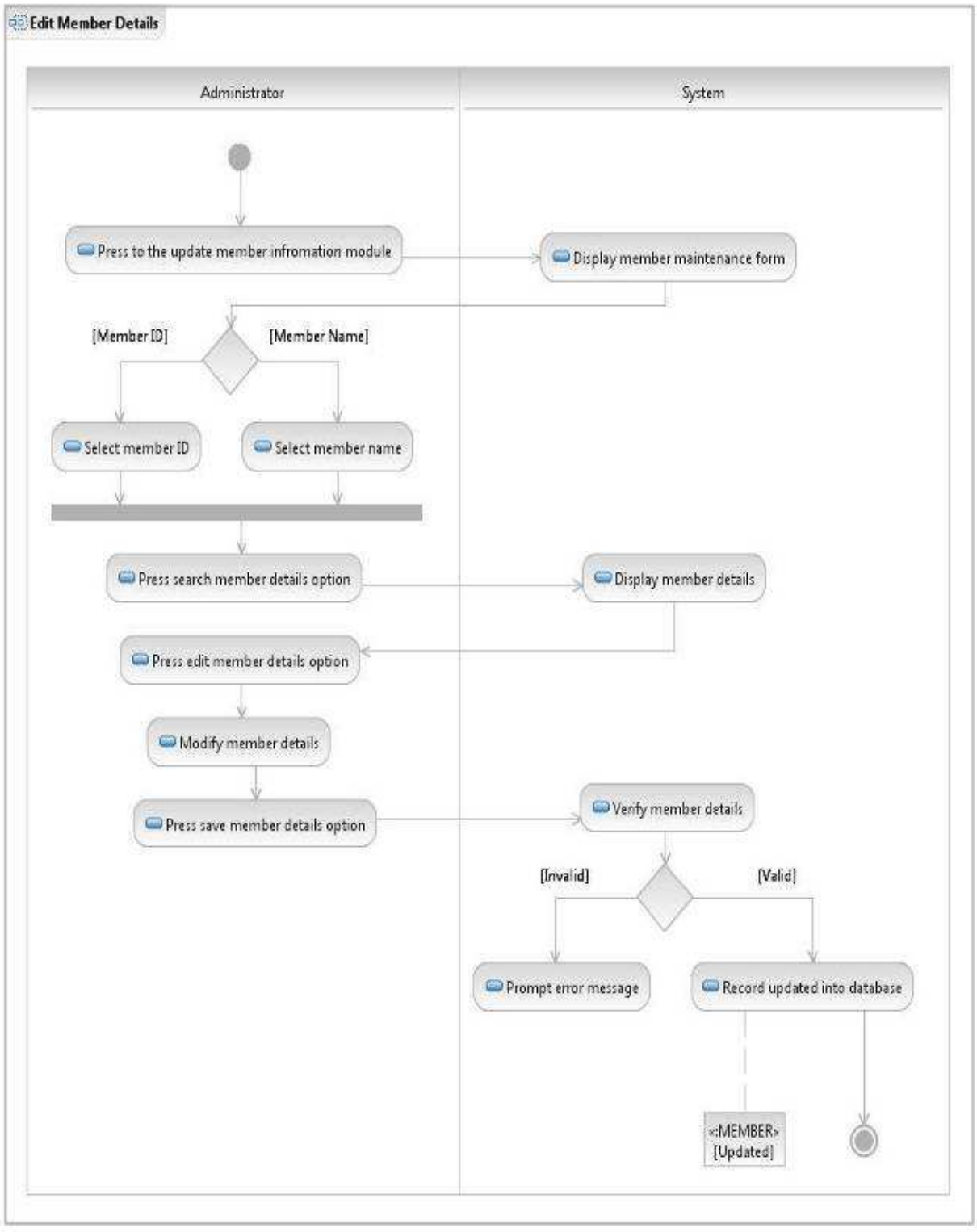


Figure 3.3.6 Activity Diagram for Edit Member Details

Delete Member

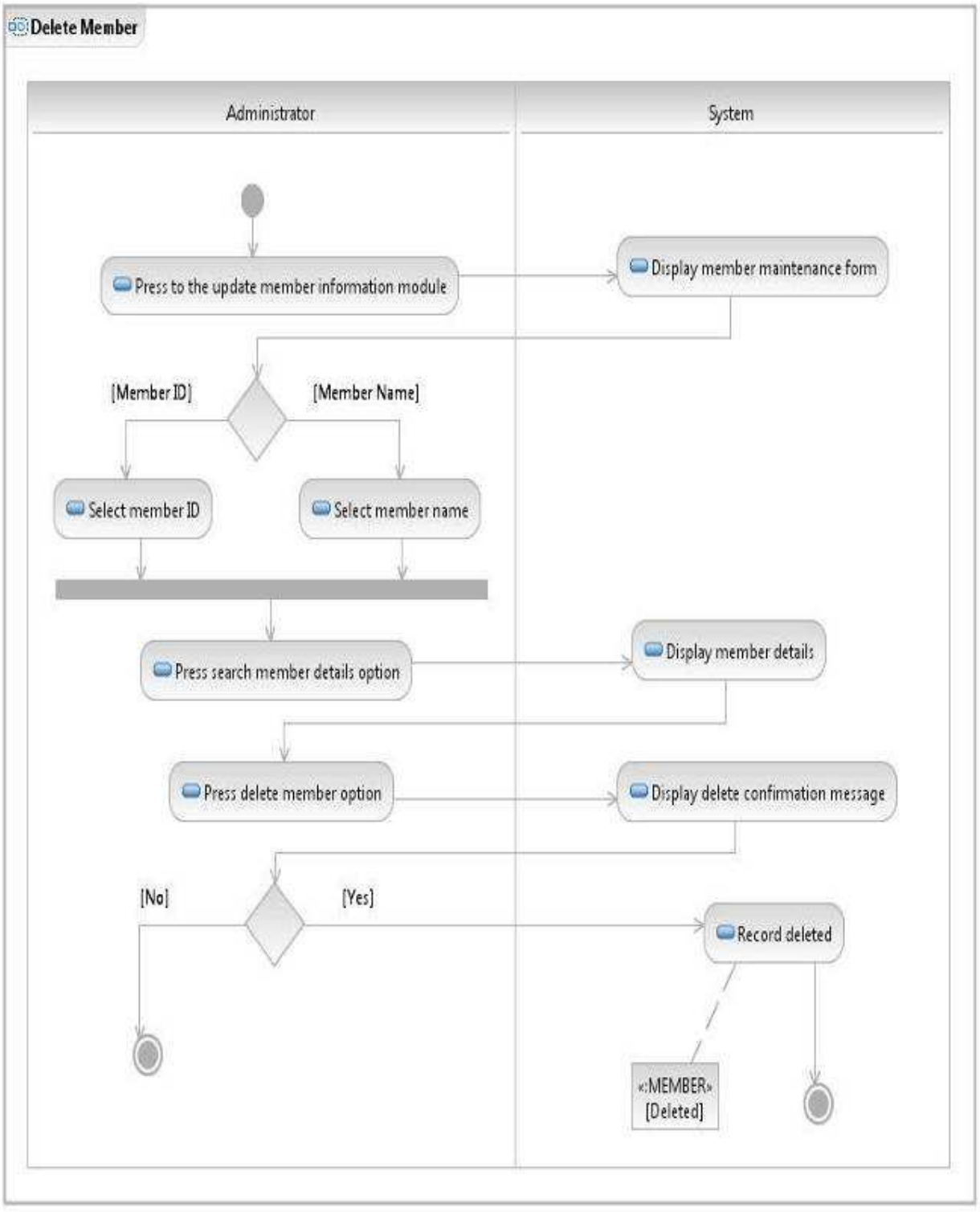


Figure 3.3.7 Activity Diagram for Delete Member

View Member Details

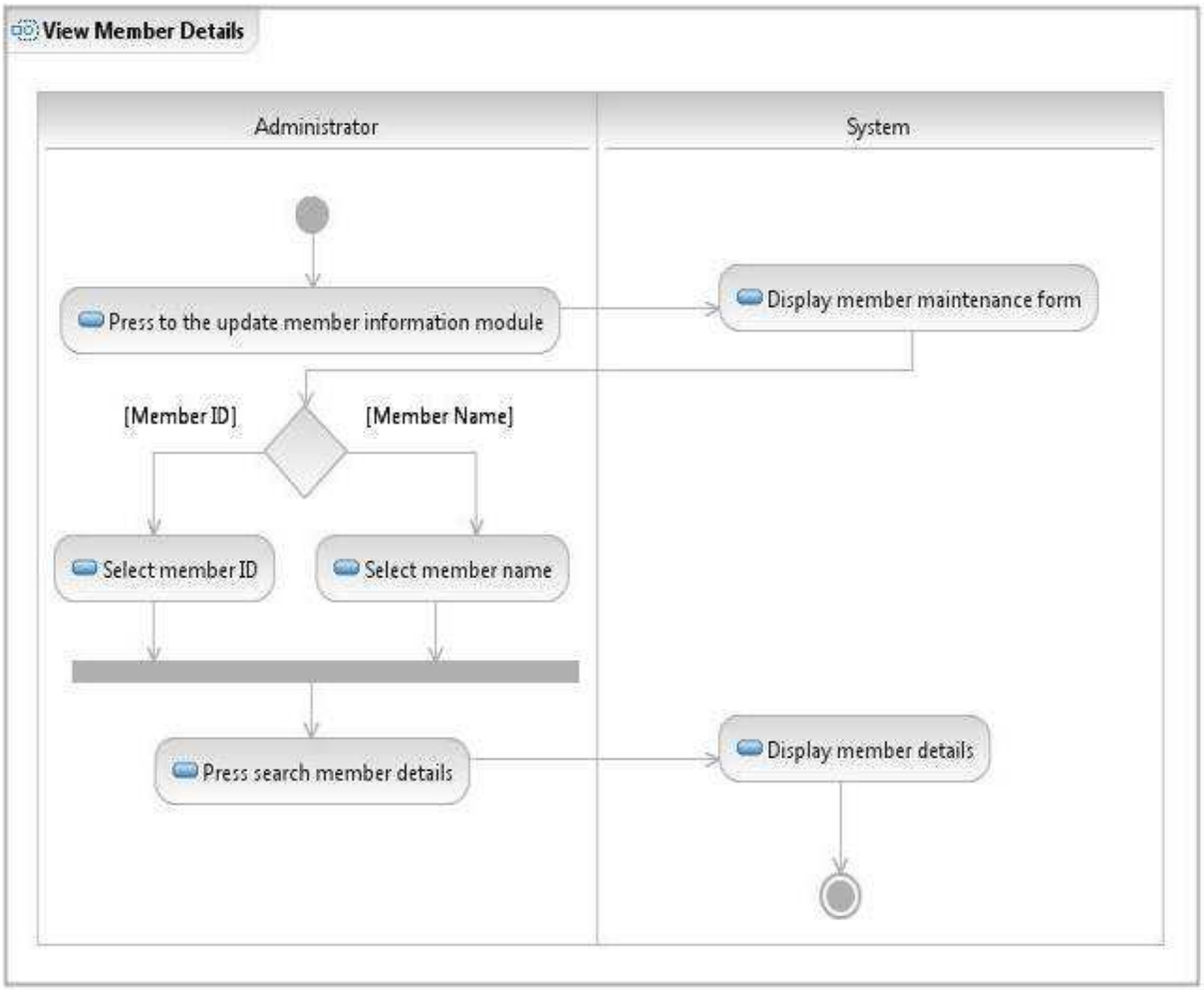


Figure 3.3.8 Activity Diagram for View Member Details

Bus Maintenance

Add New Bus

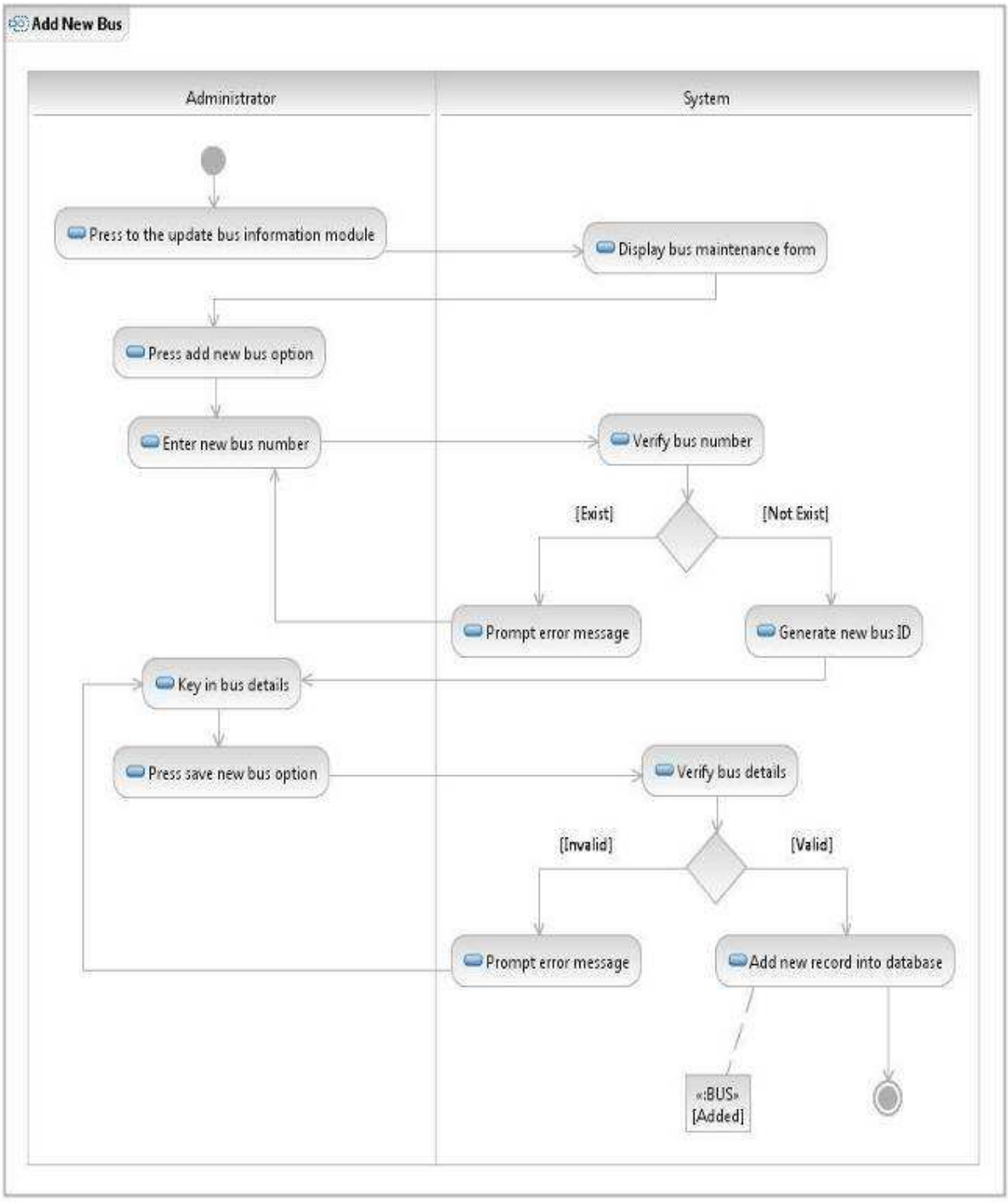


Figure 3.3.9 Activity Diagram for Add New Bus

Edit Bus Details

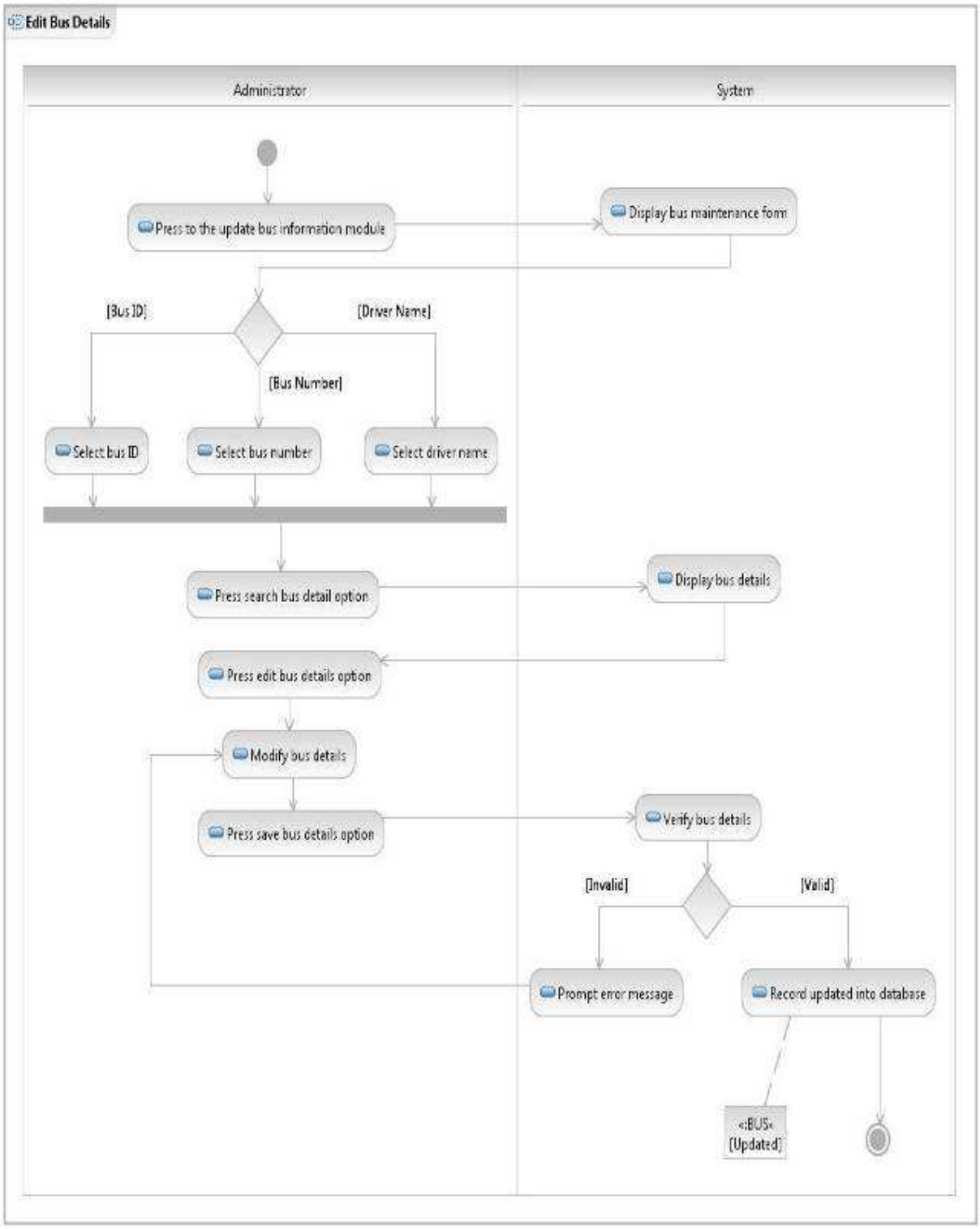


Figure 3.3.10 Activity Diagram for Edit Bus Details

Delete Bus

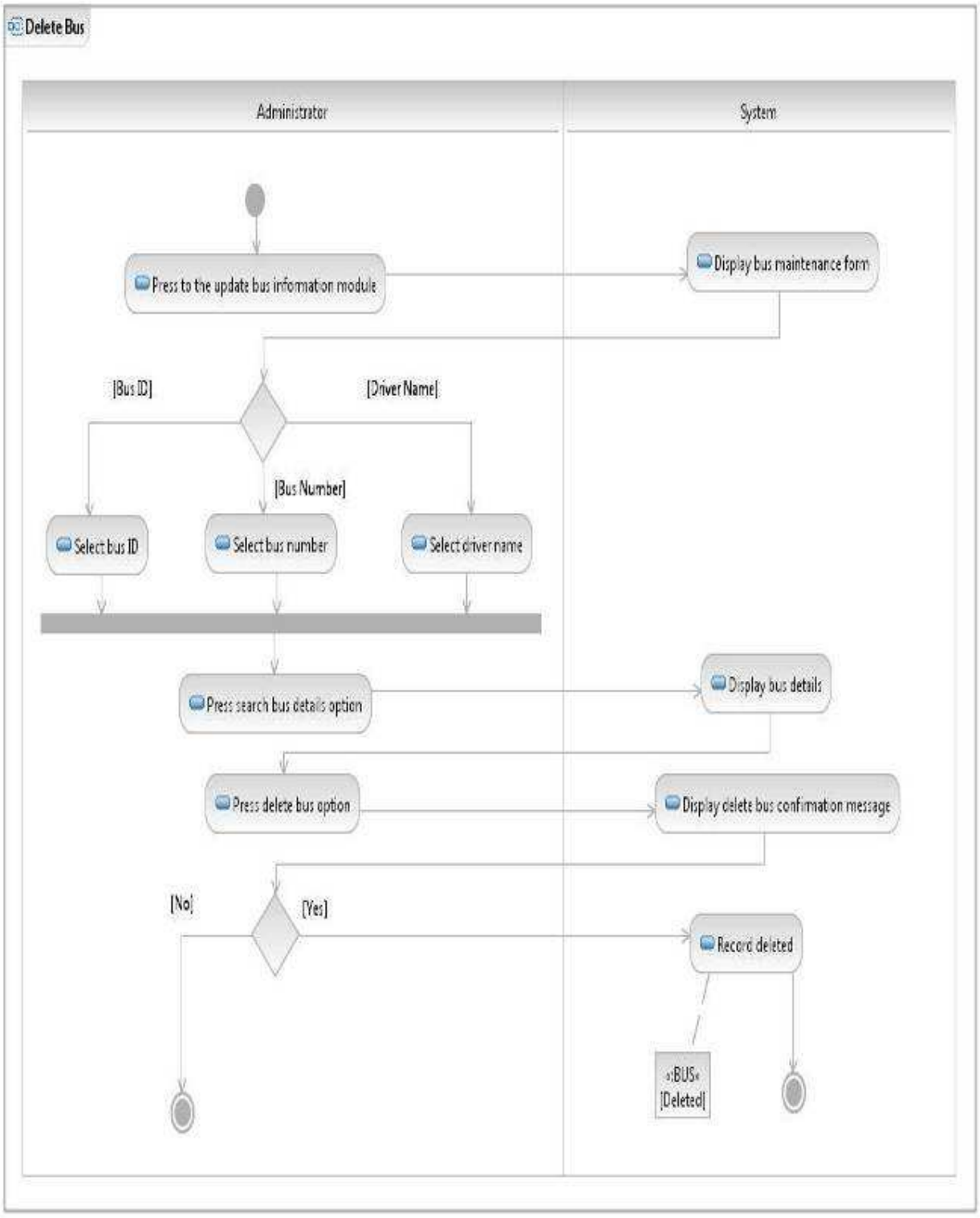


Figure 3.3.11 Activity Diagram for Delete Bus

View Bus Details

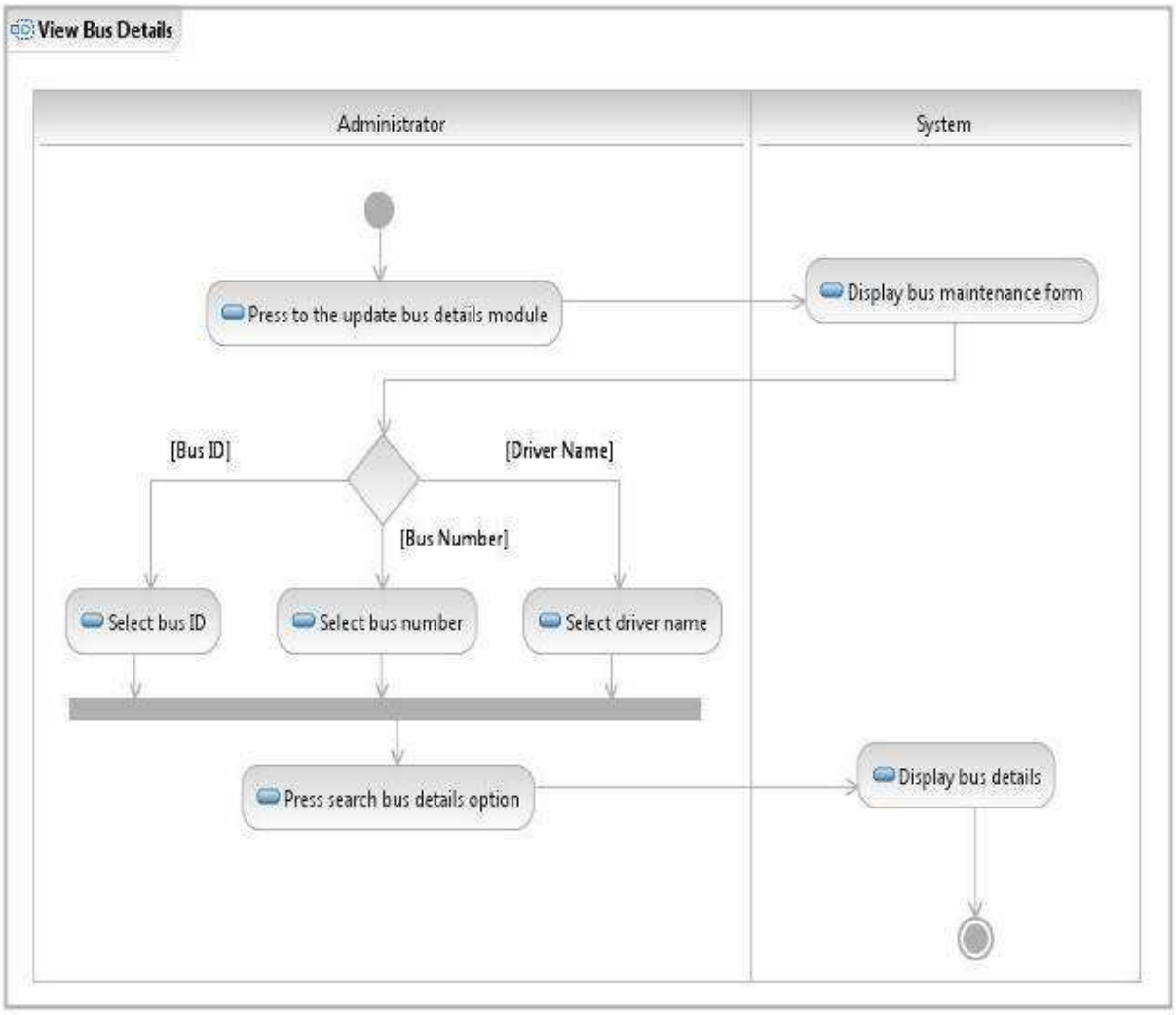


Figure 3.3.12 Activity Diagram for View Bus Details

Activity Diagram for Reservation Module

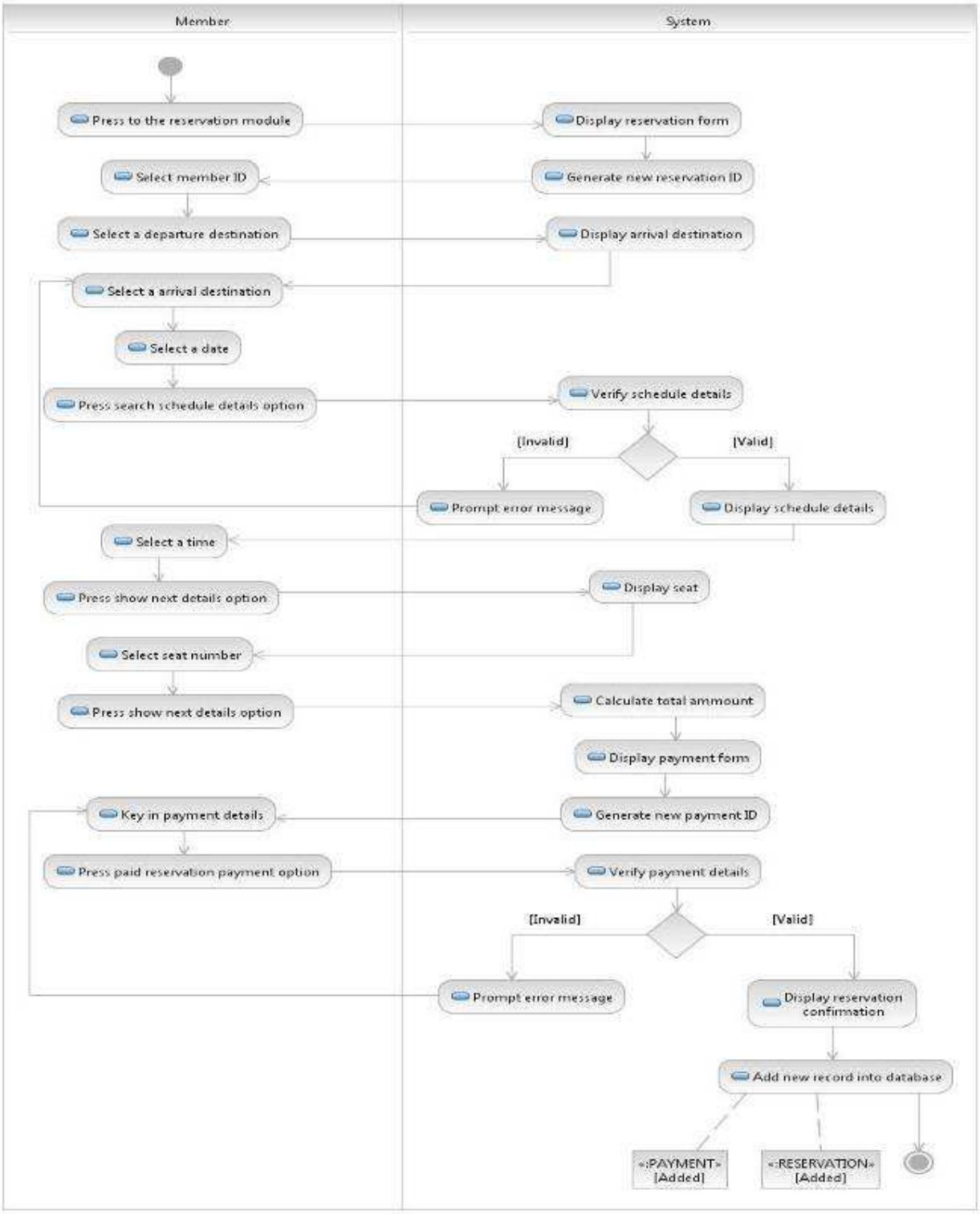


Figure 3.3.13 Activity Diagram for Add Reservation and Reservation Payment

Activity Diagram for Bus Scheduling Module

Add New Schedule

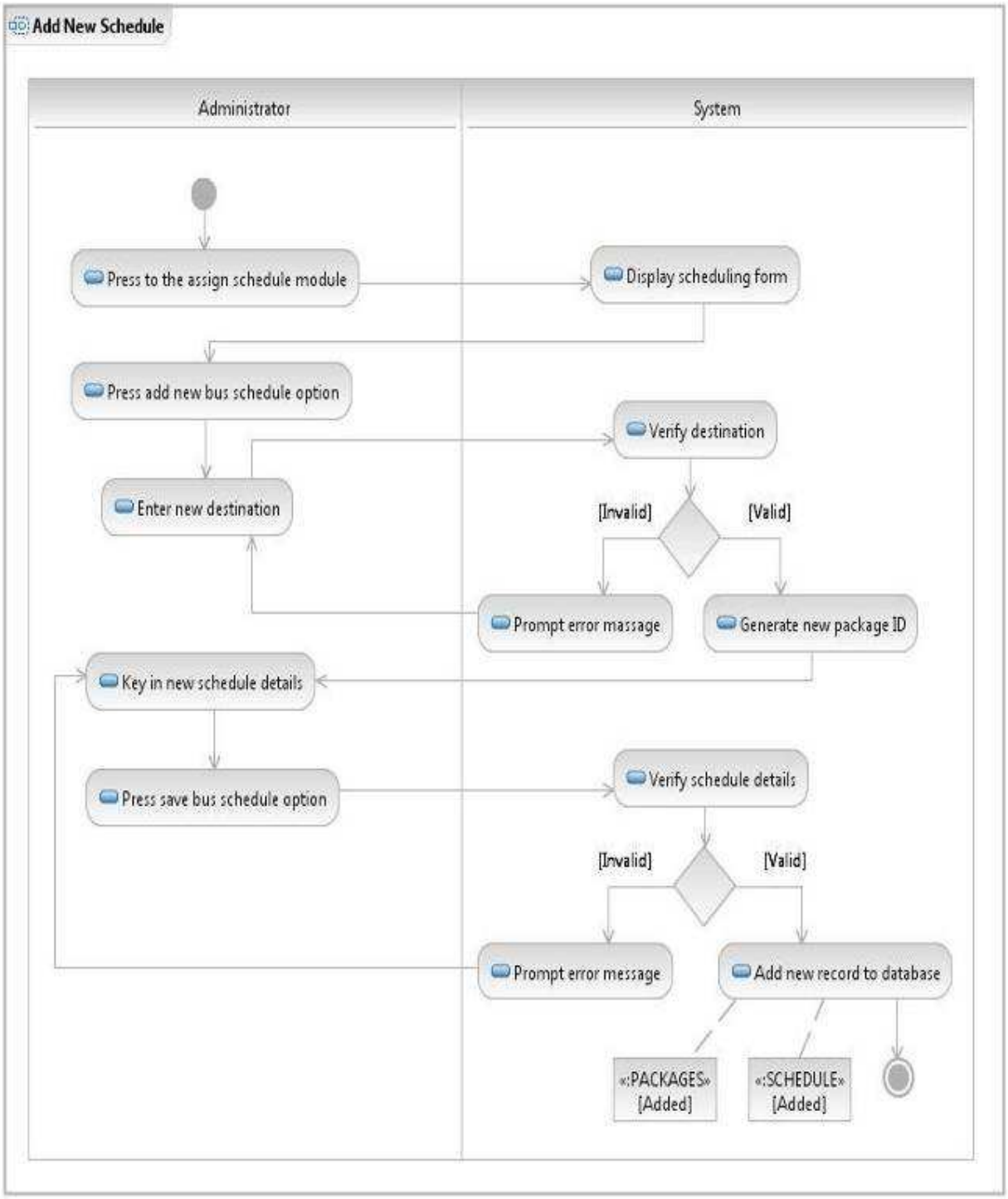


Figure 3.3.14 Activity Diagram for Add New Schedule

Edit Schedule Details

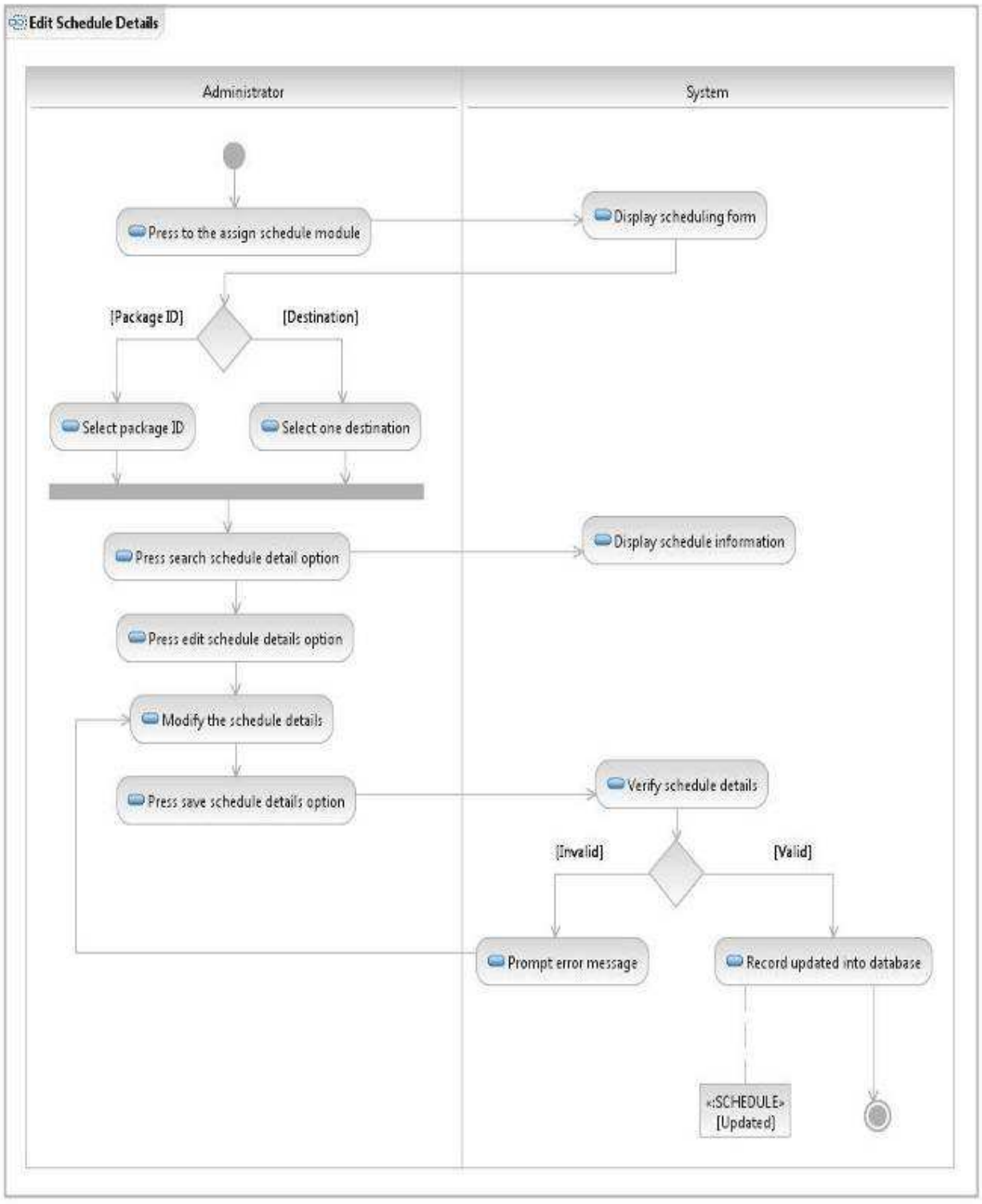


Figure 3.3.15 Activity Diagram for Edit Schedule Details

Delete Schedule

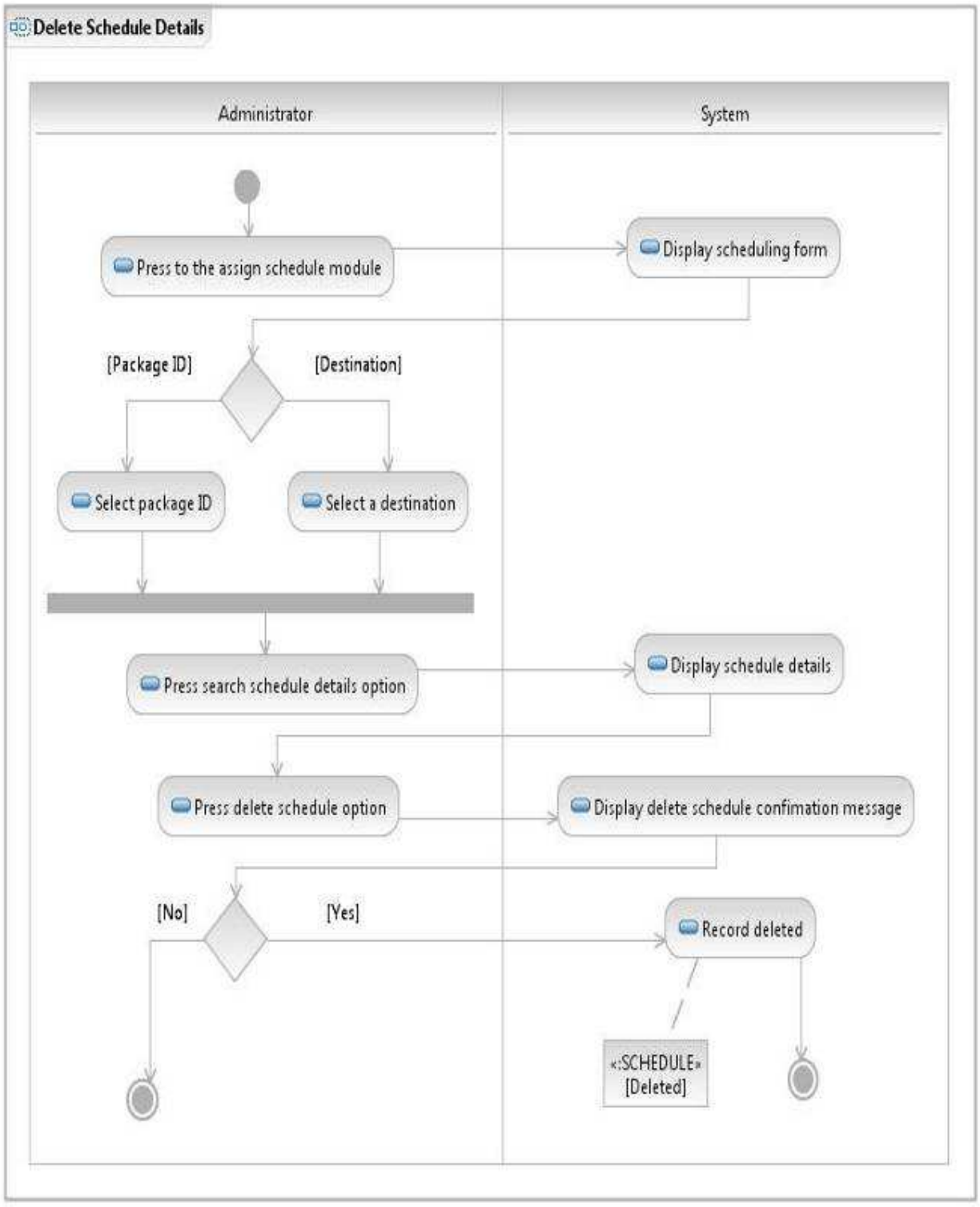


Figure 3.3.16 Activity Diagram for Delete Schedule

View Schedule Details

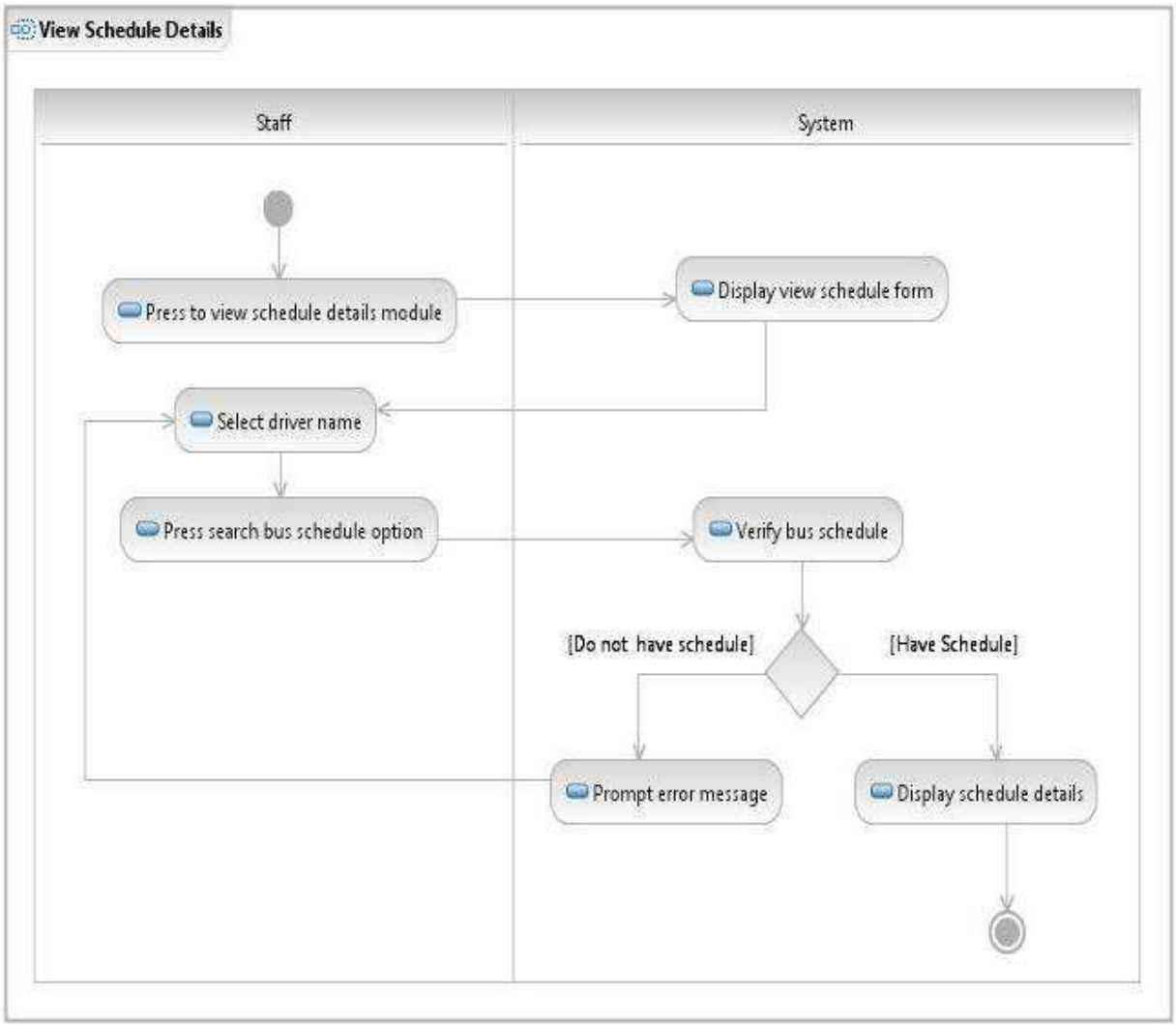


Figure 3.3.17 Activity Diagram for View Schedule Details

Activity Diagram for Repairing Module

Add New Service

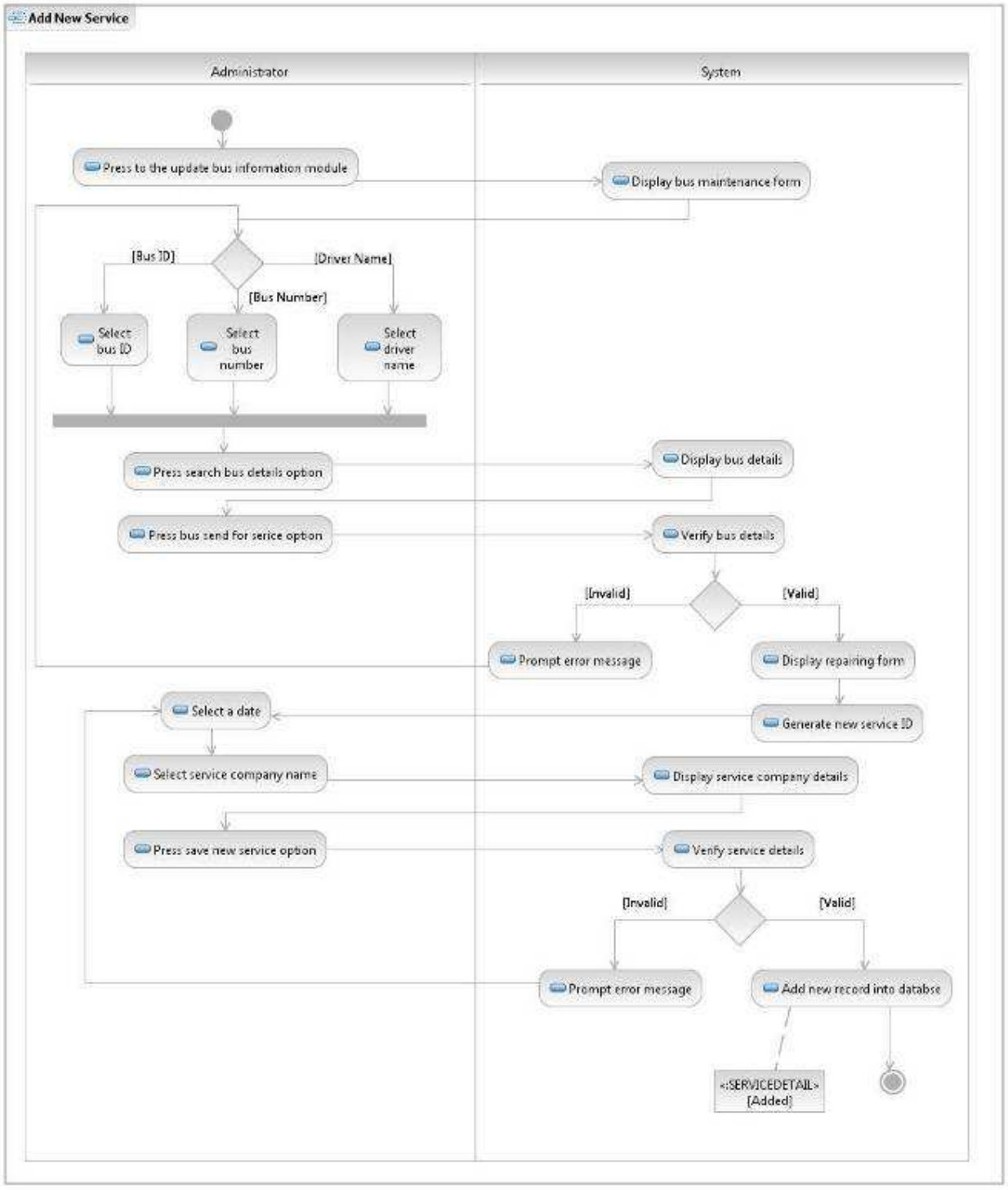


Figure 3.3.18 Activity Diagram for Add New Service

Edit Service Details

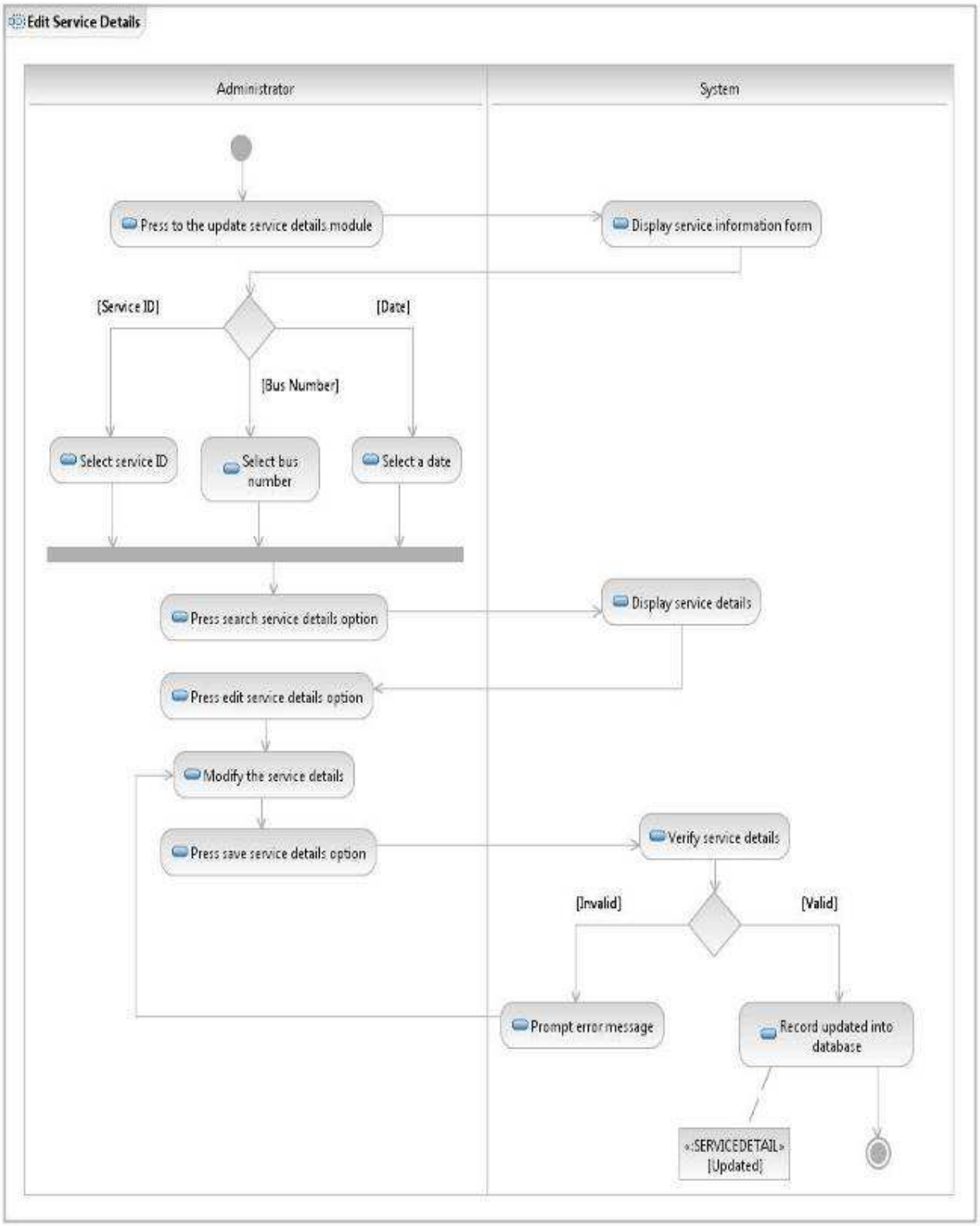


Figure 3.3.19 Activity Diagram for Edit Service Details

View Service Details

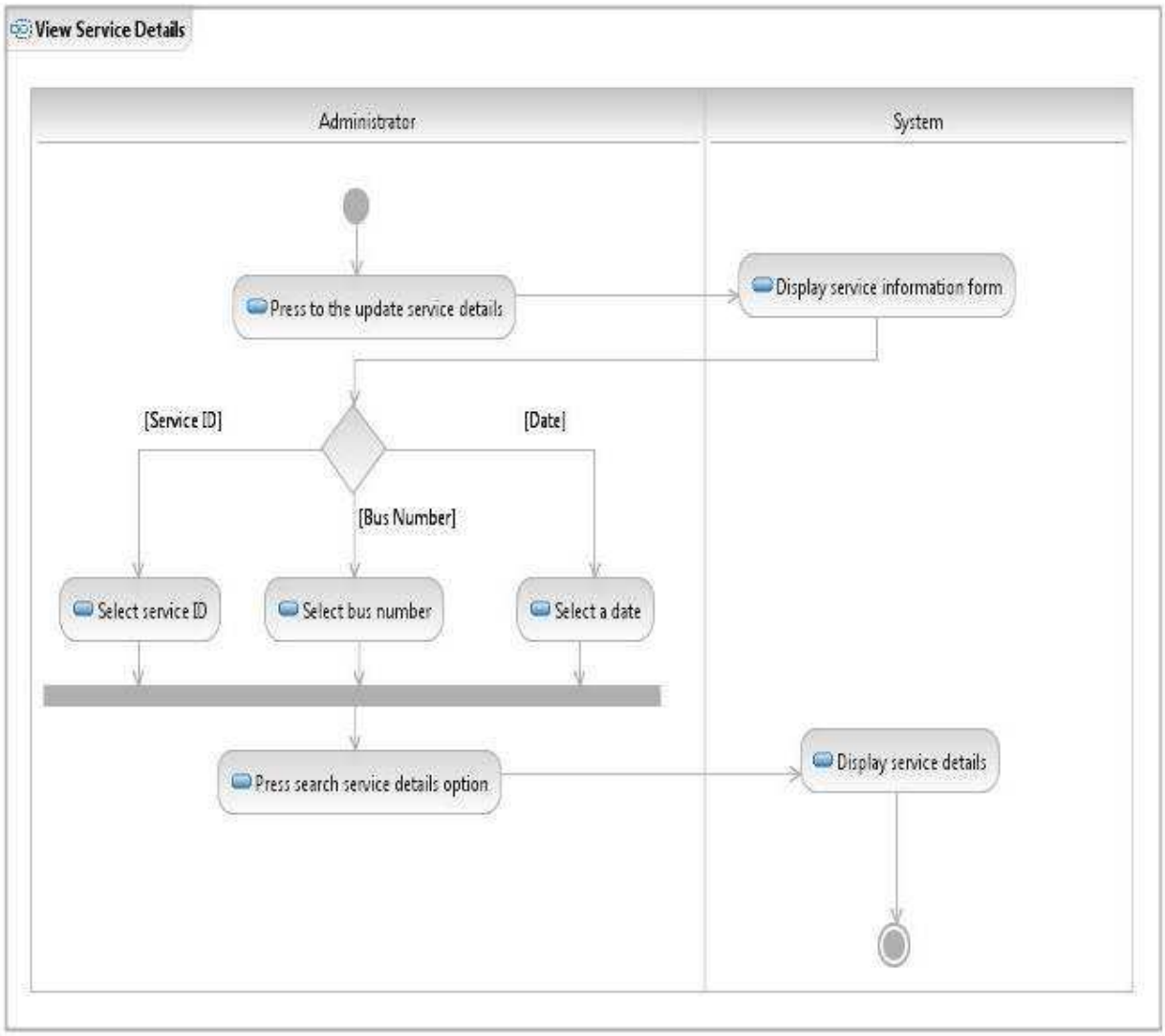


Figure 3.3.20 Activity Diagram for View Service Details

3.4 Sequence Diagram for Proposed Bus Ticketing System

Sequence Diagram for Maintenance Module

Staff Maintenance

Add New Staff

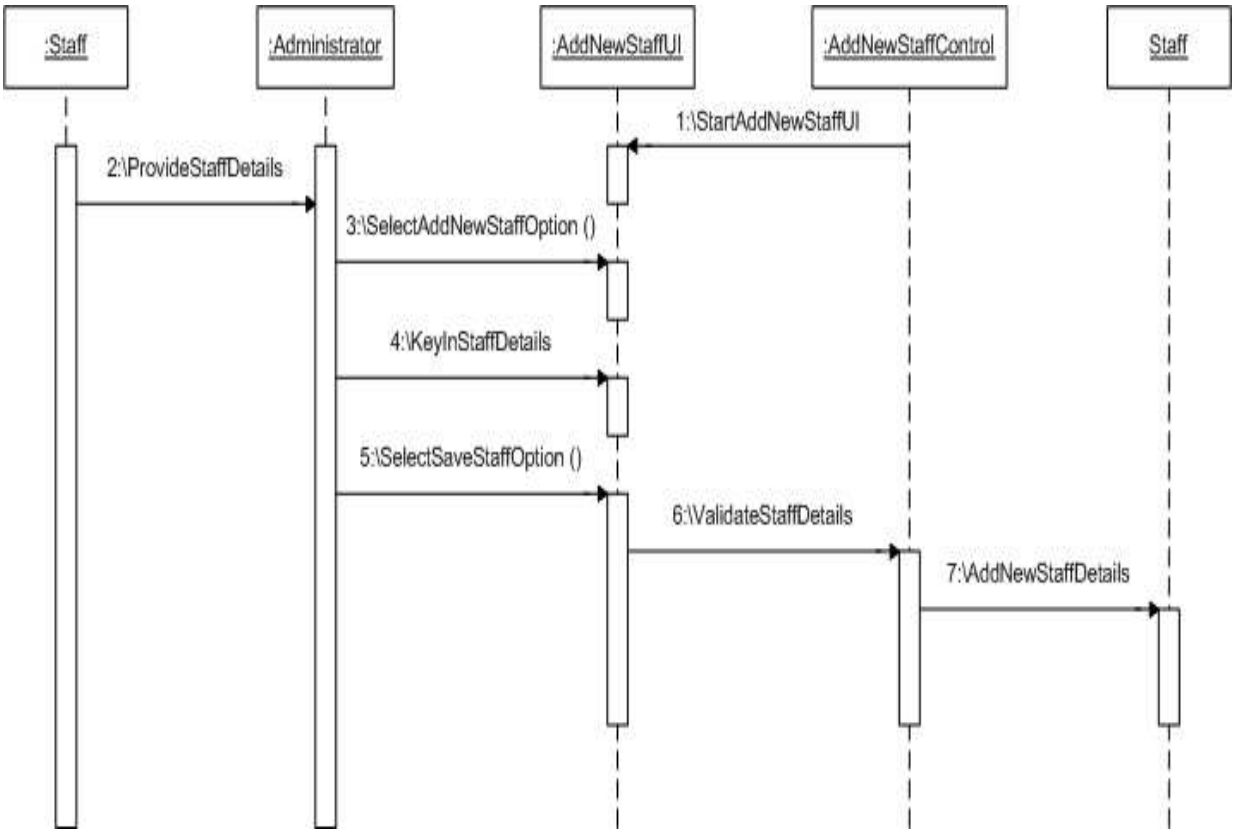


Figure 3.4.1 Sequence Diagram for Add New Staff

Edit Staff Details

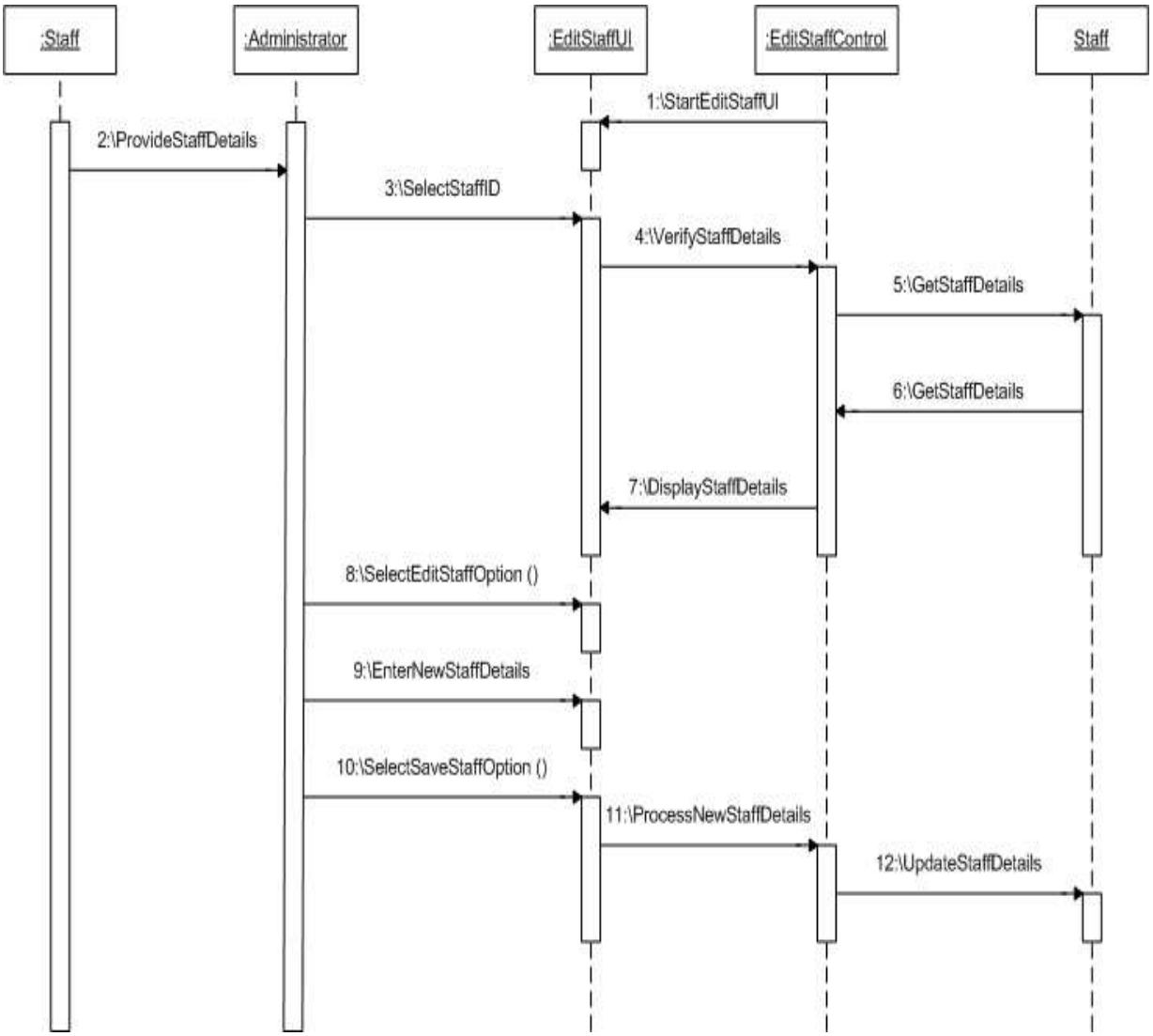


Figure 3.4.2 Sequence Diagram for Edit Staff Details

Delete Staff

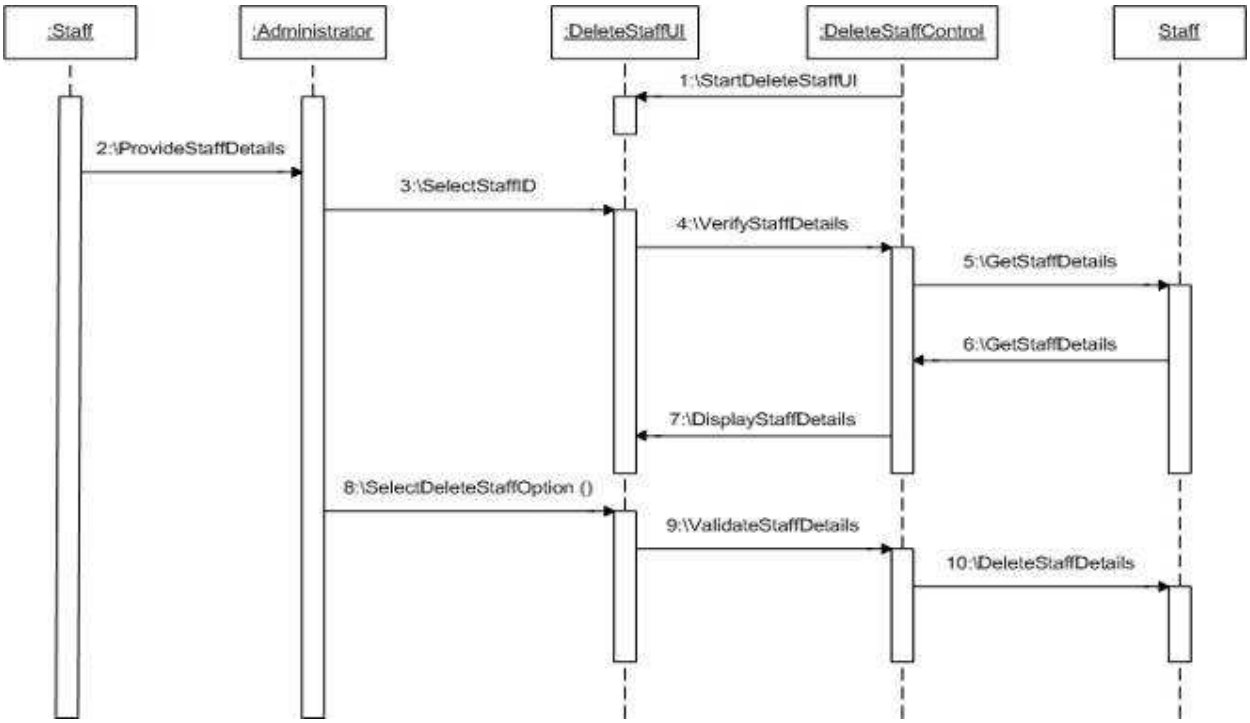


Figure 3.4.3 Sequence Diagram for Delete Staff

View Staff Details

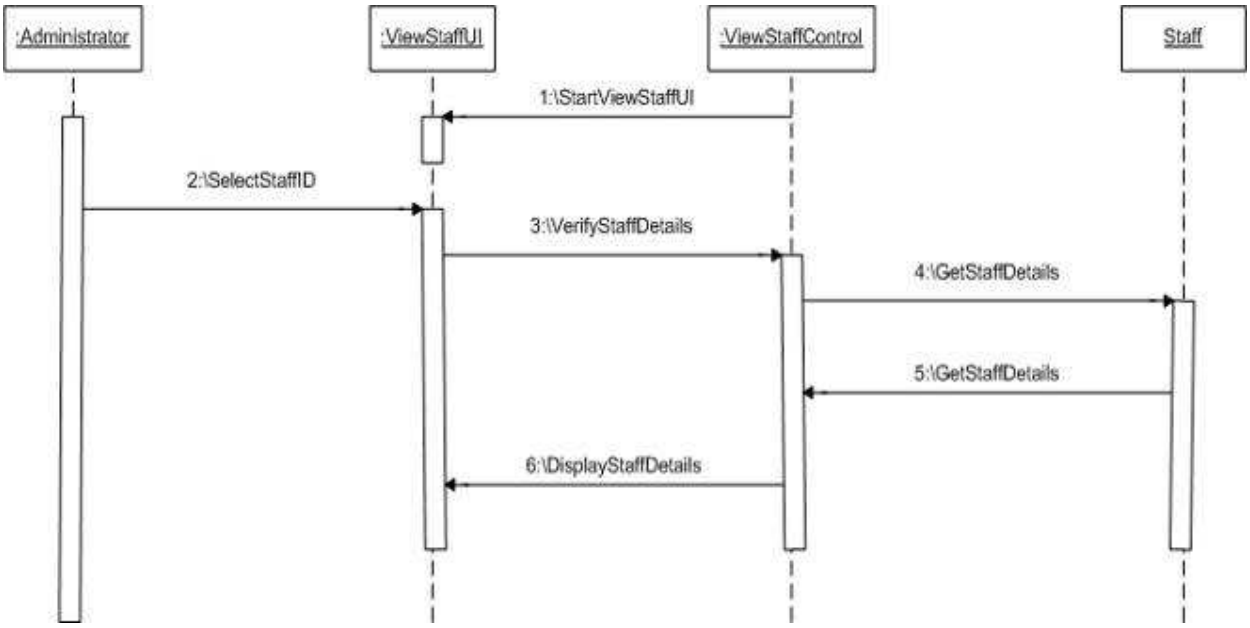


Figure 3.4.4 Sequence Diagram for View Staff Details

Member Maintenance

Add New Member

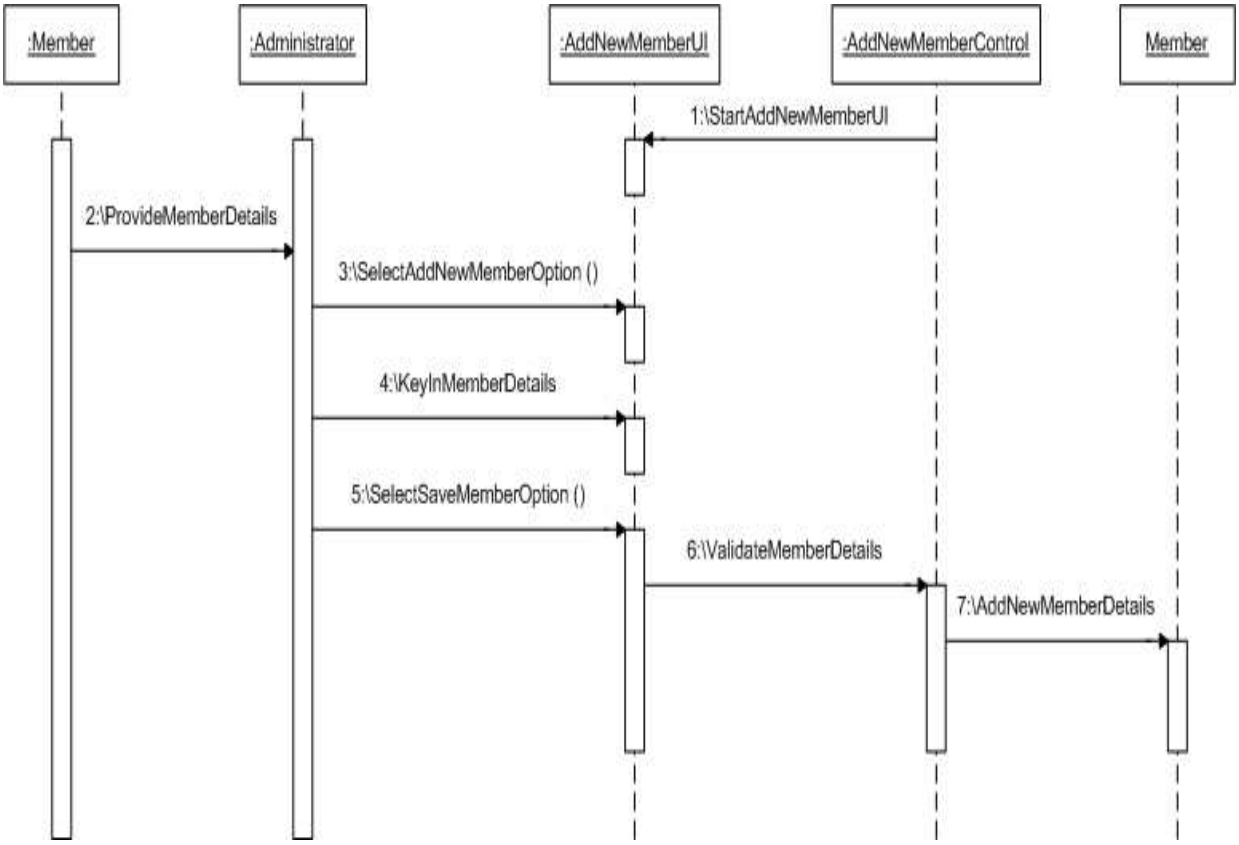


Figure 3.4.5 Sequence Diagram for Add New Member

Edit Member Details

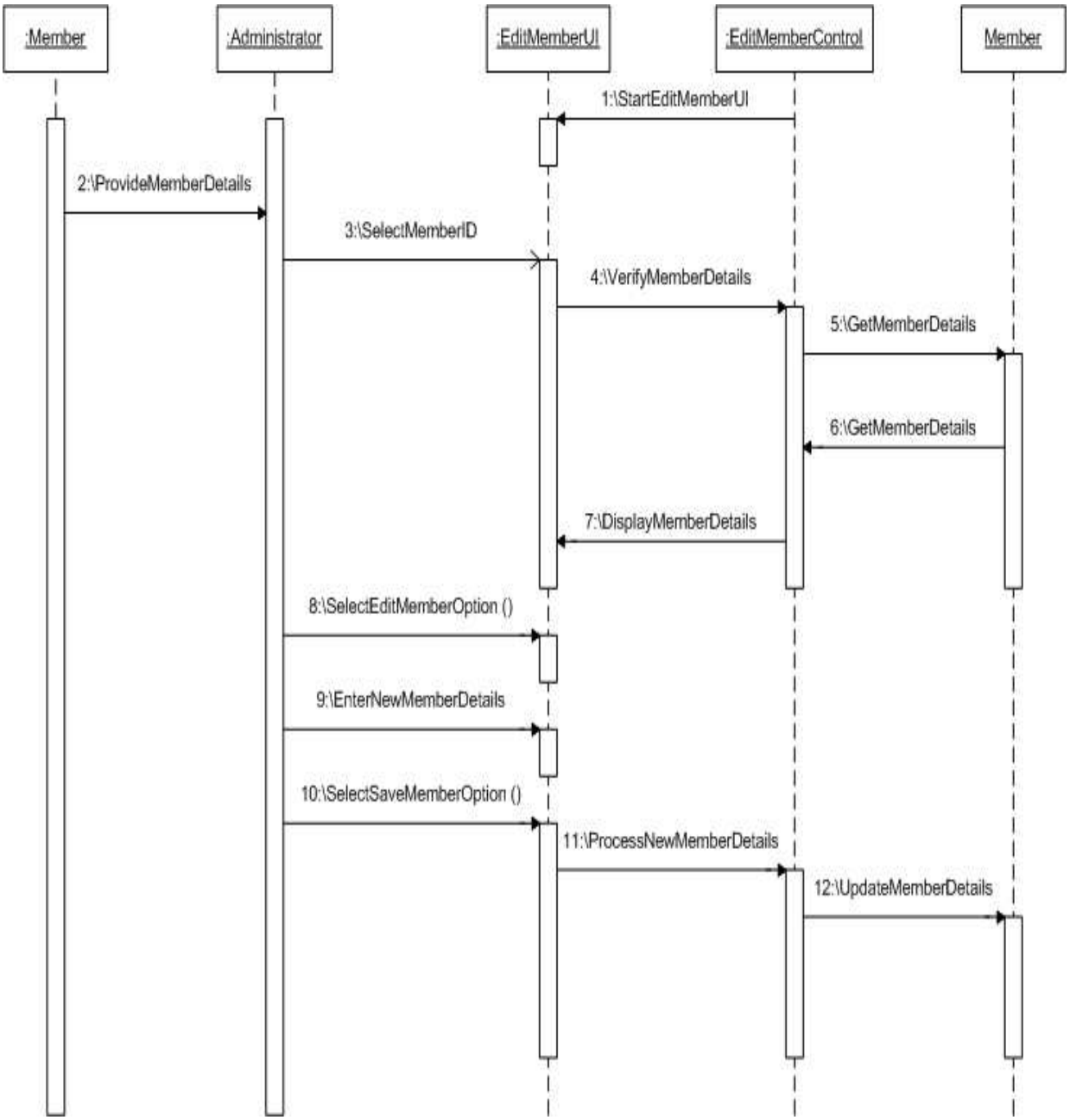


Figure 3.4.6 Sequence Diagram for Edit Member Details

Delete Member

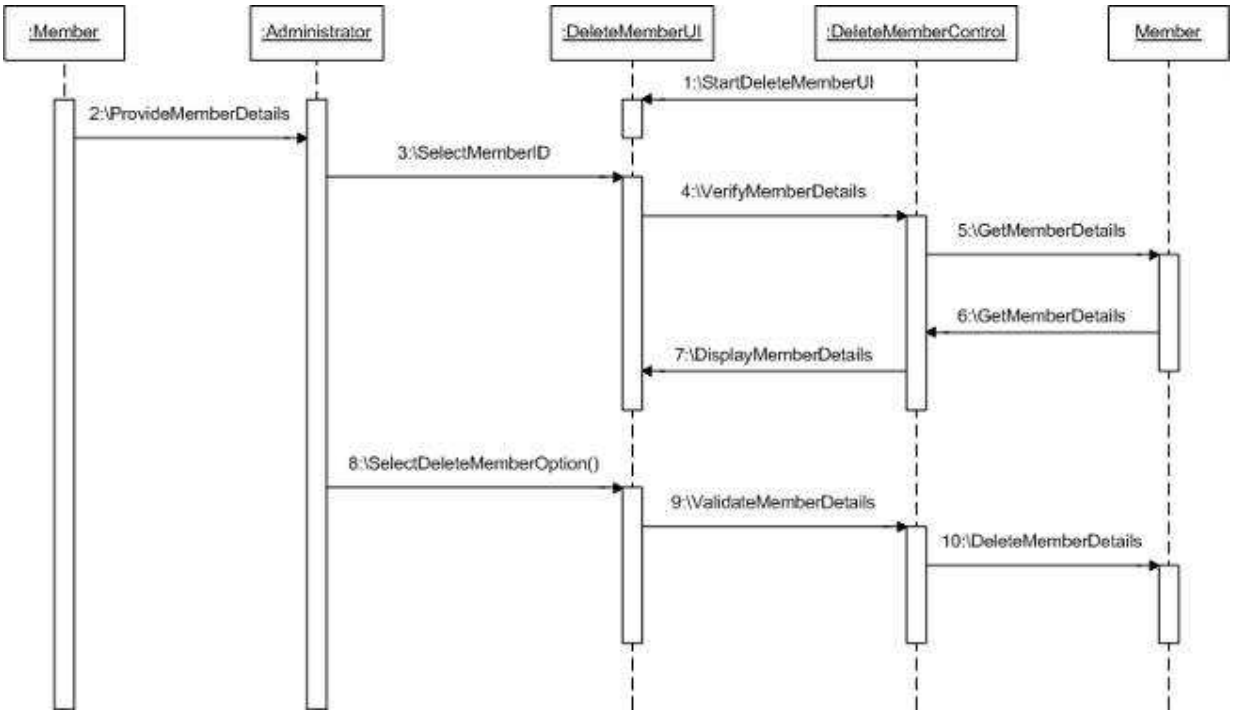


Figure 3.4.7 Sequence Diagram for Delete Member

View Member Details

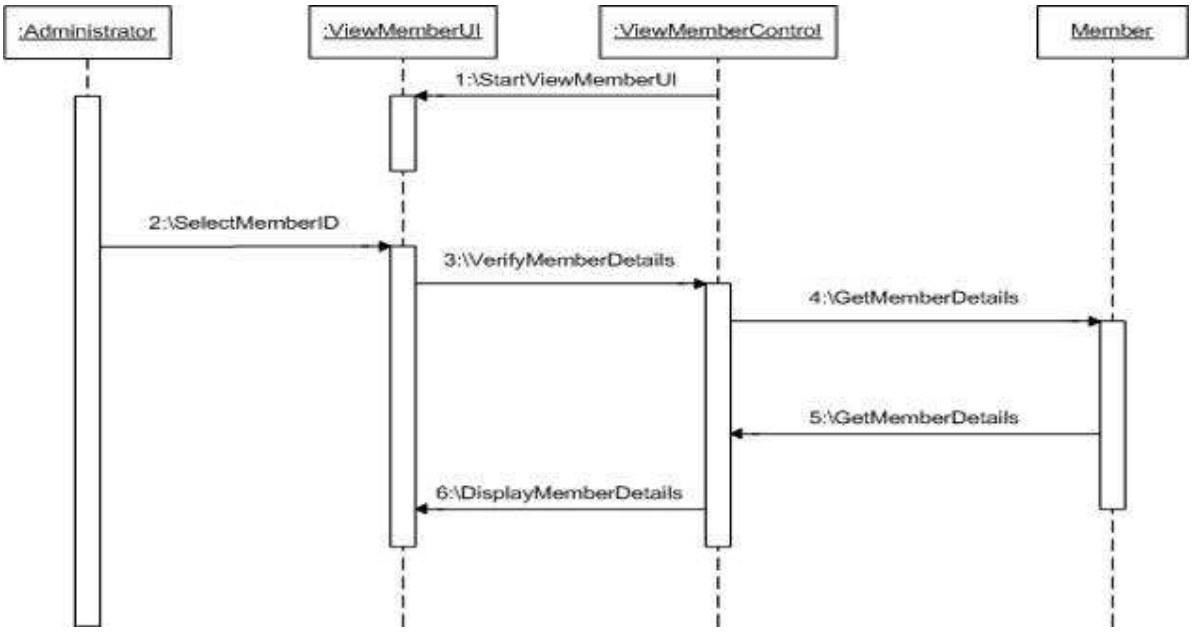


Figure 3.4.8 Sequence Diagram for View Member Details

Bus Maintenance

Add New Bus

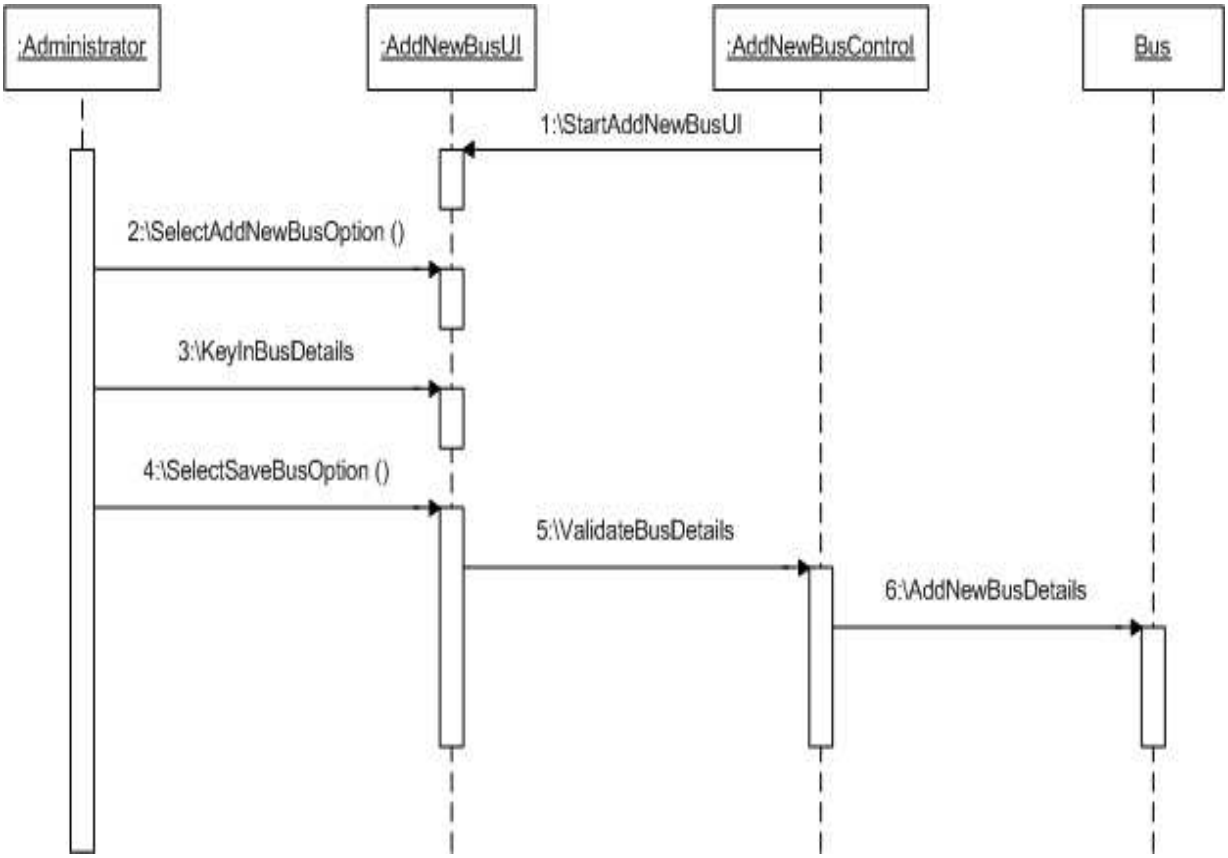


Figure 3.4.9 Sequence Diagram for Add New Bus

Edit Bus Details

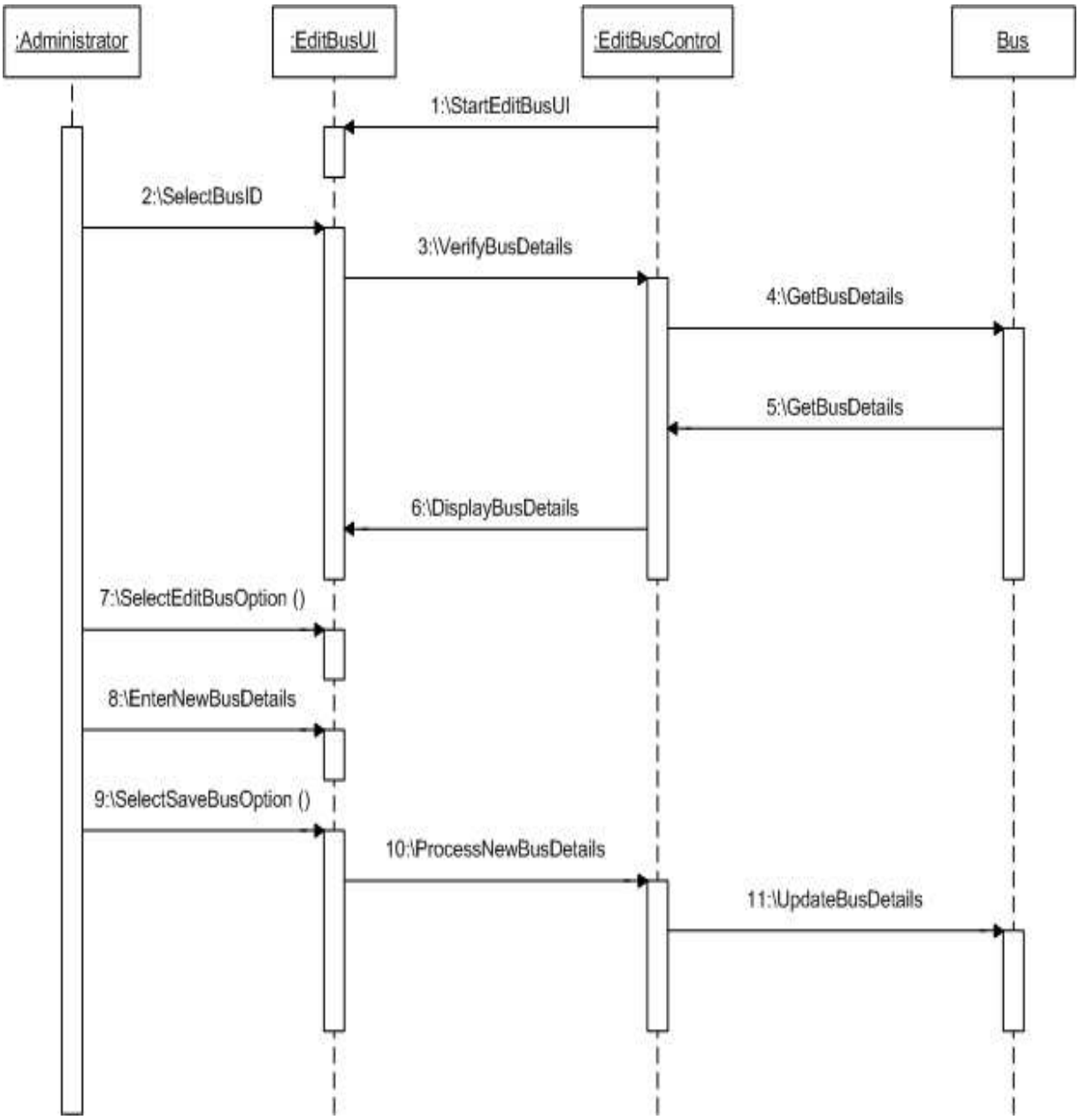


Figure 3.4.10 Sequence Diagram for Edit Bus Details

Delete Bus

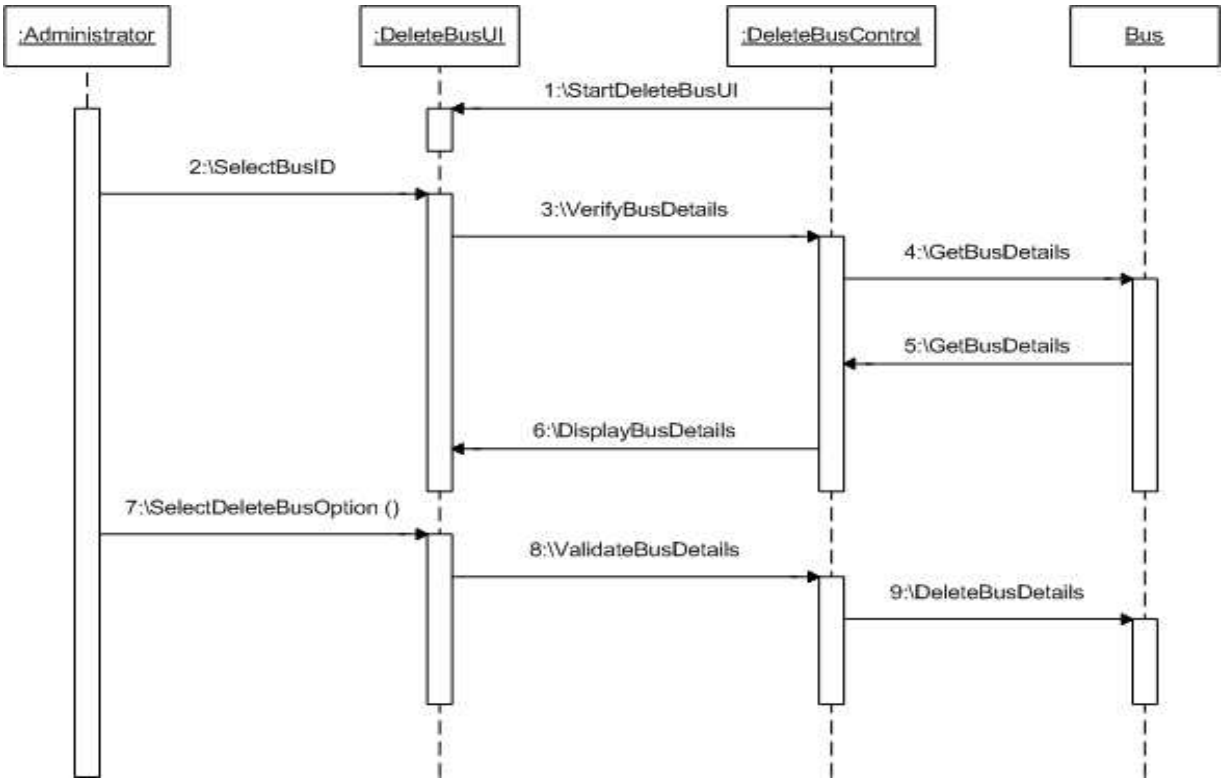


Figure 3.4.11 Sequence Diagram for Delete Bus

View Bus Details

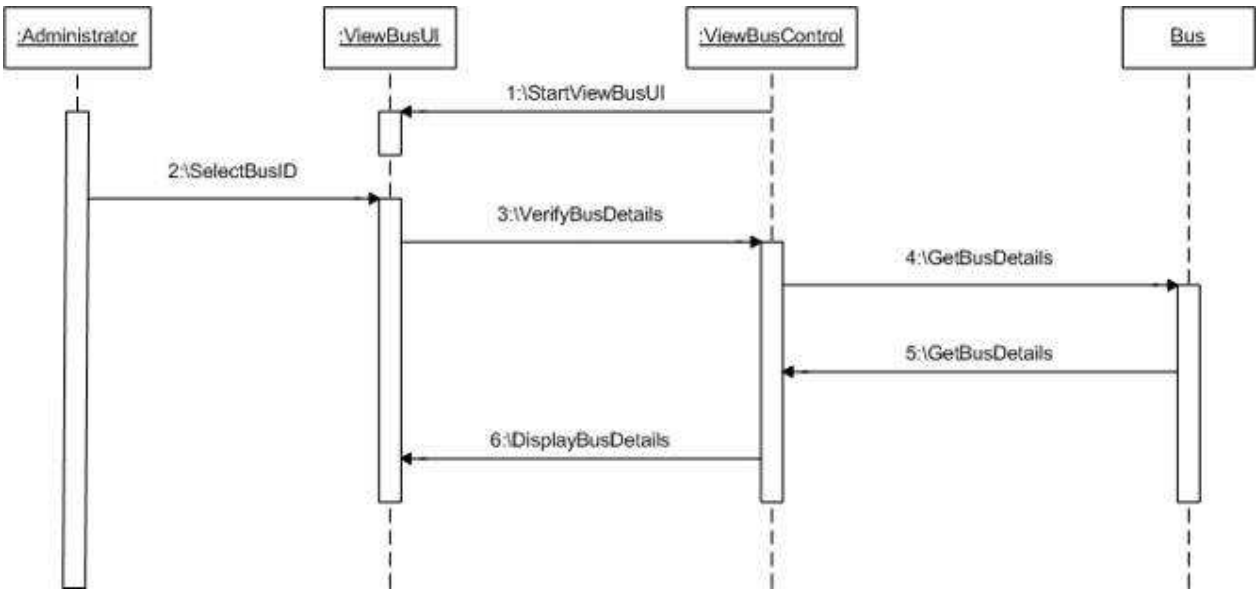


Figure 3.4.12 Sequence Diagram for View Bus Details

Sequence Diagram for Reservation Module

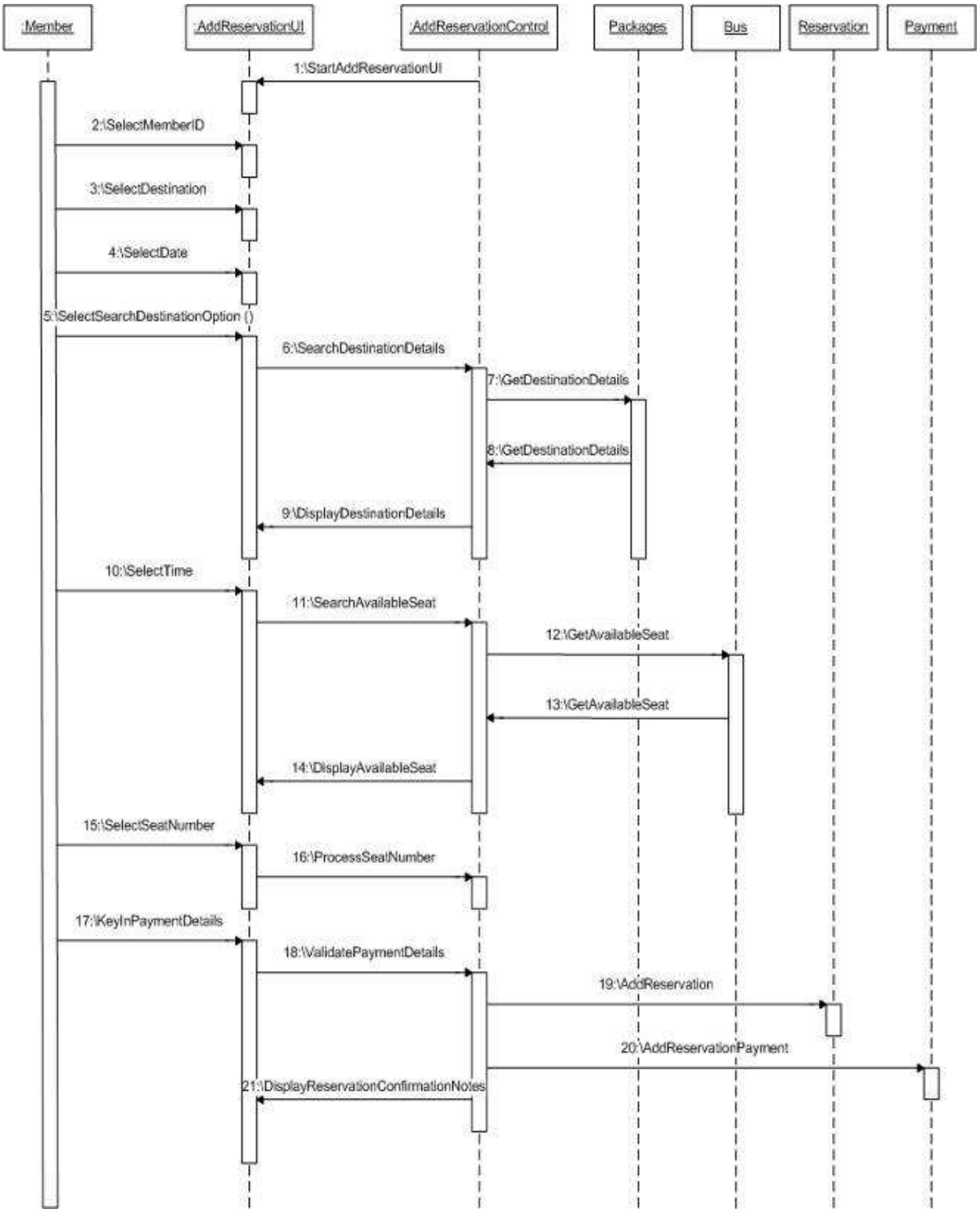


Figure 3.4.13 Sequence Diagram for Add Reservation and Add Reservation Payment

Sequence Diagram for Bus Scheduling Module

Add New Schedule

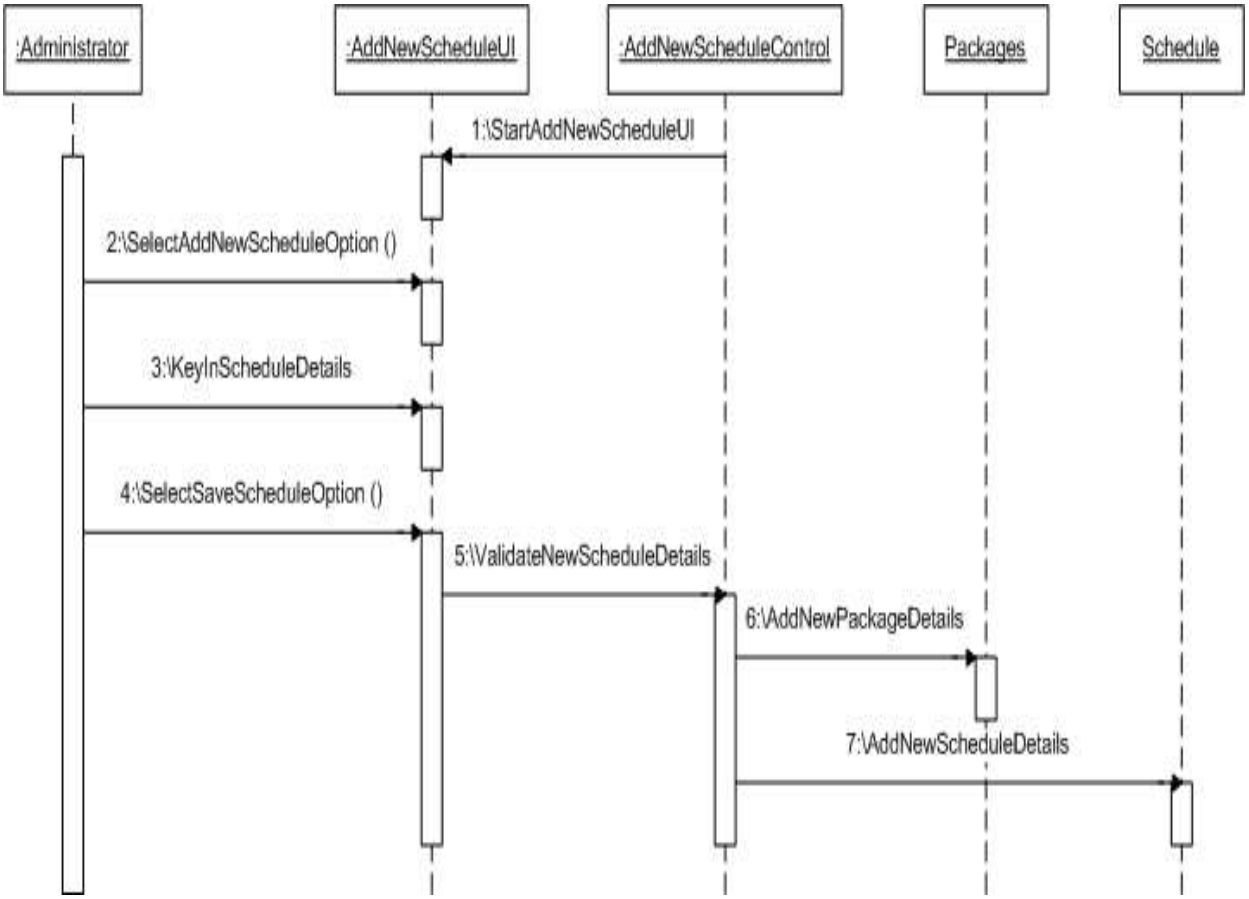


Figure 3.4.14 Sequence Diagram for Add New Schedule

Edit Schedule Details

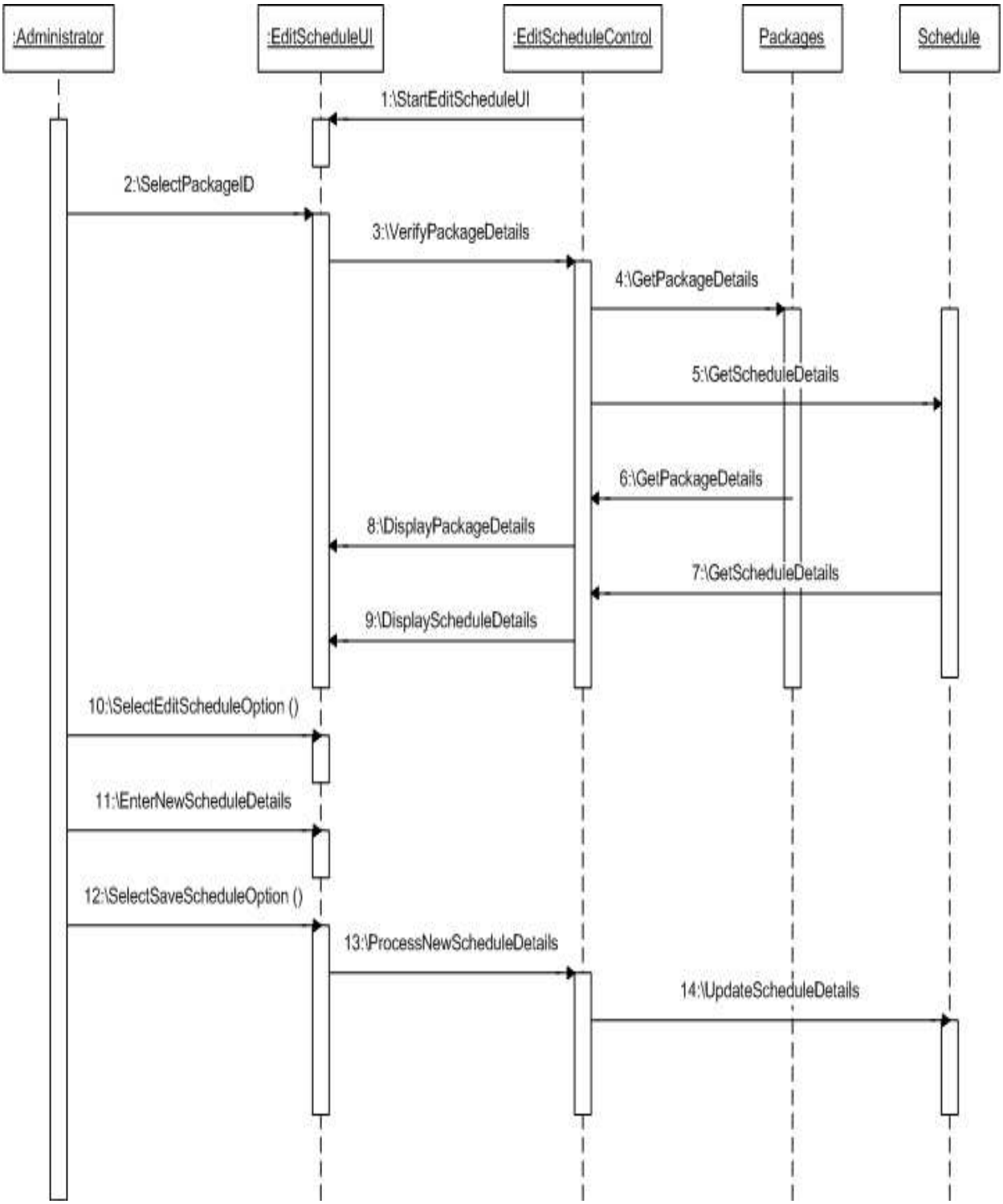


Figure 3.4.15 Sequence Diagram for Edit Schedule Details

