

SYNOPSIS

TOURS AND TRAVELS MANAGEMENT SYSTEM

INTRODUCTION

The objective of the project is to develop a system that books the hotels and buses and keeps all the records of a Tours and Travels Management System and is developed as a part of VI Semester Project for the partial fulfilment of the BSc. IT degree.

PROJECT

The proposed system is a desktop based application and maintains a centralized repository of all the details of bookings undertaken. This project is developed in the favour of SHREE TRAVELS' (client).

OBJECTIVE

The main objective of the project is to computerize all details regarding tours details management. This will enable my client to operate all his current operation sitting on a single chair.

SCOPE

This application is used for decreasing the paper work and keeping digitalized records of each and every stock.

PROBLEM IN CURRENT SYSTEM

Looking up on the current system it's a hectic work of using pen and paper each and every time and managing each and every record manually has become a risk which is unreliable. As this current system is not that secure. So, my client is looking for a simple and effective system to solve this problem which may not cause him to conduct any bad relation with his customer.

PROPOSED SYSTEM

The proposed system is a desktop application. Client has to operate it on the system which makes him away from pen and paper and helps to be secure. This System will help client to not to travel anywhere for data will provide all operation at one location.

SYSTEM REQUIREMENTS

- Storing list of all tours information.
- Storing list of Registered Customer's information.
- Admin Login screen for security purpose.
- System needs quality performance.

LANGUAGE USED

The Application is developed in java language.

HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirements

Processor	: Intel Core i3
RAM	: 4 GB
Hard disk	: 20 GB or more
Monitor	: 15''CRT or LCD Monitor
Keyboard	: Normal or Multimedia

Software Requirements

Front End	: NetBeans 8.0.1
Back End	: MySQL Workbench 8.0
Operating System	: Windows 8 and above

Objective and Scope

Objective

- To assist the tours and travels businessmen and shopkeepers capturing the efforts spent on their respective working areas .
- Overcoming the issues of using the pen and paper system.
- Keeping a record of each and every tour received.
- Maintaining details of tours and customers.

Scope

- As this is generic software it can be used by a wide variety of outlets to automate the process of manually maintaining the records related to the subject of maintaining the tours and cash flows.
- This project is basically updating the manual tours and travels management inventory System to Automated inventory system, so that organization can manage their record in efficient and organized form.

THEORETICAL BACKGROUND

- **Lack of storage Space**

Paperwork can take up a significant amount of space, and this requirement will only get bigger as the number of documents increase.

- **Inefficient document transportation**

Transporting documents in a paper-based system is often problematic. While it is possible to fax, courier or deliver files in person, this is slow and inefficient. This is particularly true if you need to send over numerous pages or folders. With an electronic document management system, users can simply add attachments to an email and send information instantly.

- **Hard to Make Changes**

When you are working with paper documents it is much harder to make changes. Every time you want to make a change you will have to make a copy, so you don't destroy the original with any edits or comments you might add.

- **Lack of Security**

Paper based work filing can be less secure than electronic filing systems. Misplaced documents can easily be placed in the wrong hands. Clients expect their information to be secure in your hands.

- **Increase Cost and Time**

Manual document filing is very time consuming. Not only do you have to organize and store the files, hunting down the information when it is need can take time. When you use paper documents your costs are going to be higher because you are paying for ink and paper.

PROBLEM DEFINITION

Defining the problems is one of the important activities of the project. The objective is to define precisely the business problem to be solved and thereby determine the scope of the system.

So, this software provide the authentication for the record to avoid the risk of security.

And software is also provide adding, deleting, modifying and searching of required record are developed rather than finding data in big amount of documents.

This software also decrease time consuming rather than filling document manually.

System study and analysis

The Tours and Travels Management system provides functions on identify tours usages instruction, provides various tours according to the customer's expectations and hotel packages, facilitate accessibility of tours information and information management among employees, providing optimal tours for various customers, enable reports with in significantly short period of time, despite simultaneous usage of database for the purpose stated above. The system will solve the problem of the current system by minimizing time wastage and reduce resources which simply change manual based system to computerized system.

Feasibility Study

What is a feasibility study? As the name implies, a feasibility study is used to determine the viability of an idea, such as ensuring a project is legally and technically feasible as well as economically justifiable. It tells us whether a project is worth the investment. There can be many reasons for this, including requiring too many resources, which not only prevents those resources from performing other tasks but also may cost more than an organization would earn back by taking on a project that isn't profitable.

1. Technical Feasibility :

This assessment focuses on the technical resources available to the organization. My project or application is overall developed in NetBeans IDE which is well known software easy to use. Also I have used MySQL Workbench for back-end purpose which is also very easily available on internet and easy to download. So my application is completely technically feasible.

2. Economic Feasibility:

This assessment typically involves a cost/ benefits analysis of the project, helping my client to determine the viability, cost, and benefits associated with a project. Specially software's used in my project i.e. Netbeans IDE for front end purpose and MySQL Workbench for back-end purpose are free and open source. They are free to use. So, my project is also economically feasible.

3. Marketing Feasibility:

A market feasibility study determines the depth and condition of a particular real estate market and its ability to support a particular development. As my project is Desktop based application it is very feasible. As sitting on one place my client is able to access all his data. My application is able to change according to the different marketing strategies.

4. Operational Feasibility:

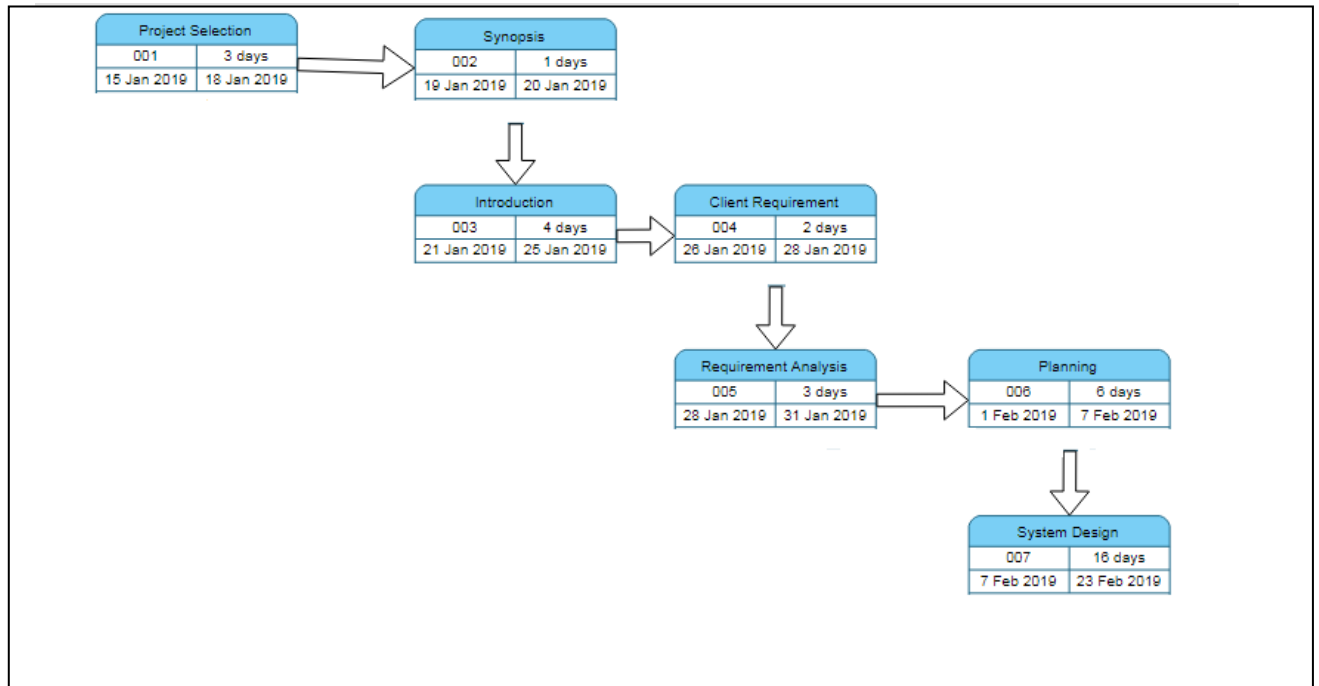
My project is very feasible operationally as it counters all the issues faced by the previous system and digitalizes the current system by removing the usage of paper and pen. In my application my client can also perform any changes according to his wish. It is easily mutable.