Delete Schedule

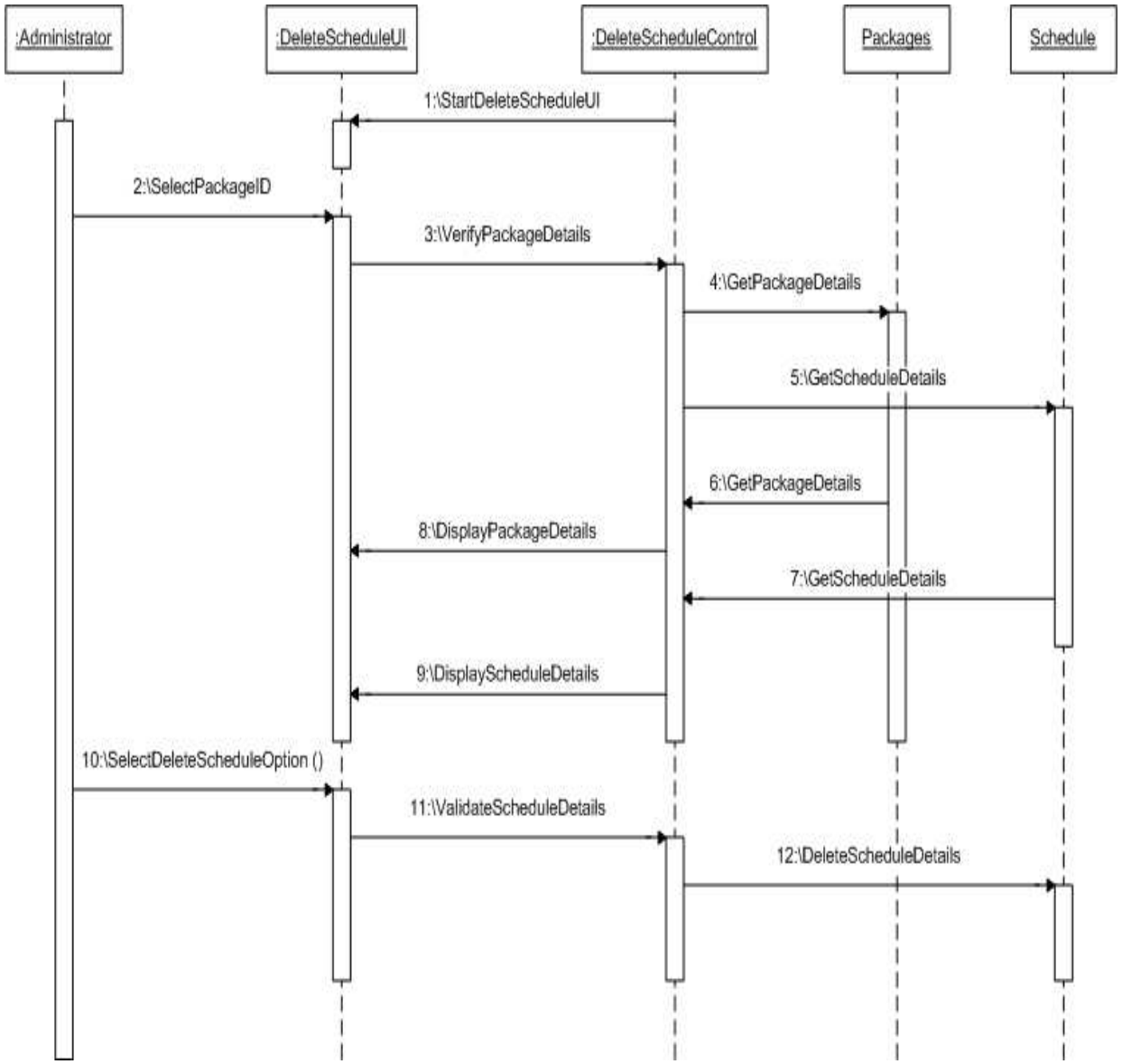


Figure 3.4.16 Sequence Diagram for Delete Schedule

View Schedule Details

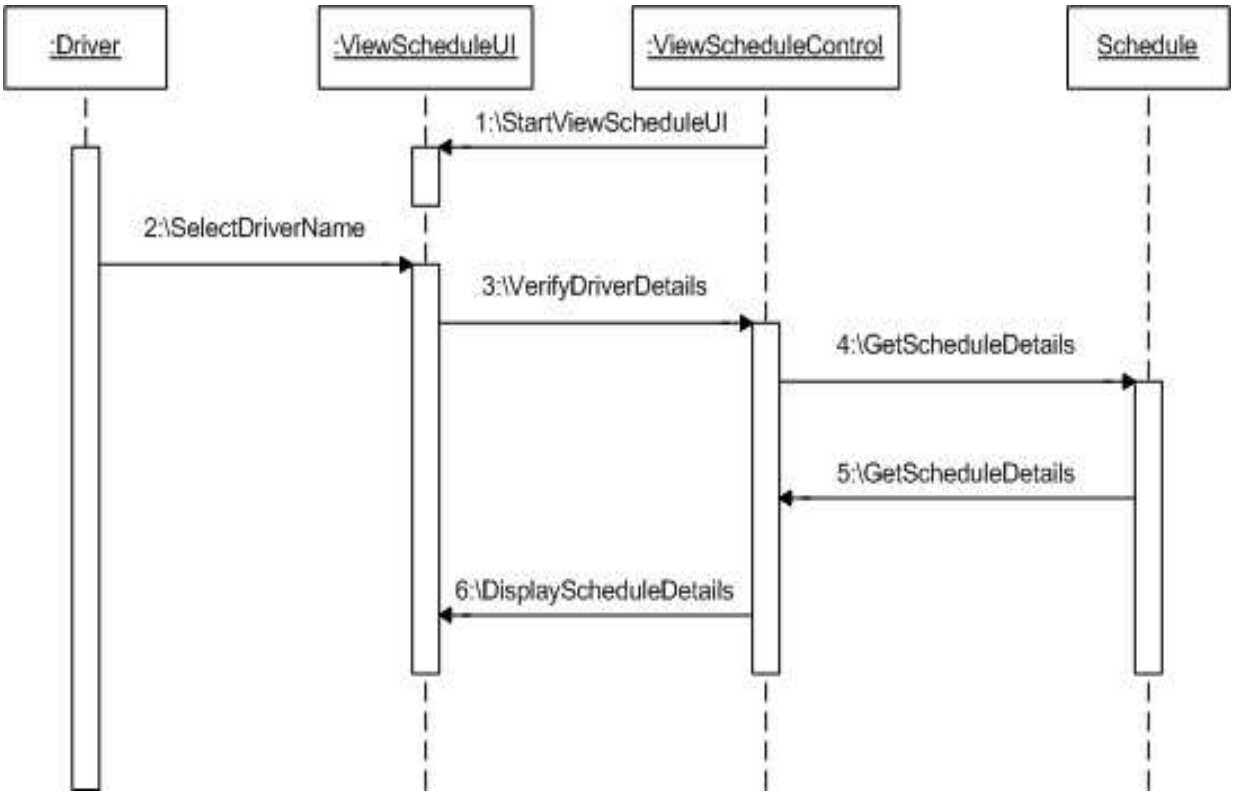


Figure 3.4.17 Sequence Diagram for View Schedule Details

Sequence Diagram for Repairing Module

Add New Service

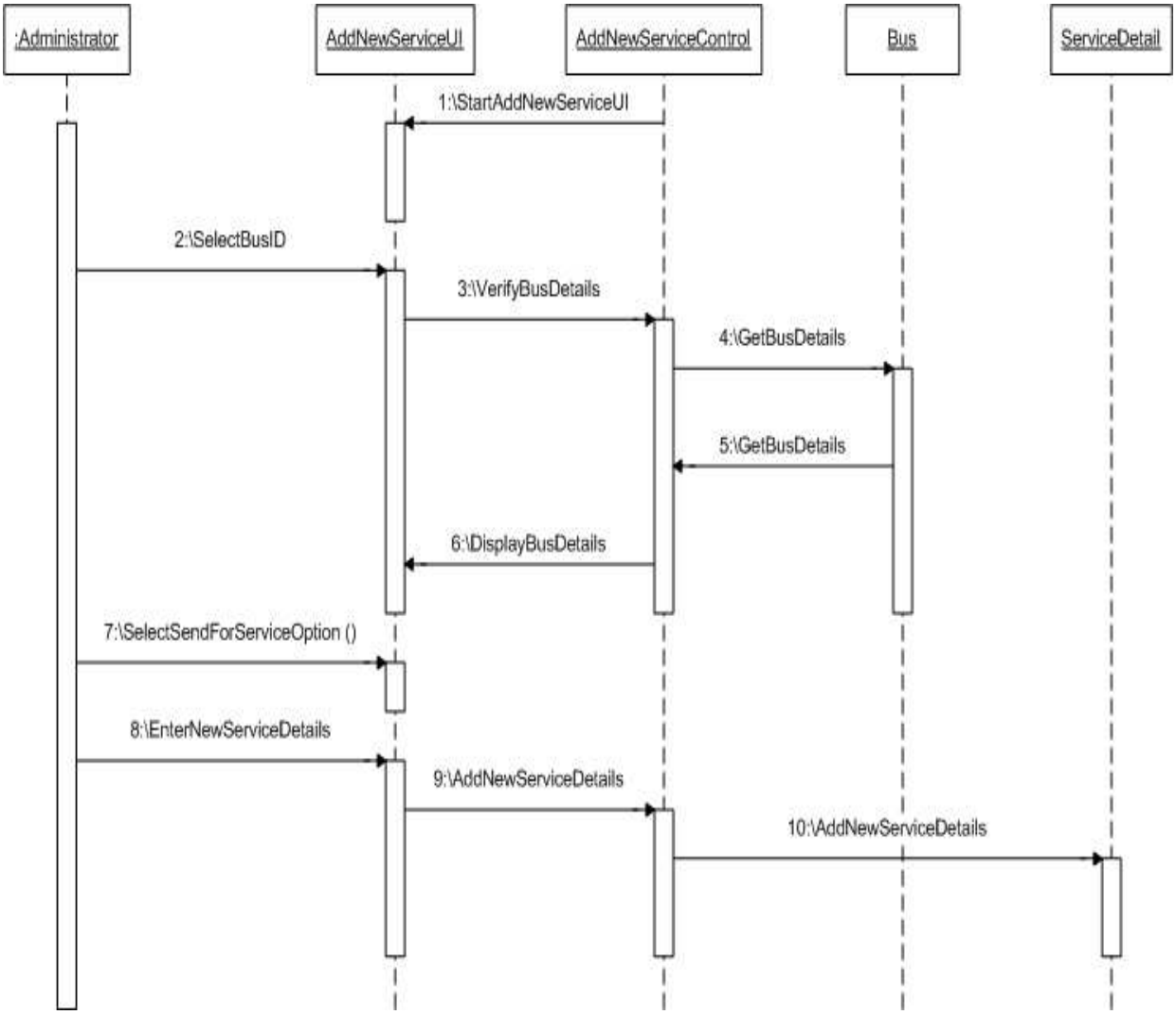


Figure 3.4.18 Sequence Diagram for Add New Service

Edit Service Details

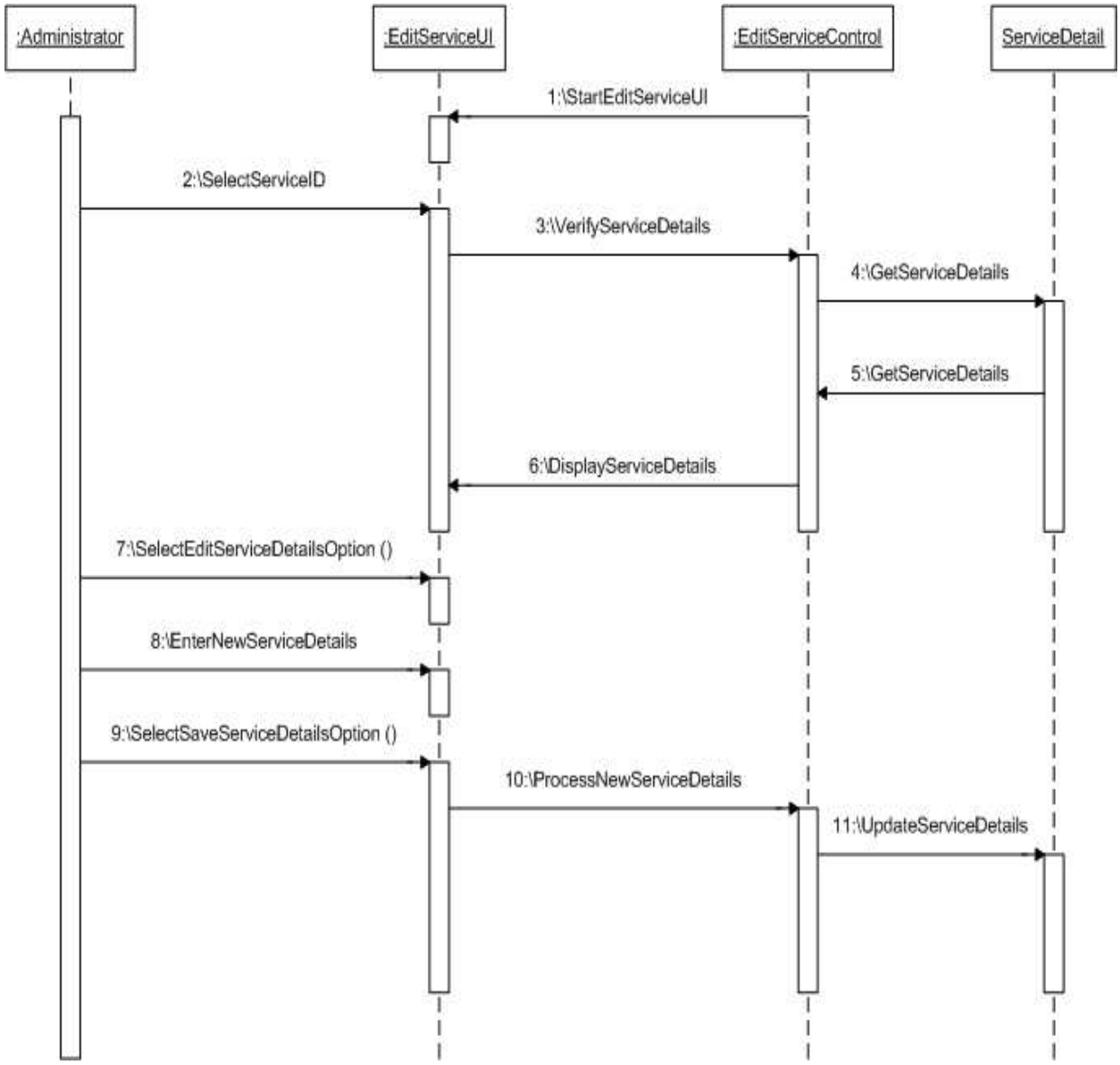


Figure 3.4.19 Sequence Diagram for Edit Service Details

View Service Details

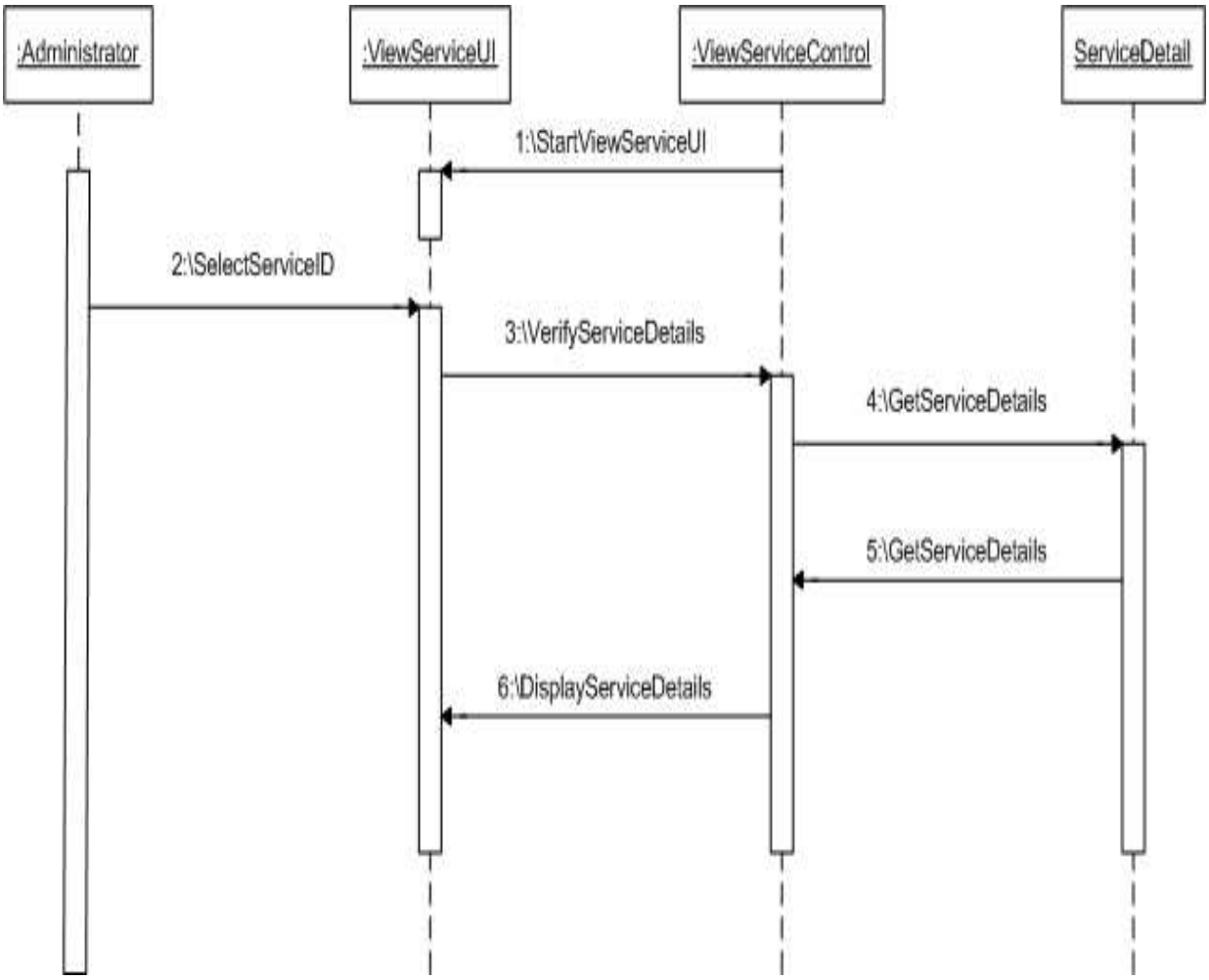


Figure 3.4.20 Sequence Diagram for View Service Details

3.5 Database Design

3.5.1 Data Dictionary

Table Name: BRANCH

Field Name	Data Type	Length	Key	Reference	Description
BRANCHID	nchar	5	Primary Key		A unique ID to identify each branch
COUNTER	nvarchar	30			A field that indicates branch's counter
ADDRESS	nvarchar	max			A field that indicates branch's address
POSTCODE	nchar	5			A field that indicates branch's postcode
STATE	nvarchar	20			A field that indicates branch's state
CONTACTNO	nvarchar	12			A field that indicates branch's contact number
EMAIL	nvarchar	50			A field that indicates branch's email address

Table Name: BUS

Field Name	Data Type	Length	Key	Reference	Description
BUSID	nchar	5	Primary Key		A unique ID to identify each bus
BUSNO	nchar	10			A field that indicates bus's number
SEAT	int	-			A field that indicates bus's seat
ENGINEMODEL	nvarchar	max			A field that indicates bus's engine model
CHASSISNO	nvarchar	30			A field that indicates bus's

					chassis number
MANUFACTURER	nvarchar	100			A field that indicates bus's manufacturer
MODEL	nvarchar	50			A field that indicates bus's model
COLOR	nvarchar	20			A field that indicates bus's colour
REGISTERNO	nvarchar	30			A field that indicates bus's register number
MADEIN	nvarchar	20			A field that indicates bus made in which year
MAKE	nvarchar	50			A field that indicates bus make by which company
PRICE	decimal	18, 0			A field that indicates bus's price
FUEL	nvarchar	15			A field that indicates bus's fuel
ENGINE CAPACITY	nvarchar	20			A field that indicates bus's engine capacity
BODYBUILDER	nvarchar	50			A field that indicates bus's body builder
PURCHASE YEAR	int	-			A field that indicates bus's purchase year
TRANSMISSION MODEL	nvarchar	50			A field that indicates bus's transmission model
AIRCONMODEL	nvarchar	50			A field that indicates bus's air-con model
STATUS	nchar	10			A field that indicates bus's status
STAFFID	nchar	10	Foreign Key	STAFF	A field that indicates bus's driver
PHOTO	image	-			A field that indicates bus's photo

Table Name: BUSTENANT

Field Name	Data Type	Length	Key	Reference	Description
BUSTENANTID	nchar	5	Primary Key		A unique ID to identify each bus tenant
NAME	nvarchar	30			A field that indicates bus tenant's name
COMPANY	nvarchar	50			A field that indicates bus tenant's company
IC	nvarchar	12			A field that indicates bus tenant's IC number
CONTACTNO	nchar	10			A field that indicates bus tenant's contact number
EMAIL	nvarchar	50			A field that indicates bus tenant's email address

Table Name: COMPANYPROFILE

Field Name	Data Type	Length	Key	Reference	Description
PROFILEID	nchar	5	Primary Key		A unique ID to identify the company profile
BACKGROUND	nvarchar	max			A field that indicates company's background
ABOUTUS	nvarchar	max			A field that indicates company's information
SAFETY	nvarchar	max			A field that indicates company's safety
SCHEDULE	nvarchar	max			A field that indicates company's schedule
CONTACTUS	nvarchar	50			A field that indicates company's contact number

Table Name: DAY

Field Name	Data Type	Length	Key	Reference	Description
DAYID	nchar	5	Primary Key		A unique ID to identify each day
DAY	nchar	10			A field that indicates days

Table Name: FEEDBACK

Field Name	Data Type	Length	Key	Reference	Description
FEEDBACKNO	nchar	5	Primary Key		A unique No to identify each feedback
FEEDBACK	nvarchar	max			A field that indicates feedback
DATE	datetime	-			A field that indicates feedback's date
TIME	datetime	-			A field that indicates feedback's time
MEMBERID	nchar	5	Foreign Key	MEMBER	A unique ID to identify each member

Table Name: MEMBER

Field Name	Data Type	Length	Key	Reference	Description
MEMBERID	nchar	5	Primary Key		A unique ID to identify each member
NAME	nvarchar	30			A field that indicates member's name
IC	nvarchar	12			A field that indicates member's IC number
GENDER	nchar	6			A field that indicates member's gender
ADDRESS	nvarchar	max			A field that indicates member's home address
CITY	nvarchar	20			A field that indicates

					member's city stay
STATE	nvarchar	20			A field that indicates member's state stay
POSTCODE	int	-			A field that indicates member's postcode stay
CONTACTNO	nchar	10			A field that indicates member's contact number
EMAIL	nvarchar	50			A field that indicates member's email address
USERNAME	nvarchar	15			A field that indicates member's username
PASSWORD	nchar	10			A field that indicates member's password
QUESTION	nvarchar	50			A field that indicates member's privacy question
ANSWER	nvarchar	50			A field that indicates member's privacy answer
PHOTO	image	-			A field that indicates member's photo

Table Name: ORDERDETAIL

Field Name	Data Type	Length	Key	Reference	Description
ORDERID	nchar	5	Primary Key, Foreign Key	ORDER TABLE	A unique ID to identify each order
PACKAGEID	nchar	5	Primary Key, Foreign Key	PACKAGES	A unique ID to identify each packages
DEPARTURE DATE	datetime	-			A field that indicates departure date in the order details

DEPARTURE TIME	datetime	-			A field that indicates departure time in the order details
STATUS	nvarchar	15			A field that indicates order detail's status
SUBTOTAL	decimal	18, 2			A field that indicates subtotal of the order details
SEATNO	nchar	2	Primary Key		A unique No to identify each seat number
TICKETNO	nchar	5			A field that indicates ticket number
RETURNID	nchar	5	Foreign Key	RETURN TICKET	A unique ID to identify each return ticket

Table Name: ORDERTABLE

Field Name	Data Type	Length	Key	Reference	Description
ORDERID	nchar	5	Primary Key		A unique ID to identify each order
ORDERDATE	datetime	-			A field that indicates order's date
ORDERTIME	datetime	-			A field that indicates order's time

Table Name: PACKAGES

Field Name	Data Type	Length	Key	Reference	Description
PACKAGEID	nchar	5	Primary Key		A unique ID to identify each packages
DEPARTURE PLACE	nvarchar	30			A field that indicates package's departure place
DESTINATION	nvarchar	30			A field that indicates package's arrival destination

PRICE	decimal	18, 2			A field that indicates package's price
DURATION	nvarchar	30			A field that indicates package's duration
DISTANCE	nchar	10			A field that indicates package's distance
DEPARTURE TERMINAL	nvarchar	30			A field that indicates package's departure terminal
ARRIVAL TERMINAL	nvarchar	30			A field that indicates package's arrival terminal

Table Name: PAYMENT

Field Name	Data Type	Length	Key	Reference	Description
PAYMENTID	nchar	5	Primary Key		A unique ID to identify each payment
PAYMENT METHOD	nvarchar	15			A field that indicates payment's method
PAYMENT STATUS	nchar	10			A field that indicates payment's status
CREDITCARD TYPE	nvarchar	20			A field that indicates credit card type for payment
CREDITCARDNO	nvarchar	20			A field that indicates credit card number for payment
EXPDATE	nchar	10			A field that indicates credit card expired date for payment
DISCOUNT	decimal	18, 2			A field that indicates discount of the payment