5. Behavioural Feasibility

People are inherently resistant to change and computers have been known to facilitate change. My project or application has behavioural feasibility, therefore it is understandable that the introduction of a tours and travels system requires special effort to educate, sell and train the staff on new ways of conducting business.

SYSTEM PLANNING

- **PERT CHART**



Methodology adopted

A methodology is a model, which project managers employ for the design, planning, implementation and achievement of their project objectives. There are different project management methodologies to benefit different projects.

For example, there is a specific methodology, which NASA uses to build a space station while the Navy employs a different methodology to build submarines. Hence, there are different project management methodologies that cater to the needs of different projects spanned across different business domains.

Project Methodologies

Following are the most frequently used project management methodologies in the project management practice:

1 - Adaptive Project Framework

In this methodology, the project scope is a variable. Additionally, the time and the cost are constants for the project. Therefore, during the project execution, the project scope is adjusted in order to get the maximum business value from the project.

2 - Agile Software Development

Agile software development methodology is for a project that needs extreme agility in requirements. The key features of agile are its short-termed delivery cycles (sprints), agile requirements, dynamic team culture, less restrictive project control and emphasis on real-time communication.

3 - Crystal Methods

In crystal method, the project processes are given a low priority. Instead of the processes, this method focuses more on team communication, team member skills, people and interaction. Crystal methods come under agile category.

4 - Dynamic Systems Development Model (DSDM)

This is the successor of Rapid Application Development (RAD) methodology. This is also a subset of agile software development methodology and boasts about the training and documents support this methodology has. This method emphasizes more on the active user involvement during the project life cycle.

5 - Extreme Programming (XP)

Lowering the cost of requirement changes is the main objective of extreme programming. XP emphasizes on fine scale feedback, continuous process, shared understanding and programmer welfare. In XP, there is no detailed requirements specification or software architecture built.

Details of hardware and software

- **Hardware Requirements**

Processor	: Intel Core i3
RAM	: 4 GB
Hard disk	: 20 GB or more
Monitor	: 15''CRT or LCD Monitor
Keyboard	: Normal or Multimedia

- **Software Requirements**

Front End	: NetBeans 8.0.1
Back End	: MySQL Workbench 8.0
Operating System	: Windows 8 and above

System Maintenance and evaluation

System Maintenance / Enhancement Maintenance means restoring something to its original conditions. Enhancement means adding, modifying the code to support the changes in the user specification. System maintenance conforms the system to its original requirements and enhancement adds to system capability by incorporating new requirements.

Thus, maintenance changes the existing system, enhancement adds features to the existing system, and development replaces the existing system. It is an important part of system development that includes the activities which corrects errors in system design and implementation, updates the documents, and tests the data.

- **Maintenance Types**

System maintenance can be classified into three types -

Corrective Maintenance - Enables user to carry out the repairing and correcting leftover problems.

Adaptive Maintenance - Enables user to replace the functions of the programs.

Perfective Maintenance - Enables user to modify or enhance the programs according to the users' requirements and changing needs.

Cost And Benefit Analysis:

When managing a project, one is required to make a lot of key decisions. There is always something that needs executing, and often that something is critical to the success of the venture. Because of the high stakes, good managers don't just make decisions based on gut instinct. They prefer to minimize risk to the best of their ability and act only when there is more certainty than uncertainty.

But how can you accomplish that in a world with myriad variables and constantly shifting economics? The answer: consult hard data collected with reporting tools, charts and spreadsheets. You can then use that data to evaluate your decisions with a process called cost benefit analysis (CBA). An intelligent use of cost benefit analysis will help you minimize risks and maximize gains both for your project and your organization.

What Is Cost Benefit Analysis?

Cost benefit analysis in project management is one more tool in your toolbox. This one has been devised to evaluate the cost versus the benefits in your project proposal. It begins with a list, as so many processes do.

There's a list of every project expense and what the benefits will be after successfully executing the project. From that you can calculate the return on investment (ROI), internal rate of return (IRR), net present value (NPV) and the payback period.

The difference between the cost and the benefits will determine whether action is warranted or not. In most cases, if the cost is 50 percent of the benefits and the payback period is not more than a year, then the action is worth taking.

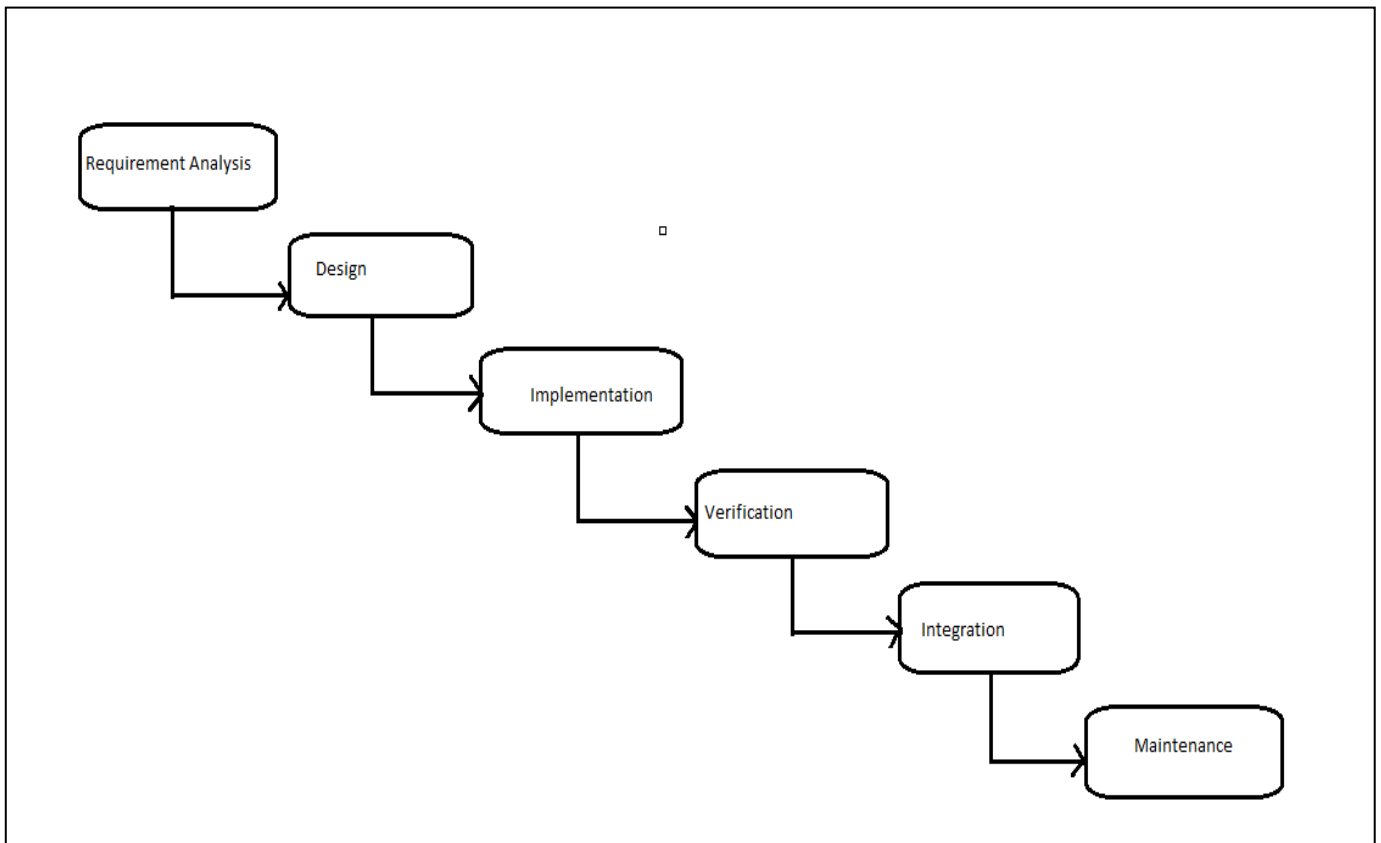
The Purpose of Cost Benefit Analysis:

The purpose of cost benefit analysis in project management is to have a systemic approach to figure out the pluses and minuses of various paths through a project, including transactions, tasks, business requirements and investments. Cost benefit analysis gives you options, and it offers the best approach to achieve your goal while saving on investment.

There are two main purposes in using CBA:

1. To determine if the project is sound, justifiable and feasible by figuring out if its benefits outweigh costs.
2. To offer a baseline for comparing projects by determining which project's benefits are greater than its costs.