TOTALAMOUNT	decimal	18, 2			A field that indicates payment's total amount
PAYDATE	datetime	-			A field that indicates

					payment's date
PAYTIME	datetime	-			A field that indicates payment's time
STAFFID	nchar	5	Foreign Key	STAFF	A unique ID to identify each staff
MEMBERID	nchar	5	Foreign Key	MEMBER	A unique ID to identify each member
ORDERID	nchar	5	Foreign Key	ORDER TABLE	A unique ID to identify each order

Table Name: RENTBUS

Field Name	Data Type	Length	Key	Reference	Description
RENTORDERID	nchar	5	Primary Key		A unique ID to identify each rent order
DATE	date	-			A field that indicates rent order's date
TIME	datetime	-			A field that indicates rent order's time
QUANTITY	nchar	2			A field that indicates rent order's quantity
STATUS	nchar	10			A field that indicates rent order's status
BUSTENANTID	nchar	5	Foreign Key	BUS TENANT	A unique ID to identify each bus tenant

Table Name: RENTDETAIL

Field Name	Data Type	Length	Key	Reference	Description
RENTPACKAGEID	nchar	5	Primary Key, Foreign Key	RENT PACKAGES	A unique ID to identify each rent package
RENTORDERID	nchar	5	Primary	RENTBUS	A unique ID to identify each

			Key, Foreign Key		rent order
DATESTART	date	-			A field that indicates rent package's date start
DATERETURN	date	-			A field that indicates rent package's date return
PURPOSE	nvarchar	max			A field that indicates purpose of rent package
BUSID	nchar	5	Foreign Key	BUS	A unique ID to identify each bus

Table Name: RENTPACKAGES

Field Name	Data Type	Length	Key	Reference	Description
RENTPACKAGEID	nchar	5	Primary Key		A unique ID to identify each rent package
DAY	nchar	3			A field that indicates rent package's day
PRICE	decimal	18, 0			A field that indicates rent package's price

Table Name: RENTPAYMENT

Field Name	Data Type	Length	Key	Reference	Description
RENTPAYID	nchar	5	Primary Key		A unique ID to identify each rent payment
PAYMETHOD	nvarchar	15			A field that indicates rent payment's method
PAYSTATUS	nvarchar	20			A field that indicates rent payment's status
CREDITCARD	nvarchar	30			A field that indicates credit card number of the rent payment

EXPDATE	nchar	10			A field that indicates credit card expired date for rent payment
TOTALAMOUNT	decimal	18, 0			A field that indicates total amount of rent payment
PAYDATE	datetime	-			A field that indicates rent payment's date
PAYTIME	datetime	-			A field that indicates rent payment's time
STAFFID	nchar	5	Foreign Key	STAFF	A unique ID to identify each staff
RENTORDERID	nchar	5	Foreign Key	RENTBUS	A unique ID to identify each rent order
BUSTENANTID	nchar	5	Foreign Key	BUS TENANT	A unique ID to identify each bus tenant

Table Name: RESERVATION

Field Name	Data Type	Length	Key	Reference	Description
RESERVATIONNO	nchar	5	Primary Key		A unique ID to identify each reservation
SEATNO	int	-	Primary Key		A unique ID to identify each seat number
RESERVATION DATE	datetime	-			A field that indicates reservation's date
RESERVATION TIME	datetime	-			A field that indicates reservation's time
MEMBERID	nchar	5	Foreign Key	MEMBER	A unique ID to identify each member
PACKAGEID	nchar	5	Foreign Key	PACKAGES	A unique ID to identify each package
PAYMENTID	nchar	5	Foreign Key	PAYMENT	A unique ID to identify each payment

STATUS	nvarchar	15			A field that indicates reservation's status

Table Name: RETURNTICKET

Field Name	Data Type	Length	Key	Reference	Description
RETURNID	nchar	5	Primary Key		A unique ID to identify each return ticket
RETURNDATE	datetime	-			A field that indicates ticket return's date
RETURNTIME	datetime	-			A field that indicates ticket return's time
AMOUNTRETURN	decimal	18, 2			A field that indicates amount return for the ticket return

Table Name: SCHEDULE

Field Name	Data Type	Length	Key	Reference	Description