Life cycle of the project



1) Requirements analysis

- This document outlines the software requirements for the Tours and Travels management system for the shop- 'Shree Travels'. It describes the functional and non-functional requirements, modelling requirements, diagrams and user profiles of the proposed system. The Tours and Travels Management System enables Shree Travel's staff to maintain computerized Record of Customer Who visits, their Employees & tours. It also performs quick search for their tours, And Keep record of the tours with the packages. This system provides detailed information on the

internal and external view of the system as well as interfaces required by Tours and Travels Management System.

2) Design

- Decide on the software architecture, or the big picture/skeleton.
- Make such an architecture which handles the tour details digitally.

3) Implementation

- The system is implemented in Intel(R) Core(TM) i3 processor with 2 GB RAM, 32-bit computer. And it is implemented through testing on both Black and White testing. The language we use implement the system is NetBeans IDE. The tours and travels management system registers tours stored on the database, modified when additional information is existing, and remove when needed.

4) Verification

- It is a type of testing in which it checks whether the required analysis is analyzed or not.
- It gives a common idea to the inventor whether to deliver his/her system to the client or still make any changes.

5) Integration

- Any familiar in using windows operation can operate the system since it has user friendly user interface. Which have the instruction menu's how to use it which self-directive application then can be used the system without ambiguity.
- Release the software to be used for managing the records of all the tours.
- Sometimes, this step is merged with or implied by step 4.

6) Maintenance

- This Tours and Travels management system operates in any version of windows operating system. Such as windows xp, windows 2003, windows 7, windows 8 and other related versions. The system can be easily maintained by the manager of the tours and travels shops by using the prepared documents of the system for easy maintenance. Other ways it is maintained by the system developers for corrective and other heavy problems.
- Fix any bugs or issues that come up during operations and try them to maintain them regularly.

Data Flow Diagrams

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated DFDs can also be used for the visualization of data processing (structured design).