PACKAGEID	nchar	5	Primary Key, Foreign Key	PACKAGES	A unique ID to identify each package
TIMEID	nchar	5	Primary Key, Foreign Key	TIMES1	A unique ID to identify each time
DAYID	nchar	5	Primary Key, Foreign Key	DAY	A unique ID to identify each day
STAFFID	nchar	5	Foreign Key	STAFF	A unique ID to identify each staff
DEPARTURETIME	datetime	-			A field that indicates

					schedule's departure time
ARRIVALTIME	datetime	-			A field that indicates schedule's arrival time

Table Name: SERVICECOMPANYDETAIL

Field Name	Data Type	Length	Key	Reference	Description
COMPANYID	nchar	5	Primary Key		A unique ID to identify each service company
NAME	nvarchar	40			A field that indicates service company's name
ADDRESS	nvarchar	max			A field that indicates service company's address
CITY	nvarchar	20			A field that indicates city of service company
STATE	nvarchar	20			A field that indicates state of service company
POSTCODE	int	-			A field that indicates service company's postcode
CONTACTNO	nvarchar	15			A field that indicates service company's contact number
EMAIL	nvarchar	50			A field that indicates service company's email address

Table Name: SERVICEDETAIL

Field Name	Data Type	Length	Key	Reference	Description
SERVICEID	nchar	5	Primary Key		A unique ID to identify each service
SERVICEDATE	datetime	-			A field that indicates service's date
SERVICETIME	datetime	-			A field that indicates service's time
MAINTENANCE	decimal	18, 2			A field that indicates

FEES					maintenance fee of service
DURATION	nvarchar	20			A field that indicates duration of service
COMPANYID	nchar	5	Foreign Key	SERVICE COMPANY DETAIL	A unique ID to identify each service company
BUSID	nchar	5	Foreign Key	BUS	A unique ID to identify each bus

Table Name: STAFF

Field Name	Data Type	Length	Key	Reference	Description
STAFFID	nchar	5	Primary Key		A unique ID to identify each staff
NAME	nvarchar	30			A field that indicates staff's name
IC	nvarchar	12			A field that indicates staff's IC number
GENDER	nchar	6			A field that indicates staff's gender
ADDRESS	nvarchar	max			A field that indicates staff's home address
CITY	nvarchar	20			A field that indicates staff's city of stay
STATE	nvarchar	20			A field that indicates staff's state of stay
POSTCODE	int	-			A field that indicates staff's postcode of stay
CONTACTNO	nvarchar	10			A field that indicates staff's contact number
POSITION	nvarchar	20			A field that indicates staff's position
USERNAME	nvarchar	10			A field that indicates staff's

					username
PASSWORD	nchar	10			A field that indicates staff's password
QUESTION	nvarchar	50			A field that indicates staff's privacy question
ANSWER	nvarchar	50			A field that indicates staff's privacy answer
EMAIL	nvarchar	50			A field that indicates staff's email address
SALARY	int	-			A field that indicates staff's salary
EPFNO	nvarchar	50			A field that indicates staff's EPF number
SOCSONO	nvarchar	12			A field that indicates staff's SOCSO number
BANK	nvarchar	25			A field that indicates staff's bank name
BANKACCOUNTNO	nvarchar	30			A field that indicates staff's bank account number
PHOTO	image				A field that indicates staff's photo

Table Name: TIMES1

Field Name	Data Type	Length	Key	Reference	Description
TIMEID	nchar	5	Primary Key		A unique ID to identify each time
PERIOD	nvarchar	20			A field that indicates time's period
TIMES	datetime	-			A field that indicates times

Data Type Definition:

1. Nvarchar/Nchar – Character data types that are fixed-length, nchar, or variable-length, nvarchar.

2. Decimal – Data types that represent decimal floating-point numbers.
3. Integer – Exact-number data types that use integer data.
4. DateTime – Data types that represent date time format.
5. Date – Data types that represent date format.
6. Image – Data types that represent image format.

3.5.2 Normalization

3rd Normal Form

BRANCH (BRANCHID, COUNTER, ADDRESS, POSTCODE, STATE, CONTACTNO, EMAIL)

BUS (BUSID, BUSNO, SEAT, ENGINEMODEL, CHASSISNO, MANUFACTURER, MODEL, COLOR, REGISTERNO, MADEIN, MAKE, PRICE, FUEL, ENGINECAPACITY, BODYBUILDER, PURCHASEYEAR, TRANSMISSIONMODEL, AIRCONMODEL, STATUS, *STAFFID, PHOTO)

BUSTENANT (BUSTENANTID, NAME, COMPANY, IC, CONTACTNO, EMAIL)

COMPANYPROFILE (PROFILEID, BACKGROUND, ABOUTUS, SAFETY, SCHEDULE, CONTACTUS)

DAY (DAYID, DAY)

FEEDBACK (FEEDBACKNO, FEEDBACK, DATE, TIME, *MEMBERID)

MEMBER (MEMBERID, NAME, IC, GENDER, ADDRESS, CITY, STATE, POSTCODE, CONTACTNO, EMAIL, USERNAME, PASSWORD, QUESTION, ANSWER, PHOTO)

ORDERDETAIL (ORDERID, PACKAGEID, DEPARTUREDATE, DEPARTURETIME, STATUS, SUBTOTAL, SEATNO, TICKETNO, *RETURNID)

ORDERTABLE (ORDERID, ORDERDATE, ORDERTIME)

PACKAGES (PACKAGEID, DEPARTUREPLACE, DESTINATION, PRICE, DURATION, DISTANCE, DEPARTURETERMINAL, ARRIVALTERMINAL)

PAYMENT (PAYMENTID, PAYMENTMETHOD, PAYMENTSTATUS, CREDITCARDTYPE, CREDITCARDNO, EXPDATE, DISCOUNT, TOTALAMOUNT, PAYDATE, PAYTIME, *STAFFID, *MEMBERID, *ORDERID)

RENTBUS (RENTORDERID, DATE, TIME, QUANTITY, STATUS, *BUSTENANTID)

RENTDETAIL (RENTPACKAGEID, RENTORDERID, DATESTART, DATERETURN, PURPOSE, *BUSID)

RENTPACKAGES (RENTPACKAGEID, DAY, PRICE)

RENTPAYMENT (RENTPAYID, PAYMETHOD, PAYSTATUS, CREDITCARD, EXPDATE, TOTALAMOUNT, PAYDATE, PAYTIME, *STAFFID, *RENTORDERID, *BUSTENANTID)

RESERVATION (RESERVATIONNO, SEATNO, RESERVATIONDATE, RESERVATIONTIME, *MEMBERID, *PACKAGEID, *PAYMENTID, STATUS)

RETURNTICKET (RETURNID, RETURNDATE, RETURNTIME, AMOUNTRETURN)

SCHEDULE (PACKAGEID, TIMEID, DAYID, *STAFFID, DEPARTURETIME, ARRIVALTIME)

SERVICECOMPANYDETAIL (COMPANYID, NAME, ADDRESS, CITY, STATE, POSTCODE, CONTACTNO, EMAIL)

SERVICEDETAIL (SERVICEID, SERVICEDATE, SERVICETIME, MAINTENANCEFEES, DURATION, *COMPANYID, *BUSID)

STAFF (STAFFID, NAME, IC, GENDER, ADDRESS, CITY, STATE, POSTCODE, CONTACTNO, POSITION, USERNAME, PASSWORD, QUESTION, ANSWER, EMAIL, SALARY, EPFNO, SOCSONO, BANK, BANKACCOUNTNO, PHOTO)

TIMES1 (TIMEID, PERIOD, TIMES)

*Define as Foreign Key

3.6 Class Diagram

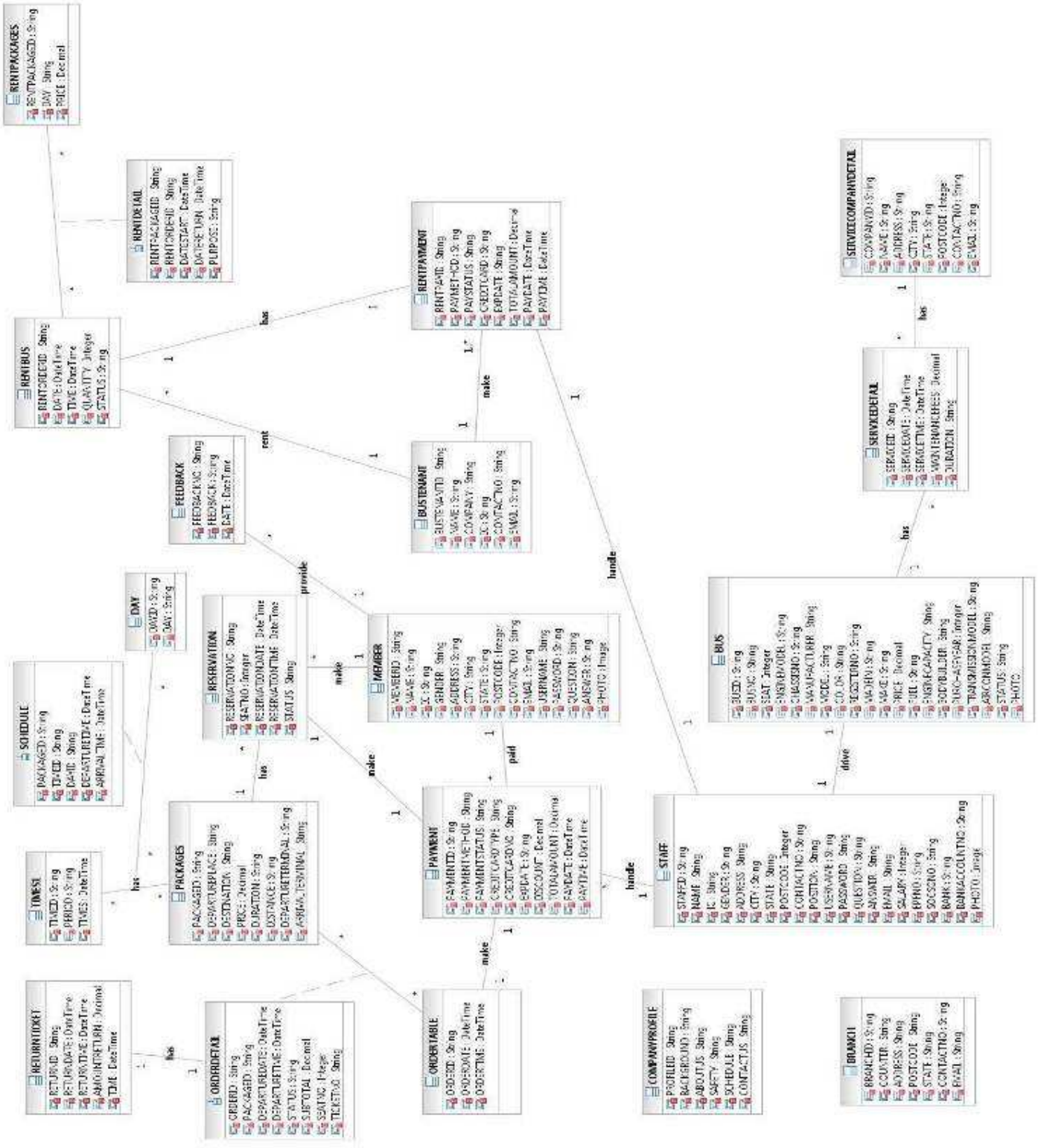


Figure 3.6.1 Class Diagram

3.7 Screen Design

Figure 3.7.1 Maintenance Menu

This maintenance menu is to let the administrator to choose which information that needs to be modifying by the administrator so they can choose either update staff information, update member information or update bus information. If they choose update staff information then the staff maintenance form will prompt out to the user. If they choose update member information then the member maintenance form will prompt out to the user. If they choose update bus information then the bus maintenance form will prompt out to the user.

Figure 3.7.2 Staff Maintenance Form

When a new employee hired by the company or quit from the company, administrator will get the new employee's information to record in the system or delete it from the system. Administrator allows reviewing and updating the information such as change address, change telephone number and so on only the appointed system administrators are allowed to maintain the staff information. If is new staff added so that the ID for the staff will automatically generated by the system so the administrator no need key in the ID by themselves and reduces the duplicate data. The staff also can search the staff detail using the Staff ID or Staff Name when the administrator clicks the search button.

Figure 3.7.3 Member Maintenance Form

This module provides a convenient way for the system administrator to maintain the registered customers. The maintenance activities include view members' information, add new members' information, edit members' information and delete members' information but only the appointed system administrator are allowed to maintain the member. Members can register or update their information through inform the system administrators, after that administrators will be process members' request. The member ID will be auto generate if the member is new register. Administrator can use search button to search member information by using their member ID or member Name.

Figure 3.7.4 Bus Maintenance Form

When a new bus bought by the company or cannot use anymore for the company, administrator will get the new bus's information to record in the system or delete it from the system. Administrator allows reviewing and updating the information such as change driver name, change colour and so on only the appointed system administrators are allowed to maintain the bus information. If is new bus added so that the ID for the bus will automatically generated by the system so the administrator no need key in the ID by themselves and reduces the duplicate data. Administrator also can search the bus detail using the Bus ID or Bus Number or Driver Name when the administrator clicks the search button.

Figure 3.7.5 Repairing Menu

This repairing menu is to let the administrator to choose update service details for the bus. When the administrator click the update service details then the service information form will prompt out for the user to edit the information.

Figure 3.7.6 Service Information Form

This module provides a convenient way for the system administrator to maintain the service information. The activities include view services' information, add new services' information, and edit services' information but only the appointed system administrator are allowed to add the new service. Administrator will update the service information based on the receipt that given by the service company, after that administrator will be processes the service information. The service ID will be auto generate if the bus is new service. Administrator can use search button to search service information by using their service ID or Bus Number or Date.

Figure 3.7.7 Scheduling Menu

This scheduling menu is to let the administrator to choose which schedule information that needs to be modifying so they can choose assign schedule. If they choose assign schedule then the scheduling form will prompt out to the user. This menu also let the driver to view the schedule information. So when the drivers choose view schedule details then the view schedule form will prompt out to the user.

Figure 3.7.8 View Schedule Form

This module is let the driver view the schedule for themselves. Driver can view the schedule information that assign by the administrator so they can follow the schedule to departure on time.

Figure 3.7.9 Scheduling Form

When a new destination added by the company or delete from the company, administrator will get the new destination's information to record in the system or delete it from the system. Administrator allows reviewing and updating the information such as change driver name, change time and so on only the appointed system administrators are allowed to assign the schedule information. If is new destination added so that the ID for the destination will automatically generated by the system so the administrator no need key in the ID by themselves and reduces the duplicate data. The administrator also can search the destination detail using the Package ID or Destination from where to where when the administrator clicks the search button.

Figure 3.7.10 Reservation Form

This module is let the member to add reservation and the reservation payment. Member can select the destination and the date then click the search button so the information will display. If is new reservation and reservation payment added so that the ID for the reservation and the reservation payment will automatically generated by the system so the administrator no need key in the ID by themselves and reduces the duplicate data.

3.8 Chapter Summary

This chapter has listed out the activity diagram, use case diagram, sequence diagram, design for user interface and the database. While creating the class diagram, Microsoft Visio 2007 is used to design the Entity Relationship Diagram. Unified Modeling Language (UML) diagrams are used to show the interaction between actor and the functions. This enables an easier way to design the system.

IBM Rational XDE is used to design the use case diagram. Rational XDE is useful and user-friendly software. Period of learning this software is short since it has been taught during Advance Diploma first year.

Sequence diagram is drawn so that user able to knows the process flow of the function. By using Rational XDE software, sequence diagram is able to be drawn successfully.

User interface is the interface that is going to be used by the general public. Font-size, color and the attractiveness of the user interface must be good in order to attract user and ease of usage.

Chapter 4

Programming

4 Programming

4.1 Introduction

This chapter will study about the system implementation, the system implementation phase will include programming use to develop the system, program code organized and validation and verification for the system.

4.2 Programming Language Used

Visual Basic .NET (VB.NET) is an object-oriented computer programming language that can be viewed as an evolution of the classic Visual Basic (VB) which is implemented on the .NET Framework. Microsoft currently supplies two major implementations of Visual Basic: Microsoft Visual Studio, which is commercial software and Microsoft Visual Studio Express, which is free of charge.

4.3 Coding

This part of coding is taking from the frmStaff.vb. It show how “Add New Staff” working in the system

```
Private Sub btnSave_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
btnSave.Click
        Dim response As DialogResult

            If txtSName.Text = "" Then
                MessageBox.Show("Please key in staff name!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtPhone.Text = "" Then
                MessageBox.Show("Please key in staff telephone number!", "ERROR",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            ElseIf txtEmail.Text = "" Then
                MessageBox.Show("Please key in staff email!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtSIC.Text = "" Then
                MessageBox.Show("Please key in staff ic number!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtSalary.Text = "" Then
                MessageBox.Show("Please key in staff salary!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtSocso.Text = "" Then
                MessageBox.Show("Please key in staff socso number!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtBankAcc.Text = "" Then
                MessageBox.Show("Please key in staff bank account number!", "ERROR",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            ElseIf txtBankAcc.Text = "" Then
```

```
                MessageBox.Show("Please key in bank name!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtEPF.Text = "" Then
                MessageBox.Show("Please key in staff EPF number!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf cboGender.Text = "" Then
                MessageBox.Show("Please select staff gender!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtAddress.Text = "" Then
                MessageBox.Show("Please key in staff address!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtSIC.Text <> txtSocso.Text Then
                MessageBox.Show("Your Socso No must same as your ic number!", "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            ElseIf txtCity.Text = "" Then
                MessageBox.Show("Please key in staff city!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtState.Text = "" Then
                MessageBox.Show("Please key in staff state!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtPostcode.Text = "" Then
                MessageBox.Show("Please key in staff postcode!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtPostcode.Text.Length <> 5 Then
                MessageBox.Show("Please key in correct staff postcode!", "ERROR",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            ElseIf PicStaff.Image Is Nothing Then
                MessageBox.Show("Please select a photo!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf cboPosition.Text = "" Then
                MessageBox.Show("Please select staff position!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf Not IsNumeric(txtPhone.Text) Then
                MsgBox("Please key in right format for the telephone number!")
            ElseIf Not IsNumeric(txtBankAcc.Text) Then
                MsgBox("Please key in right format for the account number!")
            ElseIf Not IsNumeric(txtPostcode.Text) Then
                MsgBox("Please key in right format for the postcode number!")
            ElseIf Not IsNumeric(txtSalary.Text) Then
                MsgBox("Please key in right format for the salary!")
            Else
                response = MessageBox.Show("Save record?", "CONFIRMATION", MessageBoxButtons.YesNo,
MessageBoxIcon.Question)

                If response = Windows.Forms.DialogResult.Yes Then
                    Try
                        Initialize()
                        Dim sqlInsert As String = " INSERT INTO STAFF VALUES( @staffID,
@name,@ic,@gender,@address,@city, @state,@postcode,@telno,
@position,@username,@password,@privacyquestion,@privacyanswer,  @email,@salary,@epf,@socso,@bank,
@bankacc, @photo)"

                        Dim cmd As SqlCommand = New SqlCommand(sqlInsert, conBus)

                        cmd.Parameters.AddWithValue("@staffID", lblStaffID.Text)
                        cmd.Parameters.AddWithValue("@name", txtSName.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@ic", txtSIC.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@gender", cboGender.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@address", txtAddress.Text.ToUpper)
```

```
                        cmd.Parameters.AddWithValue("@city", txtCity.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@state", txtState.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@postcode", Integer.Parse(txtPostcode.Text))
                        cmd.Parameters.AddWithValue("@telno", txtPhone.Text)
                        cmd.Parameters.AddWithValue("@position", cboPosition.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@username", txtUserName.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@password", PasswordTextBox.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@privacyquestion",
cboPrivacyQuestion.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@privacyanswer",
PrivacyAnswerTextBox.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@email", txtEmail.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@salary", Integer.Parse(txtSalary.Text))
                        cmd.Parameters.AddWithValue("@epf", txtEPF.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@socso", txtSocso.Text)
                        cmd.Parameters.AddWithValue("@bank", cboBank.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@bankacc", txtBankAcc.Text)

                        Dim ms As MemoryStream = New MemoryStream()

                        PicStaff.Image.Save(ms, PicStaff.Image.RawFormat)

                        Dim data As Byte() = ms.GetBuffer()

                        Dim p As New SqlParameter("@photo", SqlDbType.Image)

                        p.Value = data

                        cmd.Parameters.Add(p)

                        cmd.ExecuteNonQuery()

                        MessageBox.Show("This Particular Data Has Been Added!")

                        btnSave.Enabled = False
                        btnCancel.Enabled = False
                        btnAdd.Visible = True
                        btnAdd.Enabled = True
                        btnEdit.Enabled = True
                        btnEdit.Visible = True
                        btnDelete.Visible = True
                        grpSearchBy.Visible = True
                        btnPicture.Visible = False
                        grpCreditCheck.Visible = False
                        btnCompare.Enabled = False

                        txtSIC.ReadOnly = True
                        txtSName.ReadOnly = True
                        txtPhone.ReadOnly = True
                        txtAddress.ReadOnly = True
                        txtCity.ReadOnly = True
                        txtState.ReadOnly = True
                        txtPostcode.ReadOnly = True
                        txtBankAcc.ReadOnly = True
                        txtEmail.ReadOnly = True
                        txtEPF.ReadOnly = True
                        txtSalary.ReadOnly = True
                        txtSocso.ReadOnly = True
                        txtUserName.ReadOnly = True
```

```
                        PasswordTextBox.ReadOnly = True
                        RetypePasswordTextBox.ReadOnly = True
                        PrivacyAnswerTextBox.ReadOnly = True

                        cboGender.Enabled = False
                        cboBank.Enabled = False
                        cboPosition.Enabled = False
                        cboPrivacyQuestion.Enabled = False

                        cboStaffID.Text = ""
                        cboStaffID.Focus()

                    Catch ex As SqlClient.SqlException
                        MessageBox.Show("Please reenter the information!" & ControlChars.NewLine &
"System cannot accept some information!", "WARNING", MessageBoxButtons.OK, MessageBoxIcon.Warning)
                    Finally
                        Terminate()
                    End Try

                    FillByID()
                    FillByName()
                End If
            End If

    End Sub
```

This part of coding is taking from the frmMember.vb. It show how “Edit Member Details” working in the system

```
  Private Sub btnSave_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
btnSave.Click
        Dim response As DialogResult

            If txtName.Text = "" Then
                MessageBox.Show("Please key in staff name!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtTelNo.Text = "" Then
                MessageBox.Show("Please key in staff telephone number!", "ERROR",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            ElseIf txtEmail.Text = "" Then
                MessageBox.Show("Please key in staff email!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtIC.Text = "" Then
                MessageBox.Show("Please key in staff ic number!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf cboGender.Text = "" Then
                MessageBox.Show("Please select staff gender!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtAddress.Text = "" Then
                MessageBox.Show("Please key in staff address!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf PicMem.Image Is Nothing Then
                MessageBox.Show("Please select a photo!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtCity.Text = "" Then
                MessageBox.Show("Please key in staff city!", "ERROR", MessageBoxButtons.OK,
```

```
MessageBoxIcon.Error)
            ElseIf txtState.Text = "" Then
                MessageBox.Show("Please key in staff state!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtPostCode.Text = "" Then
                MessageBox.Show("Please key in staff postcode!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            ElseIf txtPostCode.Text.Length <> 5 Then
                MessageBox.Show("Please key in correct staff postcode!", "ERROR",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            ElseIf Not IsNumeric(txtTelNo.Text) Then
                MsgBox("Please key in right format for the telephone number!")
            ElseIf Not IsNumeric(txtPostCode.Text) Then
                MsgBox("Please key in right format for the postcode!")
            Else
                response = MessageBox.Show("Save record?", "CONFIRMATION", MessageBoxButtons.YesNo,
MessageBoxIcon.Question)

                If response = Windows.Forms.DialogResult.Yes Then
                    Try
                        Initialize()
                        Dim sqlUpdate As String
                        sqlUpdate = "Update MEMBER Set NAME=@name,  IC=@ic, GENDER=@gender,
ADDRESS=@address, CITY=@city, STATE=@state, POSTCODE=@postcode, CONTACTNO=@telno,
EMAIL=@email,USERNAME=@username, PASSWORD=@password, QUESTION=@privacyquestion,
ANSWER=@privacyanswer ,PHOTO=@photo WHERE MEMBERID = @memberID"

                        Dim cmd As SqlCommand = New SqlCommand(sqlUpdate, conBus)

                        cmd.Parameters.AddWithValue("@memberID", lblMemberID.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@name", txtName.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@ic", txtIC.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@gender", cboGender.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@address", txtAddress.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@city", txtCity.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@state", txtState.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@postcode", txtPostCode.Text)
                        cmd.Parameters.AddWithValue("@telno", txtTelNo.Text)
                        cmd.Parameters.AddWithValue("@email", txtEmail.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@username", txtUserName.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@password", PasswordTextBox.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@privacyquestion",
cboPrivacyQuestion.Text.ToUpper)
                        cmd.Parameters.AddWithValue("@privacyanswer",
PrivacyAnswerTextBox.Text.ToUpper)

                        Dim ms As MemoryStream = New MemoryStream()

                        PicMem.Image.Save(ms, PicMem.Image.RawFormat)

                        Dim data As Byte() = ms.GetBuffer()

                        Dim p As New SqlParameter("@photo", SqlDbType.Image)

                        p.Value = data

                        cmd.Parameters.Add(p)

                        cmd.ExecuteNonQuery()
```

```
                        MessageBox.Show("This Particular Data Has Been Updated!")

                        btnSave.Enabled = False
                        btnCancel.Enabled = False
                        btnAdd.Visible = True
                        btnAdd.Enabled = True
                        btnEdit.Enabled = True
                        btnEdit.Visible = True
                        btnDelete.Visible = True
                        grpSearchBy.Visible = True
                        btnPicture.Visible = False
                        grpCreditCheck.Visible = False
                        btnCompare.Enabled = False

                        txtIC.ReadOnly = True
                        txtName.ReadOnly = True
                        txtTelNo.ReadOnly = True
                        txtAddress.ReadOnly = True
                        txtCity.ReadOnly = True
                        txtState.ReadOnly = True
                        txtPostCode.ReadOnly = True
                        txtEmail.ReadOnly = True

                        cboGender.Enabled = False
                        cboPrivacyQuestion.Enabled = False
                        cboMemberID.Text = ""
                        cboMemberID.Focus()

                    Catch ex As Exception
                        MessageBox.Show("Please reenter the information!" & ControlChars.NewLine &
"System cannot accept some information!", "WARNING", MessageBoxButtons.OK, MessageBoxIcon.Warning)
                    End Try
                    Terminate()

                End If
            End If

    End Sub
```

This part of coding is taking from the frmBus.vb. It show how “Delete Bus” working in the system

```
Private Sub btnDelete_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
btnDelete.Click
        Dim response As DialogResult

            If cboBusID.Text = "" Then
                MessageBox.Show("Please select a bus record!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            Else
                response = MessageBox.Show("Confirm to delete this record?", "CONFIRMATION",
MessageBoxButtons.YesNo, MessageBoxIcon.Question)

                If response = Windows.Forms.DialogResult.Yes Then
```

```
                    Dim sqlDelete As String

                    sqlDelete = "DELETE FROM BUS WHERE BUSID='" & cboBusID.Text.ToUpper & "'"

                    Try
                        Initialize()

                        Dim adpBus As New SqlClient.SqlDataAdapter()

                        adpBus.DeleteCommand = New SqlClient.SqlCommand(sqlDelete)
                        adpBus.DeleteCommand.Connection = conBus
                        adpBus.DeleteCommand.ExecuteNonQuery()

                        MessageBox.Show("This Selected Data Has Been Deleted")

                        btnSave.Enabled = False
                        btnCancel.Enabled = False
                        btnAdd.Visible = True
                        btnAdd.Enabled = True
                        btnEdit.Enabled = True
                        btnEdit.Visible = True
                        btnDelete.Visible = True
                        grpSearchBy.Visible = True
                        btnPicture.Visible = False
                        btnSendForService.Enabled = False

                        lblBusID.Text = ""
                        txtBusNo.Text = ""
                        txtSeat.Text = ""
                        txtEngineModel.Text = ""
                        txtChassisNo.Text = ""
                        txtManufacturer.Text = ""
                        txtModel.Text = ""
                        txtColor.Text = ""
                        txtRegisterNo.Text = ""
                        txtMadeIn.Text = ""
                        txtPrice.Text = ""
                        txtFuel.Text = ""
                        txtEngineCapacity.Text = ""
                        txtBodyBuilder.Text = ""
                        txtTransmissionModel.Text = ""
                        txtMake.Text = ""
                        txtPurchaseYear.Text = ""
                        cboStatus.Text = ""
                        txtAirConModel.Text = ""
                        cboDriverName.Text = ""

                    Catch ex As SqlClient.SqlException
                        MessageBox.Show(ex.Message)
                    Finally
                        Terminate()
                    End Try
                End If
            End If

    End Sub
```

This part of coding is taking from the frmViewSchedule.vb. It show how “View Schedule Details” working in the system

```
Private Sub btnSearches_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
btnSearches.Click
        dgvDetails.Rows.Clear()

        Dim dsDestination As New DataSet()

        Try
            Initialize()

            Dim sqlQuery As String = "SELECT DISTINCT [BUS SYSTEM].dbo .PACKAGES .[PACKAGEID],[BUS
SYSTEM].dbo .PACKAGES .DEPARTUREPLACE, [BUS SYSTEM].dbo .PACKAGES.DESTINATION ,[BUS
SYSTEM].dbo .PACKAGES .PRICE ,[BUS SYSTEM].dbo .PACKAGES.DURATION ,[BUS
SYSTEM].dbo .PACKAGES .DISTANCE, [BUS SYSTEM].dbo .PACKAGES .DEPARTURETERMINAL ,[BUS
SYSTEM].dbo .PACKAGES .ARRIVALTERMINAL  ,[BUS SYSTEM].dbo.[SCHEDULE].DAYID,[BUS
SYSTEM].dbo.[DAY].DAY ,[BUS SYSTEM].dbo.[SCHEDULE].TIMEID, CONVERT(VARCHAR(5),[BUS
SYSTEM].dbo.TIMES1.TIMES,108)AS TIMES,[BUS SYSTEM].dbo.[SCHEDULE].STAFFID,[BUS
SYSTEM] .dbo .STAFF.NAME,[BUS SYSTEM].dbo.[BUS].BUSNO FROM (((([BUS SYSTEM].dbo.PACKAGES INNER JOIN
[BUS SYSTEM].dbo.[SCHEDULE]ON [BUS SYSTEM] .dbo .PACKAGES .PACKAGEID =[BUS
SYSTEM] .dbo .SCHEDULE .PACKAGEID) INNER JOIN [BUS SYSTEM] .dbo .[DAY] on [BUS
SYSTEM] .dbo .[DAY]  .DAYID =[BUS SYSTEM] .dbo .[SCHEDULE]  .DAYID ) Inner Join [BUS
SYSTEM] .dbo .TIMES1 ON [BUS SYSTEM] .dbo .[SCHEDULE].TIMEID =[BUS SYSTEM] .dbo .TIMES1.TIMEID)
INNER JOIN [BUS SYSTEM] .dbo .STAFF ON  [BUS SYSTEM] .dbo .STAFF.STAFFID =[BUS
SYSTEM] .dbo .SCHEDULE.STAFFID) INNER JOIN [BUS SYSTEM] .dbo .BUS ON [BUS
SYSTEM] .dbo .STAFF .STAFFID =[BUS SYSTEM] .dbo .BUS .STAFFID WHERE [BUS SYSTEM] .dbo .STAFF .NAME
='" & cboDriverName.Text.ToUpper & "'"

            Dim adpDestination As New SqlClient.SqlDataAdapter(sqlQuery, conBus)

            adpDestination.Fill(dsDestination, "PackagesTable")

            If dsDestination.Tables("PackagesTable").Rows.Count > 0 Then

                Dim intCount As Integer = 0
                For Each iPlace As DataRow In dsDestination.Tables("PackagesTable").Rows
                    With dgvDetails
                        .Rows.Add()
                        .Rows(.Rows.Count - 1).Cells(0).Value = iPlace.Item("TIMEID")
                        .Rows(.Rows.Count - 1).Cells(1).Value =
CDate(iPlace.Item("TIMES")).ToShortTimeString
                        .Rows(.Rows.Count - 1).Cells(2).Value = iPlace.Item("DAYID")
                        .Rows(.Rows.Count - 1).Cells(3).Value = iPlace.Item("DAY")
                        .Rows(.Rows.Count - 1).Cells(4).Value = iPlace.Item("DEPARTUREPLACE")
                        .Rows(.Rows.Count - 1).Cells(5).Value = iPlace.Item("DESTINATION")
                    End With

                    lblBusNo.Text = iPlace.Item("BUSNO")
                Next
            Else
                MessageBox.Show("This driver do not have any schedule ")
            End If

        Catch ex As SqlClient.SqlException
            MessageBox.Show(ex.Message)
        Finally
            Terminate()
```

```
        End Try

    End Sub
```

This part of coding is taking from the frmReservation.aspx. It show how "Credit Card Validation" working in the system

```
Protected Sub imgBtnPaid_Click(ByVal sender As Object, ByVal e As
System.Web.UI.ImageClickEventArgs) Handles imgBtnPaid.Click
        Dim bolcheck As Boolean

        If txtCCNo.Text = "" Then
            bolcheck = False
        ElseIf DropDownListCreditCardType.Text = "MasterCard" And txtCCNo.Text.Substring(0, 1) <>
"5" And txtCCNo.Text.Length <> 16 Then
            bolcheck = False
        ElseIf DropDownListCreditCardType.Text = "Visa" And txtCCNo.Text.Substring(0, 1) <> "4" And
txtCCNo.Text.Length <> 16 Then
            bolcheck = False
        ElseIf DropDownListCreditCardType.Text = "Diners" And txtCCNo.Text.Substring(0, 1) <> "3"
And txtCCNo.Text.Length <> 14 Then
            bolcheck = False
        ElseIf DropDownListCreditCardType.Text = "AMEX" And txtCCNo.Text.Substring(0, 1) <> "3" And
txtCCNo.Text.Length <> 15 Then
            bolcheck = False
        Else
            Dim sum As Int32 = 0
            Dim cdigit As String
            Dim i, digit As Int32
            Dim nDigit As Int32
            Dim parity As Int32
            nDigit = txtCCNo.Text.Length
            parity = nDigit Mod 2
            For i = 0 To nDigit - 1
                cdigit = txtCCNo.Text.Chars(i)
                digit = Integer.Parse(cdigit)
                If i Mod 2 = parity Then
                    digit = digit * 2
                End If
                If digit > 9 Then
                    digit = digit - 9
                End If
                sum = sum + digit
            Next

            If sum Mod 10 = 0 Then
                bolcheck = True
            Else
                bolcheck = False
            End If
        End If

        Dim intMonth As Integer
        Dim intYear As Integer

        If DropDownListCreditCardType.SelectedIndex = -1 Then
```

```
            lblMsg2.Visible = True
            lblMsg2.Text = "Please select a credit card type!"
        ElseIf bolcheck = False Then
            lblMsg2.Visible = True
            lblMsg2.Text = "Wrong credit card number!"
        ElseIf DropDownListBank.Text = "" Then
            lblMsg2.Visible = True
            lblMsg2.Text = "Please select a bank"
        ElseIf Not IsNumeric(txtExpMonth.Text) Or Not IsNumeric(txtExpYear.Text) Or
txtExpMonth.Text.Length <> 2 Or txtExpYear.Text.Length <> 2 Then
            lblMsg2.Visible = True
            lblMsg2.Text = "Wrong credit card expire date format!"
        ElseIf IsNumeric(txtExpMonth.Text) And IsNumeric(txtExpYear.Text) And
txtExpMonth.Text.Length = 2 And txtExpYear.Text.Length = 2 And DropDownListCreditCardType.Text <>
"" And DropDownListBank.Text <> "" And bolcheck = True Then
            intMonth = Integer.Parse(txtExpMonth.Text)
            intYear = Integer.Parse(txtExpYear.Text)

        Dim intMonth As Integer
        Dim intYear As Integer

        If DropDownListCreditCardType.SelectedIndex = -1 Then
            lblMsg2.Visible = True
            lblMsg2.Text = "Please select a credit card type!"
        ElseIf bolcheck = False Then
            lblMsg2.Visible = True
            lblMsg2.Text = "Wrong credit card number!"
        ElseIf DropDownListBank.Text = "" Then
            lblMsg2.Visible = True
            lblMsg2.Text = "Please select a bank"
        ElseIf Not IsNumeric(txtExpMonth.Text) Or Not IsNumeric(txtExpYear.Text) Or
txtExpMonth.Text.Length <> 2 Or txtExpYear.Text.Length <> 2 Then
            lblMsg2.Visible = True
            lblMsg2.Text = "Wrong credit card expire date format!"
        ElseIf IsNumeric(txtExpMonth.Text) And IsNumeric(txtExpYear.Text) And
txtExpMonth.Text.Length = 2 And txtExpYear.Text.Length = 2 And DropDownListCreditCardType.Text <>
"" And DropDownListBank.Text <> "" And bolcheck = True Then

            intMonth = Integer.Parse(txtExpMonth.Text)
            intYear = Integer.Parse(txtExpYear.Text)

            Dim dtYear As Integer = Integer.Parse(DateTime.Now.Year.ToString.Substring(2, 2))
            Dim dtMonth As Integer = DateTime.Now.Month

            If intMonth > 12 Or intMonth = 0 Then
                lblMsg2.Visible = True
                lblMsg2.Text = "Wrong credit card expire date format!"
            ElseIf intYear = 0 Or intYear < dtYear Then
                lblMsg2.Visible = True
                lblMsg2.Text = "Wrong credit card expire date format!"
            ElseIf intYear = dtYear And intMonth < dtMonth Then
                lblMsg2.Visible = True
                lblMsg2.Text = "Wrong credit card expire date format!"
            Else
                lblMsg2.Text = ""
            End If

    End Sub
```

This part of coding is taking from the frmArrangeDestination.vb. It show how “Find Available Driver” working in the system

```
Public Sub FindAvailable()

        Dim dsStaff As New DataSet()

        Dim arryStaff As ArrayList = New ArrayList
        Dim arryBusStaff As ArrayList = New ArrayList

        Try
            Initialize()

            Dim sqlQuery As String = "SELECT * FROM STAFF WHERE POSITION = 'DRIVER'"

            Dim adpTime As New SqlClient.SqlDataAdapter(sqlQuery, conBus)

            adpTime.Fill(dsStaff, "StaffTable")

            If dsStaff.Tables("StaffTable").Rows.Count > 0 Then
                For Each iPlace As DataRow In dsStaff.Tables("StaffTable").Rows
                    arryStaff.Add(iPlace.Item("NAME"))
                Next

                dsStaff = Nothing
            Else
                MessageBox.Show("This name not in the Database")
            End If

        Catch ex As SqlClient.SqlException
            MessageBox.Show(ex.Message)
        Finally
            Terminate()
        End Try

        Dim dsStaff2 As New DataSet()

        Dim arryStaffID As ArrayList = New ArrayList

        Try
            Initialize()

            Dim sqlQuery1 As String = "SELECT * FROM SCHEDULE"

            Dim adpTime As New SqlClient.SqlDataAdapter(sqlQuery1, conBus)

            adpTime.Fill(dsStaff2, "BusTable")

            If dsStaff2.Tables("BusTable").Rows.Count > 0 Then
                For Each iPlace3 As DataRow In dsStaff2.Tables("BusTable").Rows
                    arryStaffID.Add(iPlace3.Item("STAFFID"))
                Next

                dsStaff2 = Nothing
            Else
                MessageBox.Show("This time not in the Database")
            End If
```

```
        Catch ex As SqlClient.SqlException
            MessageBox.Show(ex.Message)
        Finally
            Terminate()
        End Try

        For intStaffID As Integer = 0 To arryStaffID.Count - 1
            Dim dsStaff3 As New DataSet()
            Try
                Initialize()
                Dim sqlQuery2 As String = "SELECT * FROM STAFF WHERE STAFFID = '" &
arryStaffID.Item(intStaffID) & "'"

                Dim adpTime2 As New SqlClient.SqlDataAdapter(sqlQuery2, conBus)

                adpTime2.Fill(dsStaff3, "StaffTable")

                If dsStaff3.Tables("StaffTable").Rows.Count > 0 Then
                    For Each iPlace2 As DataRow In dsStaff3.Tables("StaffTable").Rows
                        arryBusStaff.Add(iPlace2.Item("NAME"))
                        dsStaff3 = Nothing
                    Next

                Else
                    MessageBox.Show("This name not in the Database")
                End If
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
            Terminate()
        Next

        If arryStaff.Count <> 0 And arryBusStaff.Count <> 0 Then
            For intCount As Integer = 0 To arryBusStaff.Count - 1
                arryStaff.Remove(arryBusStaff.Item(intCount))
            Next
        End If
        cboDriverName.Items.Clear()

        For intAvailable As Integer = 0 To arryStaff.Count - 1
            cboDriverName.Items.Add(arryStaff.Item(intAvailable).ToString)
        Next

    End Sub
```

This part of coding is taking from the frmBus.vb. It show how “Check Bus Number” working in the system

```
Private Sub btnCheck_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
btnCheck.Click

        If txtBusNoCheck.Text = "" Then
            MessageBox.Show("Please key in bus number!", "ERROR", MessageBoxButtons.OK,
MessageBoxIcon.Error)
        Else
            Dim response As DialogResult
```

```
            Dim dsBus As New DataSet()

            Dim sqlQuery As String = "SELECT * FROM BUS WHERE BUSNO='" & txtBusNoCheck.Text.ToUpper
& "'"

            Dim adpBus As New SqlClient.SqlDataAdapter(sqlQuery, conBus)

            adpBus.Fill(dsBus, "BusTable")

            If dsBus.Tables("BusTable").Rows.Count > 0 Then

                Dim BusRow As DataRow

                BusRow = dsBus.Tables("BusTable").Rows(0)

                strID = BusRow.Item("BUSID")
                strNo = BusRow.Item("BUSNO")
                strSeat = BusRow.Item("SEAT")
                strEngineModel = BusRow.Item("ENGINEMODEL")
                strChassisNo = BusRow.Item("CHASSISNO")
                strManufacturer = BusRow.Item("MANUFACTURER")
                strModel = BusRow.Item("MODEL")
                strColor = BusRow.Item("COLOR")
                strRegisterNo = BusRow.Item("REGISTERNO")
                strMadeIn = BusRow.Item("MADEIN")
                strMake = BusRow.Item("MAKE")
                strPrice = BusRow.Item("PRICE")
                strFuel = BusRow.Item("FUEL")
                strEngineCapacity = BusRow.Item("ENGINECAPACITY")
                strBodyBuilder = BusRow.Item("BODYBUILDER")
                strPurchaseYear = BusRow.Item("PURCHASEYEAR")
                strTransmissionModel = BusRow.Item("TRANSMISSIONMODEL")
                strAirConModel = BusRow.Item("AIRCONMODEL")
                strStatus = BusRow.Item("STATUS")
                If BusRow.Item("STAFFID") Is DBNull.Value Then
                Else
                    strSID = BusRow.Item("STAFFID")
                End If

                lblCheck.Text = "Invalid Bus"
                lblCheck.ForeColor = Color.Pink

                MessageBox.Show("This bus number already in the database!", "INVALID",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)

            ElseIf dsBus.Tables("BusTable").Rows.Count = 0 Then
                lblCheck.Text = "Valid Bus"
                lblCheck.ForeColor = Color.GreenYellow
                response = MessageBox.Show("Add new bus?", "VALID", MessageBoxButtons.YesNo,
MessageBoxIcon.Question)

                If response = Windows.Forms.DialogResult.Yes Then

                    Dim BusRow As Integer
                    Dim BusNo As String
                    Dim Number As String
                    Dim dsBus1 As New DataSet()
```

```
                    Dim sqlQuery1 As String = "SELECT BUSID FROM BUS"

                    Dim adpBus1 As New SqlClient.SqlDataAdapter(sqlQuery1, conBus)

                    adpBus1.Fill(dsBus1, "BusesTable")

                    If dsBus1.Tables("BusesTable").Rows.Count = 0 Then
                        BusNo = "B0001"
                    Else

                        BusRow = dsBus1.Tables("BusesTable").Rows.Count - 1

                        Number = dsBus1.Tables("BusesTable").Rows(BusRow).Item(0)

                        BusNo = "B" & (Integer.Parse(Number.Substring(1)) + 1).ToString("D4")

                        lblBusID.Text = BusNo

                    End If
                End If
            End If
        End If

    End Sub
```

This part of coding is taking from the frmReservation.aspx. It show how “Reservation Confirmation Notes in PDF File” working in the system

```
                Dim rptPrintReservation As New rptReservation
                Dim crParameterFieldDefinitions As ParameterFieldDefinitions
                Dim crParameterFieldDefinition As ParameterFieldDefinition
                Dim crParameterValues As New ParameterValues
                Dim crParameterDiscreteValue As New ParameterDiscreteValue

                crParameterDiscreteValue.Value = lblReservation.Text
                crParameterFieldDefinitions = rptPrintReservation.DataDefinition.ParameterFields
                crParameterFieldDefinition = crParameterFieldDefinitions.Item("reservationNo")
                crParameterValues = crParameterFieldDefinition.CurrentValues

                crParameterValues.Clear()
                crParameterValues.Add(crParameterDiscreteValue)
                crParameterFieldDefinition.ApplyCurrentValues(crParameterValues)


                Dim tbCurrent As CrystalDecisions.CrystalReports.Engine.Table
                Dim tliCurrent As CrystalDecisions.Shared.TableLogOnInfo
                For Each tbCurrent In rptPrintReservation.Database.Tables
                    tliCurrent = tbCurrent.LogOnInfo
                    With tliCurrent.ConnectionInfo
                        .ServerName = "KANGLI-PC"
                        .DatabaseName = "BUS SYSTEM"
                        .UserID = "Admin"
                        .Password = "sql"
                    End With
                    tbCurrent.ApplyLogOnInfo(tliCurrent)
                Next tbCurrent
```

```
                Try
                    Dim CrExportOptions As ExportOptions
                    Dim CrDiskFileDestinationOptions As New  _
                    DiskFileDestinationOptions()
                    Dim CrFormatTypeOptions As New PdfRtfWordFormatOptions()
                    CrDiskFileDestinationOptions.DiskFileName = "D:\Report\" & lblReservation.Text
& ".pdf"
                    CrExportOptions = rptPrintReservation.ExportOptions
                    With CrExportOptions
                        .ExportDestinationType = ExportDestinationType.DiskFile
                        .ExportFormatType = ExportFormatType.PortableDocFormat
                        .DestinationOptions = CrDiskFileDestinationOptions
                        .FormatOptions = CrFormatTypeOptions
                    End With
                    rptPrintReservation.Export()

                    System.Diagnostics.Process.Start("D:\Report\" & lblReservation.Text & ".pdf")

                Catch ex As Exception
                    Response.Write(ex.Message)
                End Try
```

This part of coding is taking from the frmReservation.aspx. It show how “Find not Available Seat” working in the system

```
            Dim btnButton As Button

            For intIndex As Integer = 1 To 44
                btnButton = pnlSeat.FindControl("Button" + intIndex.ToString)
                For intArray As Integer = 0 To arryReservationNotAvailable.Count - 1
                    If btnButton.Text = arryReservationNotAvailable.Item(intArray).ToString Then
                        btnButton.BackColor = Drawing.Color.Red
                    End If
                Next
            Next
```

This part of coding is taking from the frmReservation.aspx. It show how “Auto Generate Reservation ID” working in the system

```
            Dim ReserveRow As Integer
            Dim ReserveNo As String
            Dim Number As String
            Dim dsReserve1 As New DataSet()

            Dim sqlQuery1 As String = "SELECT RESERVATIONNO FROM RESERVATION ORDER BY RESERVATIONNO
DESC"

            Dim adpReserve1 As New SqlClient.SqlDataAdapter(sqlQuery1, conBus)

            adpReserve1.Fill(dsReserve1, "ReserveTable")

            If dsReserve1.Tables("ReserveTable").Rows.Count = 0 Then
```

```
            ReserveNo = "R0001"
        Else

            ReserveNo = dsReserve1.Tables("ReserveTable").Rows.Count - 1

            Number = dsReserve1.Tables("ReserveTable").Rows(ReserveRow).Item(0)

            ReserveNo = "R" & (Integer.Parse(Number.Substring(1)) + 1).ToString("D4")

            lblReservation.Text = ReserveNo

        End If
```

This part of coding is taking from the frmStaff.vb. It show how "Insert image to database" working in the system

```
                    Dim ms As MemoryStream = New MemoryStream()

                    PicStaff.Image.Save(ms, PicStaff.Image.RawFormat)

                    Dim data As Byte() = ms.GetBuffer()

                    Dim p As New SqlParameter("@photo", SqlDbType.Image)

                    p.Value = data

                    cmd.Parameters.Add(p)
```

This part of coding is taking from the frmStaff.vb. It show how "Retrieve image from database" working in the system

```
                    Dim dataImage As Byte() = CType(StaffRow.Item("PHOTO"), Byte())

                    Dim ms As MemoryStream = New MemoryStream(dataImage)
                    With PicStaff
                        .Image = Image.FromStream(ms)
                    End With
```

4.4 Chapter Summary

This chapter discussed about the programming used in developing the system. Two-tier architecture and programming language used. Those coding will show how the system work using VB.Net such as add, edit, delete and view.

Chapter 5

Software Testing

5 Software Testing

5.1 Introduction

By preparing the pre-system testing and post system testing, we can detect the bugs, runtime error, and logic error so that we can take necessary corrective actions in time. A prefect system must be tested for several times to ensure it is bugs-free and achieve high quality.

5.2 Sample Data

Table Name: BRANCH

BRANCHID	BR001	BR002	BR003	BR004	BR005
COUNTER	PUDURAYA	TERMINAL BERSEPADU SELATAN	PEKELILING	KUALA PERLIS	KANGAR
ADDRESS	HENTIAN PUDURAYA JALAN PUDU	JALAN TERMINAL SELATAN	STESEN BAS PEKELILING	TERMINAL BAS EKSPRES	JALAN BUKIT LAGI
POSTCODE	51000	57100	50400	02000	01000
STATE	KUALA LUMPUR	KUALA LUMPUR	KUALA LUMPUR	PERLIS	PERLIS
CONTACTNO	03-20702617	03-90575804	03-40421256	05-6881690	05-6881690
EMAIL	PUDURAYA @HOTMAIL. COM	CUSTOMER CARE@TBS BTS.COM.M Y	PEKELILIN G@CUSTO MERSERVI CE.COM.M Y	TERMINAL PERLIS@K PERLIS.CO M.MY	KANGAR@ CUSTOMER SERVICES. COM.MY

Table Name: BUS

BUSID	B0001	B0002	B0003	B0004	B0005
BUSNO	WAB1223	WAD1223	WAD1233	WAG1112	WAG1332
SEAT	33	33	44	44	44

ENGINE MODEL	CUMMINS M11-305E	CUMMINS 6BT(130BHP)	CUMMINS LT10(EURO-I, INLINE 6-CYLINDER REAR-MOUNTED 252BHP, TURBOCHARGED)	MB OM 447H (250BHP @ 2200RPM, 890NM @ 1300RPM, REAR-MOUNTED)	SCANIA DC902(INLINE 6-CYLINDER REAR-MOUNTED, 280BHP, TURBOCHARGED)
CHASSIS NO	YZ6611Q	YZ6610Q	YZ6612Q	YZ6613Q	YZ6614Q
MANUFACTURER	DENNIS SPECIALIST VEHICLE LTD, UK	DENNIS SPECIALIST VEHICLE LTD, UK	LEYLAND MOTOR LTD, UK	MERCEDES-BENZ AG, GERMANY	SCANIA CV AB, SWEDEN
MODEL	TRIDENT	DART AUTO	OLYMPIAN	O405	L94UB4X2
COLOR	PURPLE	PURPLE	PURPLE	PURPLE	PURPLE
REGISTERNO	SBS9671E - SBS9690A	SBS8009A - SBS8018Z	SBS9000S - SBS9199C	SBS82P C SBS281G(WALTER ALEXANDER BODIED)	SBS2888T
MADEIN	1999 - 2000	1993 - 1994	1993	1989 - 1991	1998
MAKE	DENNIS	DENNIS	LEYLAND	MERCEDES-BENZ	SCANIA
PRICE	1200000	1200000	1100000	1100000	1100000
FUEL	DIESEL	DIESEL	DIESEL	DIESEL	DIESEL
ENGINECAPACITY	10,824CC	5,883CC	10,014CC	11,967CC	8,974CC
BODYBUILDER	DUPLE METSEC 5000	DUPLE METSEC	WALTER ALEXANDER ROYALE	WALTER ALEXANDER(WA)	VOLGREN CR221L
PURCHA	2000	1994	1994	1991	1998

SEYEAR					
TRANSMISSIONMODEL	VOITH DIWA863.3	ALLISON AT545	ZF 4HP500(4-SPEED AUTOMATIC)	W3E 110/2.2 R	ZF 4HP 590
AIRCON MODEL	TBC	SUTRAK	NIPPONDENSO	WEBASTO/FUJI (SBS82PC SBS281G)	CARRIER-SUTRAK
STATUS	OK	OK	OK	OK	OK
STAFFID	S0063	S0002	S0001	S0004	S0061
PHOTO	<Binary data>	<Binary data>	<Binary data>	<Binary data>	<Binary data>

Table Name: BUSTENANT

BUSTENANTID	BT001	BT002	BT003	BT004
NAME	JACK	JASSIE	JOEY	SIN
COMPANY	D&D TRAVEL SDN BHD	KENT TRAVEL SDN BHD	ISLAND TRAVEL SDN BHD	SIN@ SDN BHD
IC	590511065005	670809032302	870401305024	870401305022
CONTACTNO	0123049554	0163948332	0122818115	0122818115
EMAIL	JACK@HOTMAIL.COM	JASSIE@HOTMAIL.COM	JOEY@HOTMAIL.COM	swtanswtan382@hotmail.com

Table Name: COMPANYPROFILE

PROFILEID	C0001
BACKGROUND	Kang&Sin Berhad (K&S) is the largest operator of public bus transportation in Malaysia. Kang&Sin Berhad made its debut on Bursa Malaysia on 3 Dec 1999 engages primarily in the bus transportation system such as express bus operations.It express bus operations have provide the most extensive coverage throughout Peninsular Malaysia, and covering all major cities, towns as well as Singapore. The nationwide network is

	serviced more than 2000 buses that cover more than 200 routes and generating 800 trips in a day that around 50 million passengers a year. Kang&Sin Berhad has the largest market share with Kstar being the leader in the express bus industry in Peninsular Malaysia.
ABOUTUS	Kang&Sin(K&S) are further divided according to the bus routes such as Western Region, Eastern Region and Southern Region. For Western Region manages all express routes within this region that stretches from Kuala Lumpur in the south, all the way to Kangar up north. For Eastern Region originating points in Eastern Region are mainly from Kota Bharu, Kuala Terengganu and Kuantan and the Southern Region originating points in Southern Region are mainly from Johor Bahru, Melaka and Seremban. Kstar fleet availability improvement exercise, all new Kstar buses are under the repair and maintenance contract with the original chassis manufacturer. In terms of safety, all new Kstar buses met the European ECE R66 safety regulations on roof crash standards and equipped with front row seat belts since 2005. Besides that, for passengers convenience, Kstar tickets can be purchased 60 days ahead of departure time.
SAFETY	Kang&Sin is collaborating with Automotive Center of Excellence (ACE) in providing professional refresher training for all of its coach captains. Courses are conducted by certified ACE Instructors which require all coach captains to undergo both theoretical and practical classes. Those who have completed the refresher courses then are required to sit for an evaluation test to ensure the driving skills and knowledge are well applied before Certificate of Competency is issued to the participants.
SCHEDULE	NULL
CONTACTUS	NULL

Table Name: DAY

DAYID	D0001	D0002	D0003	D0004	D0005
DAY	MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY

Table Name: FEEDBACK

FEEDBACKNO	F0001
FEEDBACK	POOR SERVICES ON TICKETING
DATE	2011-01-01 00:00:00.000
TIME	2011-02-02 03:00:00.000
MEMBERID	M0001

Table Name: MEMBER

MEMBERID	M0001	M0002	M0003	M0004	M0005
NAME	FOO YU FA	LIM CHEE YUAN	LIANG CHI HAU	OI CHEE YI	LIM ZHEE WEI
IC	760513035613	890530078951	790513035613	590105768913	870917774554
GENDER	MALE	MALE	MALE	MALE	FEMALE
ADDRESS	89, JLN 01, TMN 03	9, JLN 01, TMN 03	7, JLN 01, TMN 03	89, JLN 06, TMN KLANG	67, JLN SEPANG, TMN 03
CITY	SEPANG	SEPANG	SEPANG	KLANG	SEPANG
STATE	SELANGOR	SELANGOR	SELANGOR	SELANGOR	SELANGOR
POSTCODE	48300	48300	48300	40300	48300
CONTACTNO	0127776330	0127126443	0127435967	0166966810	0163214344
EMAIL	FYUFA@GMAIL.COM	CHEEYUAN@GMAIL.COM	CHIHAU@GMAIL.COM	CHEEYI@GMAIL.COM	ZHEEWEI@GMAIL.COM
USERNAME	YUYU	YUAN	HAU	CHEE	WEI
PASSWORD	WE	yua	0003	0004	0005
QUESTION	WHAT IS YOUR FAVOURITE FOOD?	WHAT IS YOUR FAVOURITE SPORTS TEAM?	WHAT IS YOUR FAVOURITE COLOR?	WHAT IS YOUR FAVOURITE COLOR?	WHAT IS YOUR PET NAME?
ANSWER	DONUT	BASAILON	YELLOW	MAGENDA	CUTE

		A			
PHOTO	<Binary data>	<Binary data>	<Binary data>	<Binary data>	<Binary data>

Table Name: ORDERDETAIL

ORDERID	O0001	O0002	O0002	O0002	O0002
PACKAGEID	T0005	T0001	T0001	T0001	T0001
DEPARTURE DATE	2011-03-21 00:00:00.000	2011-02-28 00:00:00.000	2011-02-28 00:00:00.000	2011-02-28 00:00:00.000	2011-02-28 00:00:00.000
DEPARTURE TIME	2011-02-21 07:00:00.000	1900-01-01 07:00:00.000	1900-01-01 07:00:00.000	1900-01-01 07:00:00.000	1900-01-01 07:00:00.000
STATUS	EXIST	EXIST	EXIST	EXIST	EXIST
SUBTOTAL	51.30	12.20	12.20	12.20	12.20
SEATNO	2	11	12	19	20
TICKETNO	TA001	TA002	TA003	TA004	TA005
RETURNID	NONE	NONE	NONE	NONE	NONE

Table Name: ORDERTABLE

ORDERID	O0001	O0002	O0003	O0004	O0005
ORDERDATE	2011-02-01 00:00:00.000	2011-02-28 00:00:00.000	2011-02-28 00:00:00.000	2011-02-28 00:00:00.000	2011-02-28 00:00:00.000
ORDERTIME	1900-01-01 07:00:00.000	2011-02-28 06:13:36.000	2011-02-28 06:13:59.000	2011-02-28 11:55:07.000	2011-02-28 11:56:07.000

Table Name: PACKAGES

PACKAGEID	T0001	T0002	T0003	T0004	T0005
DEPARTURE PLACE	MELAKA	MELAKA	MELAKA	MELAKA	MELAKA
DESTINATION	KUALA LUMPUR	SEREMBAN	JOHOR	KOTA BAHRU	ALOR SETAR
PRICE	12.20	6.80	19.00	52.30	51.30

DURATION	1 HOUR(S) 45 MINUTE(S)	1 HOUR(S) 0 MINUTE(S)	2 HOUR(S) 45 MINUTE(S)	9 HOUR(S) 15 MINUTE(S)	6 HOUR(S) 45 MINUTE(S)
DISTANCE	144 KM	80 KM	224 KM	884 KM	606 KM
DEPARTURE TERMINAL	MELAKA SENTRAL	MELAKA SENTRAL	MELAKA SENTRAL	MELAKA SENTRAL	MELAKA SENTRAL
ARRIVAL TERMINAL	STD BUKIT JALIL	T. 2 SEREMBAN	T.LARKIN JOHOR	T. KOTA BHARU	T.SHAHAB PERDANA

Table Name: PAYMENT

PAYMENTID	P0001	P0002	P0003
PAYMENTMETHOD	CASH	CASH	CASH
PAYMENTSTATUS	WALK IN	WALK IN	WALK IN
CREDITCARDTYPE	NONE	NONE	NONE
CREDITCARDNO	NONE	NONE	NONE
EXPDATE	NONE	NONE	NONE
DISCOUNT	0.00	0.00	5.13
TOTALAMOUNT	51.30	48.80	46.17
PAYDATE	2011-02-21 00:00:00.000	2011-02-28 00:00:00.000	2011-02-28 00:00:00.000
PAYTIME	2011-02-21 09:03:16.000	2011-02-28 06:13:44.000	2011-02-28 06:14:06.000
STAFFID	S0019	S0019	S0019
MEMBERID	Non M	Non M	Non M
ORDERID	O0003	O0002	O0003

Table Name: RENTBUS

RENTORDERID	RB001	RB002	RB003	RB004	RB005
DATE	2011-02-21	2011-02-21	2011-02-21	2011-02-28	2011-02-28
TIME	1900-01-01 11:00:35.000	1900-01-01 11:01:56.000	1900-01-01 12:51:48.000	1900-01-01 11:10:19.000	1900-01-01 11:11:49.000
QUANTITY	2	2	1	2	1

STATUS	PAID	UNPAID	UNPAID	UNPAID	UNPAID
BUSTENANTID	BT002	BT003	BT004	BT004	BT004

Table Name: RENTDETAIL

RENTPACKAGE ID	RP001	RP001	RP001	RP001	RP001
RENTORDERID	RB001	RB002	RB003	RB004	RB005
DATESTART	2011-02-23	2011-02-23	2011-02-22	2011-02-28	2011-02-28
DATERETURN	2011-02-25	2011-03-02	2011-02-23	2011-02-28	2011-02-28
PURPOSE	HG	RWRE	Travel purpose	Travel	Travel
BUSID	B0003	B0003	B0002	B0070	B0041

Table Name: RENTPACKAGES

RENTPACKAGEID	RP001	RP002	RP003	RP004	RP005
DAY	1	2	3	4	5
PRICE	350	650	1000	1350	1700

Table Name: RENTPAYMENT

RENTPAYID	RP001	RP002	RP003	RP004	RP005
PAYMETHOD	CASH	CASH	CASH	CASH	CASH
PAYSTATUS	WALK IN	WALK IN	WALK IN	WALK IN	WALK IN
CREDITCARD	NONE	NONE	NONE	NONE	NONE
EXPDATE	NONE	NONE	NONE	NONE	NONE
TOTAL AMOUNT	4800	350	700	700	700
PAYDATE	2011-02-21 00:00:00.000	2011-02-21 00:00:00.000	2011-02-28 00:00:00.000	2011-02-28 00:00:00.000	2011-02-28 00:00:00.000
PAYTIME	2011-02-21 11:02:11.000	2011-02-21 12:53:17.000	2011-02-28 01:06:40.000	2011-02-28 01:08:37.000	2011-02-28 01:10:58.000
STAFFID	S0019	S0019	S0019	S0019	S0019

RENTORDERID	RB002	RB003	RB001	RB001	RB001
BUSTENANTID	BT001	BT001	BT002	BT002	BT002

Table Name: RESERVATION

RESERVATIONNO	R0001
SEATNO	11
RESERVATIONDATE	2011-02-28 00:00:00.000
RESERVATIONTIME	2011-02-28 09:00:00.000
MEMBERID	M0001
PACKAGEID	T0005
PAYMENTID	P0028
STATUS	RESERVED

Table Name: RETURNTICKET

RETURNID	RT001	RT002
RETURNDATE	2011-02-21 00:00:00.000	2011-02-28 00:00:00.000
RETURNTIME	1900-01-01 12:55:08.000	1900-01-01 01:27:09.000
AMOUNTRETURN	6.10	6.10

Table Name: SCHEDULE

PACKAGEID	T0001	T0001	T0001	T0001	T0001
TIMEID	TM001	TM001	TM001	TM001	TM005
DAYID	D0001	D0003	D0005	D0007	D0002
STAFFID	S0001	S0001	S0001	S0001	S0001
DEPARTURE TIME	1900-01-01 07:00:00.000	1900-01-01 07:00:00.000	1900-01-01 07:00:00.000	1900-01-01 07:00:00.000	1900-01-01 11:00:00.000
ARRIVALTIME	NULL	NULL	NULL	NULL	NULL

Table Name: SERVICECOMPANYDETAIL

COMPANYID	CY001	CY002	CY003	CY004	CY005
NAME	YONG SENG WORKSHOP	YONG LEONG COACH BUILDERS	YOHAN BUS COACH BUILDER CO	PENITARAN SVC CENT SDN BHD	EE SENG AUTO SDN BHD
ADDRESS	NO. 7328 JALAN AIR HITAM BATU 21	NO. 17 BATU 7 JALAN BUKIT KEMUNING KAMPUNG JAWA	PLOT 63 LORONG PERUSAHAAN BUKIT MINYAK 14 BUKIT MINYAK INDUSTRIAL PARK SEBERANG PRAI	927 928 JALAN SIMPANG PERAK MALAYSIA	51 MEDAN KIDD
CITY	KULAI	KLANG	PERAI	SIMPANG	IPOH
STATE	JOHOR	SELANGOR	PENANG	PERAK	PERAK
POSTCODE	81000	42450	14100	34700	30200
CONTACTNO	6076636850	60351212194	6045081053	6058476499	6052547648
EMAIL	YONGSENG@HOTMAIL.COM	YONGLEONG@HOTMAIL.COM	YOHAN@YAHOO.COM	PENITARAN@YAHOO.COM	EESHENG@HOTMAIL.COM

Table Name: SERVICEDETAIL

SERVICEID	SD001	SD002	SD003	SD004	SD005
SERVICEDATE	2011-02-02 00:00:00.000	2011-02-25 00:00:00.000	2011-03-03 00:00:00.000	2011-02-20 00:00:00.000	2011-03-03 00:00:00.000
SERVICETIME	1900-01-01 10:00:00.000	1900-01-01 10:00:00.000	2011-02-24 19:11:24.000	2011-02-24 17:04:32.000	2011-02-28 11:36:19.000
MAINTENANCE	12500.00	1800.00	6555.00	1500.00	0.00

FEES					
DURATION	2 WEEK	2 WEEK	1 WEEK	2 WEEK	
COMPANYID	CY002	CY004	CY009	CY003	CY003
BUSID	B0005	B0008	B0001	B0008	B0026

Table Name: STAFF

STAFFID	S0001	S0002	S0003	S0004	S0005
NAME	CHONG TAN FOOK	LEE WAN WAH	WONG CHONG HANN	LEE CHONG WEI	WONG MIAO CHU
IC	830104063333	790712085421	761124016761	830214038781	840723036242
GENDER	MALE	MALE	MALE	MALE	FEMALE
ADDRESS	N0.1,JLN AA, TMN USJ	A-1-9, ENDAH PURI CONDO	2, JLN HANG LEKUR, TMN LEKUR	4, KG TIGA KAKI BESAR	5, KG TIGA KAKI BEASR
CITY	PJ	SERI PETALING	DAMAI	BUTTERWORTH	BUTTERWORTH
STATE	KUALA LUMPUR	KUALA LUMPUR	KUALA LUMPUR	PULAU PINANG	PULAU PINANG
POSTCODE	56000	57000	54000	21000	21000
CONTACTNO	0124576737	0196562637	0107676366	0127675365	0126767661
POSITION	DRIVER	DRIVER	DRIVER	DRIVER	ADMINISTRATOR
USERNAME	FOOK	WAH	HANN	WEI	CHU
PASSWORD	FOOKFOOK	WAHWAH12	HANNHANN	WEIWEI11	CHUCHU12
QUESTION	WHAT IS YOUR FAVOURITE	WHAT IS YOUR FAVOURITE	WHAT IS YOUR FAVOURITE	WHAT IS YOUR FAVOURITE	WHAT IS YOUR FAVOURITE

	FOOD?	SPORTS TEAM?	COLOR?	COLOR?	COLOR?
ANSWER	FRENCH FRIED	SUNS	YELLOW	PURPLE	RED
EMAIL	CH@HOTMAIL.COM	NH@HOTMAIL.COM	SCM@HOTMAIL.COM	LCW@HOTMAIL.COM	COL@HOTMAIL.COM
SALARY	1670	1680	1600	1750	1800
EPFNO	KR/KK/123A/456	KR/KK/123A/457	KR/KK/123A/458	KR/KK/123A/459	KR/KK/123A/461
SOCSONO	830104063333	790712085421	761124016761	830214038781	840723036242
BANK	MAYBANK	MAYBANK	MAYBANK	MAYBANK	MAYBANK
BANKACCOUNTNO	148001609525	148001609526	148001609527	148001609528	148001609530
PHOTO	<Binary data>	<Binary data>	<Binary data>	<Binary data>	<Binary data>

Table Name: TIMES1

TIMEID	TM001	TM002	TM003	TM004	TM005
PERIOD	MORNING	MORNING	MORNING	MORNING	MORNING
TIMES	2011-02-18 07:00:00.000	2011-02-18 08:00:00.000	2011-02-18 09:00:00.000	2011-02-18 10:00:00.000	2011-02-18 11:00:00.000

5.3 Test Case

Bus Scheduling Module

Add New Schedule

No	Action	Expected Result	Test Result
1.	Enter destination already exist	Message Box prompt “Invalid Destination”	Pass
2.	Leave all field blank	Error Message “Price cannot be empty” is shown	Pass
3.	Select “Cancel” Button	All field empty	Pass
4.	Select “Save” Button	Massage Box prompt “This particular data has	Pass

		been Added"	
5.	Key in wrong format for the price	Message Box prompt "Please key in right format for the price"	Pass

Edit Schedule Details

No	Action	Expected Result	Test Result
1.	Package ID combo box empty	Message Box prompt "Please select a package ID"	Pass
2.	Select same Time, Day and the Driver	Message Box prompt "This time, day and the driver already have the schedule"	Pass
3.	Leave all field blank	Error Message "Please select a time " is shown	Pass
4.	Select "Save" Button	Massage Box prompt "This particular data has been Updated"	Pass

Delete Schedule

No	Action	Expected Result	Test Result
1.	Package ID combo box empty	Message Box prompt "Please select a package ID"	Pass
2.	Select "Delete" Button	Message Box prompt "Confirm delete Schedule?"	Pass
3.	Select "Yes" from the message box	Message box prompt " Data Deleted"	Pass
4.	Select "No" from the message box	Data remain in the form	Pass

View Schedule Details

No	Action	Expected Result	Test Result
1.	Driver name combo box empty	Message Box prompt "Please select a driver name"	Pass
2.	If the driver do not have any schedule	Message Box prompt "This driver do not have any schedule"	Pass

Reservation Module

Add Reservation and Add Reservation Payment

No	Action	Expected Result	Test Result
1.	Select Visa Card and enter "5999999999999"	Message show "Invalid Credit Card Number"	Pass

Repairing Module

Add New Service

No	Action	Expected Result	Test Result
1.	If did not select company name	Message Box prompt "Please select a company name"	Pass
2.	Select "Save" Button	Massage Box prompt "This particular data has been Added"	Pass

Edit Service Details

No	Action	Expected Result	Test Result
1.	Service ID combo box empty	Message Box prompt "Please select a service ID"	Pass
2.	Leave all field blank	Error Message "Please key in the data" is shown	Pass
3.	Select "Save" Button	Massage Box prompt "This particular data has been Updated"	Pass

View Service Details

No	Action	Expected Result	Test Result
1.	Service ID combo box empty	Message Box prompt "Please select a service ID"	Pass

Maintenance Module

Staff Maintenance

Add New Staff

No	Action	Expected Result	Test Result
1.	Staff IC already exist	Message Box prompt “Invalid Applicant”	Pass
2.	Leave all field blank	Error Message “Staff Name cannot be empty” is shown	Pass
3.	Select “Cancel” Button	All field empty	Pass
4.	Select “Save” Button	Massage Box prompt “This particular data has been Added”	Pass
5.	Key in wrong format for the account number	Message Box prompt “Please key in right format for the bank account number”	Pass
6.	Password length less than 8	Message Box prompt “Your password cannot less than 8 character”	Pass
7.	Password does not match	Message Box prompt “Your password does not match”	Pass

Edit Staff Details

No	Action	Expected Result	Test Result
1.	Staff ID combo box empty	Message Box prompt “Please select a staff ID”	Pass
2.	Leave all field blank	Error Message “Staff name cannot be empty” is shown	Pass
3.	Select “Save” Button	Massage Box prompt “This particular data has been Updated”	Pass
4.	Password length less than 8	Message Box prompt “Your password cannot less than 8 character”	Pass
5.	Password does not match	Message Box prompt “Your password does not match”	Pass

Delete Staff

No	Action	Expected Result	Test Result
1.	Staff ID combo box empty	Message Box prompt “Please select a staff ID”	Pass
2.	Select “Delete” Button	Message Box prompt “Confirm delete Staff?”	Pass
3.	Select “Yes” from the message box	Message box prompt “ Data Deleted”	Pass
4.	Select “No” from the message box	Data remain in the form	Pass

View Staff Details

No	Action	Expected Result	Test Result
1.	Staff ID combo box empty	Message Box prompt “Please select a staff ID”	Pass

Member Maintenance

Add New Member

No	Action	Expected Result	Test Result
1.	Member IC already exist	Message Box prompt “Invalid Applicant”	Pass
2.	Leave all field blank	Error Message “Member name cannot be empty” is shown	Pass
3.	Password length less than 8	Message Box prompt “Your password cannot less than 8 character”	Pass
4.	Password does not match	Message Box prompt “Your password does not match”	Pass
5.	Select “Cancel” Button	All field empty	Pass
6.	Select “Save” Button	Massage Box prompt “This particular data has been Added”	Pass

Edit Member Details

No	Action	Expected Result	Test Result
1.	Password length less than 8	Message Box prompt “Your password cannot less than 8 character”	Pass
2.	Password does not match	Message Box prompt “Your password does not match”	Pass
3.	Member ID combo box empty	Message Box prompt “Please select a member ID”	Pass
4.	Leave all field blank	Error Message “Member name cannot be empty” is shown	Pass
5.	Select “Save” Button	Massage Box prompt “This particular data has been Updated”	Pass

Delete Member

No	Action	Expected Result	Test Result
1.	Member ID combo box empty	Message Box prompt “Please select a member ID”	Pass
2.	Select “Delete” Button	Message Box prompt “Confirm delete Member?”	Pass
3.	Select “Yes” from the message box	Message box prompt “ Data Deleted”	Pass
4.	Select “No” from the message box	Data remain in the form	Pass

View Member Details

No	Action	Expected Result	Test Result
1.	Member ID combo box empty	Message Box prompt “Please select a member ID”	Pass

Bus Maintenance

Add New Bus

No	Action	Expected Result	Test Result
1.	Bus number already exist	Message Box prompt “Invalid Bus Number”	Pass
2.	Leave all field blank	Error Message “Engine Model cannot be empty” is shown	Pass
3.	Select “Cancel” Button	All field empty	Pass
4.	Select “Save” Button	Massage Box prompt “This particular data has been Added”	Pass

Edit Bus Details

No	Action	Expected Result	Test Result
1.	Bus ID combo box empty	Message Box prompt “Please select a Bus ID”	Pass
2.	Leave all field blank	Error Message “Please enter engine model ” is shown	Pass
3.	Select “Save” Button	Massage Box prompt “This particular data has been Updated”	Pass

Delete Bus

No	Action	Expected Result	Test Result
1.	Bus ID combo box empty	Message Box prompt “Please select a Bus ID”	Pass
2.	Select “Delete” Button	Message Box prompt “Confirm delete Bus?”	Pass
3.	Select “Yes” from the message box	Message box prompt “ Data Deleted”	Pass
4.	Select “No” from the message box	Data remain in the form	Pass

View Bus Details

No	Action	Expected Result	Test Result
1.	Bus ID combo box empty	Message Box prompt “Please select a Bus ID”	Pass

5.4 Chapter Summary

A well design system should provide an error free and well function for the user. The system may produce undesired output, terminate during run-time or error occurs if not well tested. Through the software testing, every module and fields in the system had been tested carefully; otherwise the adequate message box will prompt to the user for each input. Through the message box, the user also can understand the actions and follow the instruction of the system.

Chapter 6

Conclusion

6 Conclusion

6.1 Introduction

This chapter will conclude about Final Year Project documentation. Evaluation against the project objective of the project, project management issues during development time, future improvement for the project, justification of choice of tools and the personal reflection will be discussed in this chapter.

6.2 Link to Seminar

The seminar title is "Transport Scheduling Intelligent" which is close linked with the bus ticketing system. The bus ticketing system have using the Knowledge-Base System, that proposed by Jones & Bartlett (2009) can be defined as a computer system to provide advice in a specific area and use of human experts for knowledge.

Knowledge-Base System is a distinctive feature of the underlying knowledge, through a variety of ways, such as rules, frames or cases on behalf of the inference engine, or algorithm, which uses the knowledge base to draw a conclusion separation.

For example I used at the scheduling form while they want to add the new schedule then will need follow the rules to add the schedule for the driver in order have a nice schedule.

6.3 Justification of Choice of Tools

I was thinking to use VB.Net instead of using C#.Net in our project because I think the VB.Net is much more function than the C#.Net. Then I discussed with my teammate, she also think that using VB.Net better than the C#.Net so we decide using VB.Net as the programming language to develop this system.

For database server, I and my teammate also agree to use Microsoft SQL Server 2008 instead of other server application because we think the Microsoft SQL Server 2008 is much more flexible and easier to use, even though the Microsoft SQL Server 2008 we never learned before.

6.4 Evaluation against the project objectives

- **Convenient**

  With this system will bring a lot of convenient to the user because they can make reservation at anywhere and any place that have the internet connection.

- **Improve efficiency**

  With this system can improve the efficiency of the work because the staff no need using hand to write down the record or information of the ticket. They just need select the data from the system then click print then all information regarding the ticket will show at the ticket for the consumer and the data will direct insert into the database. If they want retrieve back the data from database just need select the ticket number then click search so the data in the database will show to the user.

- **Increase income**

  With this system will let the company to increase the income for monthly sales because member can make reservation via online so that they no need go to the counter to buy the ticket if they not free and lazy go to the counter just for buy the ticket only.

- **Bus and ticket information**

  With this system the customer can know the information about the bus schedule and ticket so they can view the details via internet. Nowadays, online are very common issues to everyone so that checking information using online can save a lot of time to the customer. Customers no need go to the counter to asking about the information of the bus schedule and the schedule.

- **Reduce error data**

  With this system can help the staff to reduce the error data because they just need select the data from the system so no need key in too much data from the user and reduce error data.

6.5 Project Management Issues

During the development of this project, there some difficulties occurred. So we managed to settle down by looking more deeply into the problems and try to find solutions over the internet. Besides, I feel frustrated when I need to use some new programming logic that I never learned before. Luckily, my friends are willing to guide me and show me the way to achieve it. Besides that, I did a lots research over internet while learning from my friends.

6.6 Future Improvement

As mentioned above, although our system had been completed but it is not perfect, we had planned to make some enhancement in the future. We think that our system still has potential to grow. Besides, we will include more functions and introduce more widgets to the system. We also plan to enhance the interface so that it looks more attractive and interactive.

6.7 Personal reflection

From system planning, requirement analysis, system design, coding and testing, finally this project was done in the given time. I found that time management very important when implementing a system. If the system cannot be done in time, the consequence is system fail and it will cause a massive loss for a company. Besides, working together is better than working individually.

During developing this project, I faced a lot of problem such as installing the Microsoft SQL Server 2008 and project programming. Hours and hours I spent to find the solution and praying for people helping me to solve the problem. Thanks to the internet and my friend, I am able to solve my problems such as retrieving data from database, view it in grid view, insert image to the database and how to use date time picker in web form.

With the help and support from my friends, I am solving the problem and complete the project in time. I also would like to thanks my partners, Tan Sin Wan who is always by my side motivating me and pushing me to complete my project. With her around, procrastination does not happen. I would like to express my sincere gratitude to my project supervisor, Ms Kong who is very kind and considerate to me. She was always giving me opinions and feedback on my project. She also guided me slowly to finish my project.

6.8 Summary

After completing the system, I had gained a lot of extra knowledge that I did not learned in college and it is very useful for me in my future undertaking. I enjoyed working together with my friends, as we are able to push each other and share knowledge. I felt so proud to able to complete my Final Year Project on time as this will be the last project for me in Tunku Abdul Rahman College.

7 References

1. Beansoftware.com. 2010. *Crystal Reports in ASP.NET*. Viewed on: 5 December 2010. Available from: < http://www.beansoftware.com/ASP.NET-Tutorials/Using-Crystal-Reports.aspx >

2. C-sharpcorner.com. 2010. *Adding and retrieving Images from a SQL Server Table*. Viewed on: 2 December 2010. Available from: < http://www.c-sharpcorner.com/UploadFile/mamtam/AddandRetrieveImages09232005080918AM/AddandRetrieveImages.aspx >

3. Elliott, R. & Powers, N., (Intellex), "One -Tier, Two-Tier, Three-Tier, A Server: Using Technology to Solve Business Problems", < http://www.pacific-electric.com/PacificElec/Product/whtpap04.htm >

4. Hemmer, F.M. (1993) "RHIC Electronic Data Collection and survey & Alignment Database" Proceedings of the Third International Workshop On Accelerator Alignment, Annecy, pp 197

5. Sujeev.wordpress.com. 2010. *Changing the default value of "Edit Top 200 Rows" – SQL Server 2008*. Viewed on: 10 December 2010. Available from: < http://sujeev.wordpress.com/2009/07/02/changing-the-default-value-of-edit-top-200-rows-sql-server-2008/ >

6. Vbdotnetheaven.com. 2010. *Save an Image to SQL Server*. Viewed on: 2 December 2010. Available from: < http://www.vbdotnetheaven.com/uploadfile/scottlysle/imagetosqlserver11242006025136am/imagetosqlserver.aspx >

7. Visual-paradigm.com. 2010. *Import Rational Software Architect Project File*. Viewed on: 28 February 2010. Available from: < http://www.visual-paradigm.com/highlight/importrsa.jsp >

8 Appendices

Figure 8.1 Update Bus Form

Figure 8.2 Example of Credit Card

No. 23, Jalan Penjaja 3, 57000 Kuala Lumpur, Malaysia.
Contact: 03-57891128 Fax: 03-57891129

Reservation Confirmation

Reservation ID : R0002

Reservation Date : 3/17/2011

Member ID : M0004
Name : OI CHEE YI
IC : 590105768913
Gender : MALE
Contact No : 0166966810
Email : CHEEYI@GMAIL.COM

Package ID : T0023
Departure From : JOHOR
Arrival At : MELAKA
Departure Date : 04-Mar-2011
Departure Time : 10:00 pm
Price : 19.00
Total Amount : 57.00
Seat No :
10
11
12

* Please Keep this Reservation Confirmation for Collect Ticket from the Counter.

Figure 8.3 Reservation Confirmation Notes

9 User Guide

Option

Users can log in to this Maintenance Menu to choose **Update Staff Information, Update Member Information, Update Bus Information**

If users choose **Update Staff Information**, will be log in to **Staff Maintenance Form**.

If users choose **Update Member Information**, will be log in to **Member Maintenance Form**.

If users choose **Update Bus Information**, will be log in to **Bus Maintenance Form**.

Maintenance Module

Staff Maintenance

Search By

- Select the **Staff ID** then click the **Search** button.
- Select the **Staff Name** then click the **Search** button.

Security

- Key in all the field then click the **Check** button.

Staff Details

- Click the **Add** button and fill out the blank of the details then click the **Save** button.
- Click the **Edit** button and choose the record which data that need to edit then click the **Save** button.
- Click the **Delete** button to choose which staff record need to be delete.
- Click the **Cancel** button if don't want Add new, Edit and Delete the staff record.

Member Maintenance

Search By

- Select the **Member ID** then click the **Search** button.
- Select the **Member Name** then click the **Search** button.

Security

- Key in all the field then click the **Check** button.

Member Details

- Click the **Add** button and fill out the blank of the details then click the **Save** button.
- Click the **Edit** button and choose the record which data that need to edit then click the **Save** button.
- Click the **Delete** button to choose which member record need to be delete.
- Click the **Cancel** button if don't want Add new, Edit and Delete the member record.

Bus Maintenance

Search By

- Select the **Bus ID** then click the **Search** button.
- Select the **Bus No** then click the **Search** button.
- Select the **Driver Name** then click the **Search** button.

Bus Details

- Click the **Add** button and fill out the blank of the details then click the **Save** button.
- Click the **Edit** button and choose the record which data that need to edit then click the **Save** button.
- Click the **Delete** button to choose which bus record need to be delete.
- Click the **Cancel** button if don't want Add new, Edit and Delete the bus record.
- Click the **Send For Service** button, log in to the **Service Information Form**

Option

Users can log in to this Scheduling Menu to choose **View Schedule Details, Assign Schedule**

If users choose **View Schedule Details**, will be log in to **View Schedule Form**.

If users choose **Assign Schedule**, will be log in to **Scheduling Form**.

Bus Scheduling Module

View Schedule

Search By

- Select the **Driver Name** then click the **Search** button.

Assign Schedule

Search By

- Select the **Packages ID** then click the **Search** button.
- Select the **From** where **To** where then click the **Search** button.

Destination Information & Data Grid View

- Click the **Add** button and fill out the blank of the details then click the **Save** button.
- Click the **Edit** button and choose the record which data that need to edit then click the **Save** button.
- Click the **Delete** button to choose which schedule record need to be delete.
- Click the **Cancel** button if don't want Add new, Edit and Delete the schedule record.

Option

Users can log in to this Repairing Menu to choose **Update Service Details**

If users choose **Update Service Details**, will be log in to **Service Information Form**.

Repairing Module

Service Information

Search By

- Select the **Service ID** then click the **Search** button.
- Select the **Bus No** then click the **Search** button.
- Select a **Date** then click the **Search** button.

Service Details

- Select a **Date**

Service Company Details

- Select a **Company Name**
- Click the **Send For Service** button and fill out the blank of the details then click the **Save** button.
- Click the **Edit** button and choose the record which data that need to edit then click the **Save** button.

Reservation Module

- Select the **Member ID**
- Select the **Departure Destination**
- Select the **Arrival Destination**
- Select a **Date** then click the **Search** button.

- Select a **Time** then click the **Next** button.
- Select the **Seat Number** then click the **Next** button.

Payment Details

- Select a **Credit Card Type**
- Key in Key in Credit Card Number
- Select a **Bank**
- Key in **Credit Card Expire Date**
- Click the **Paid** button