Context Diagram-

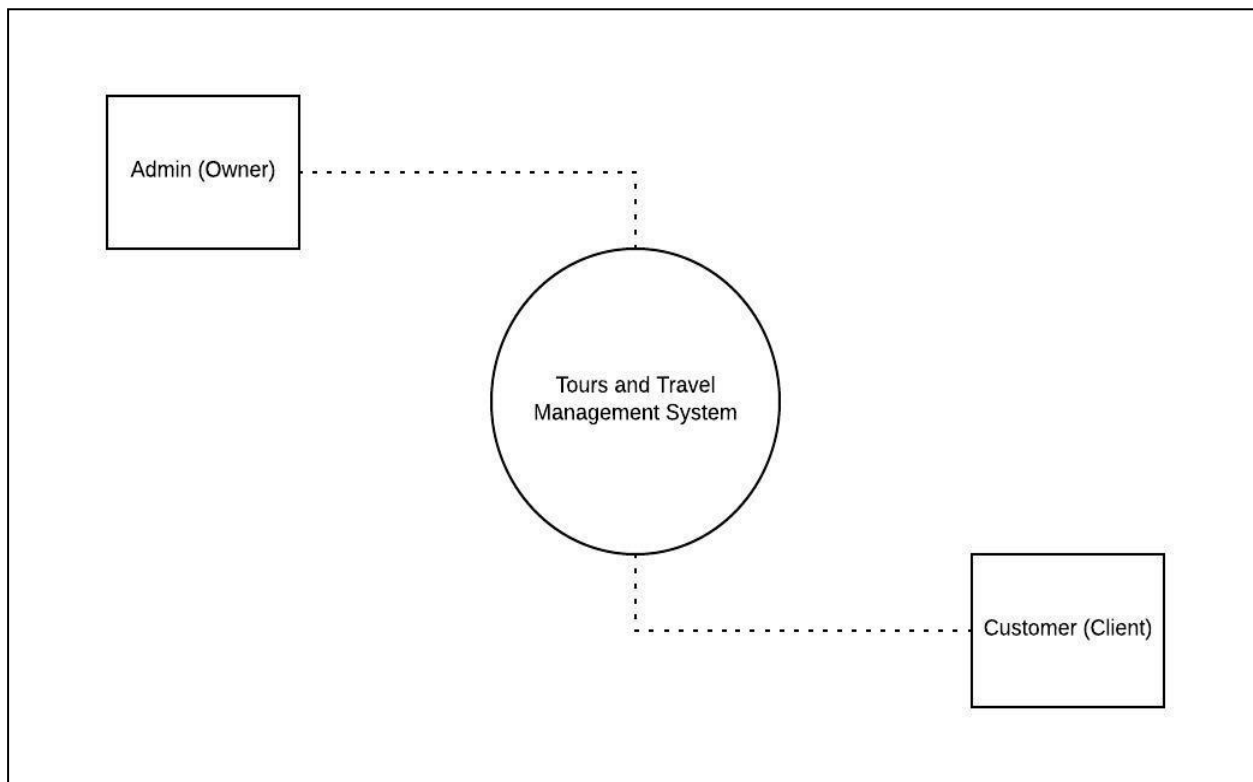
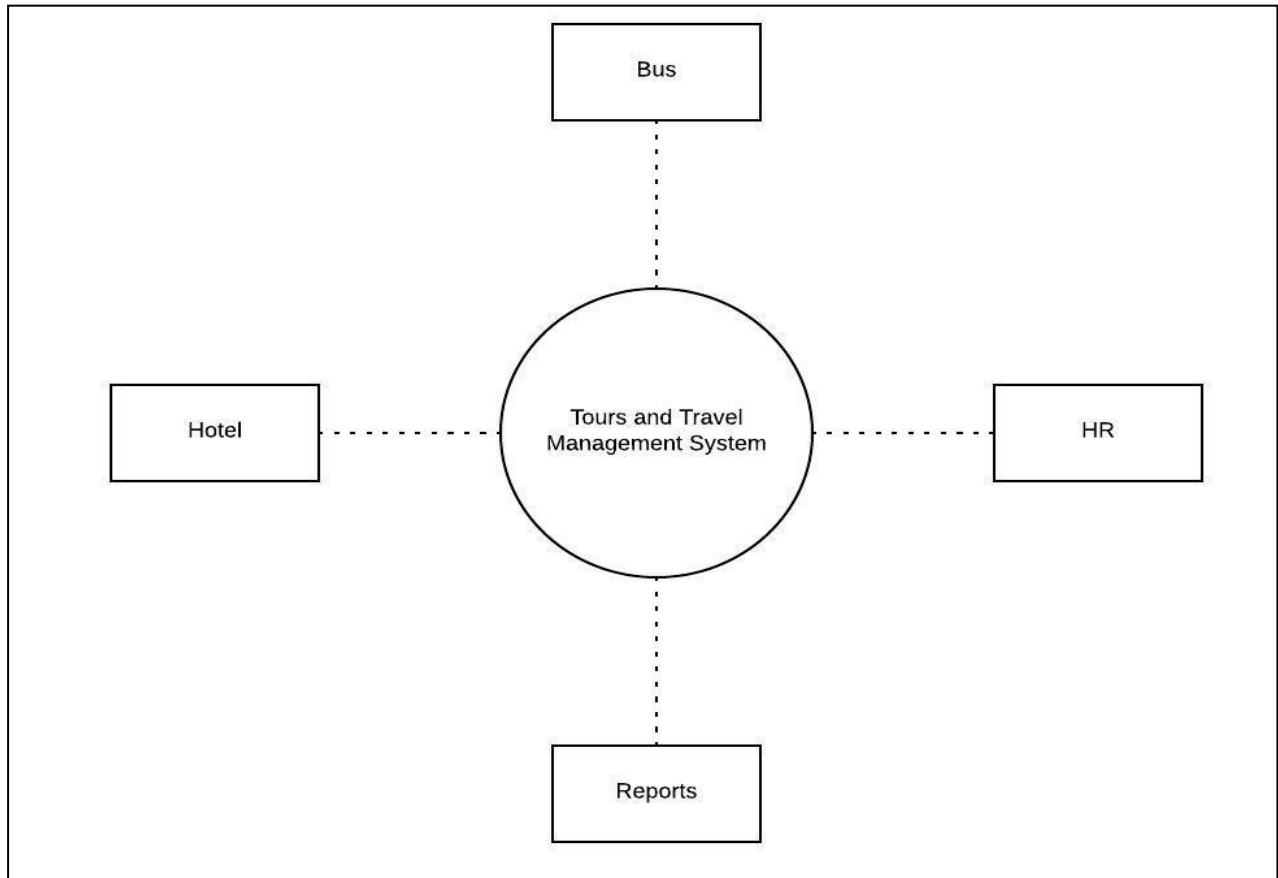
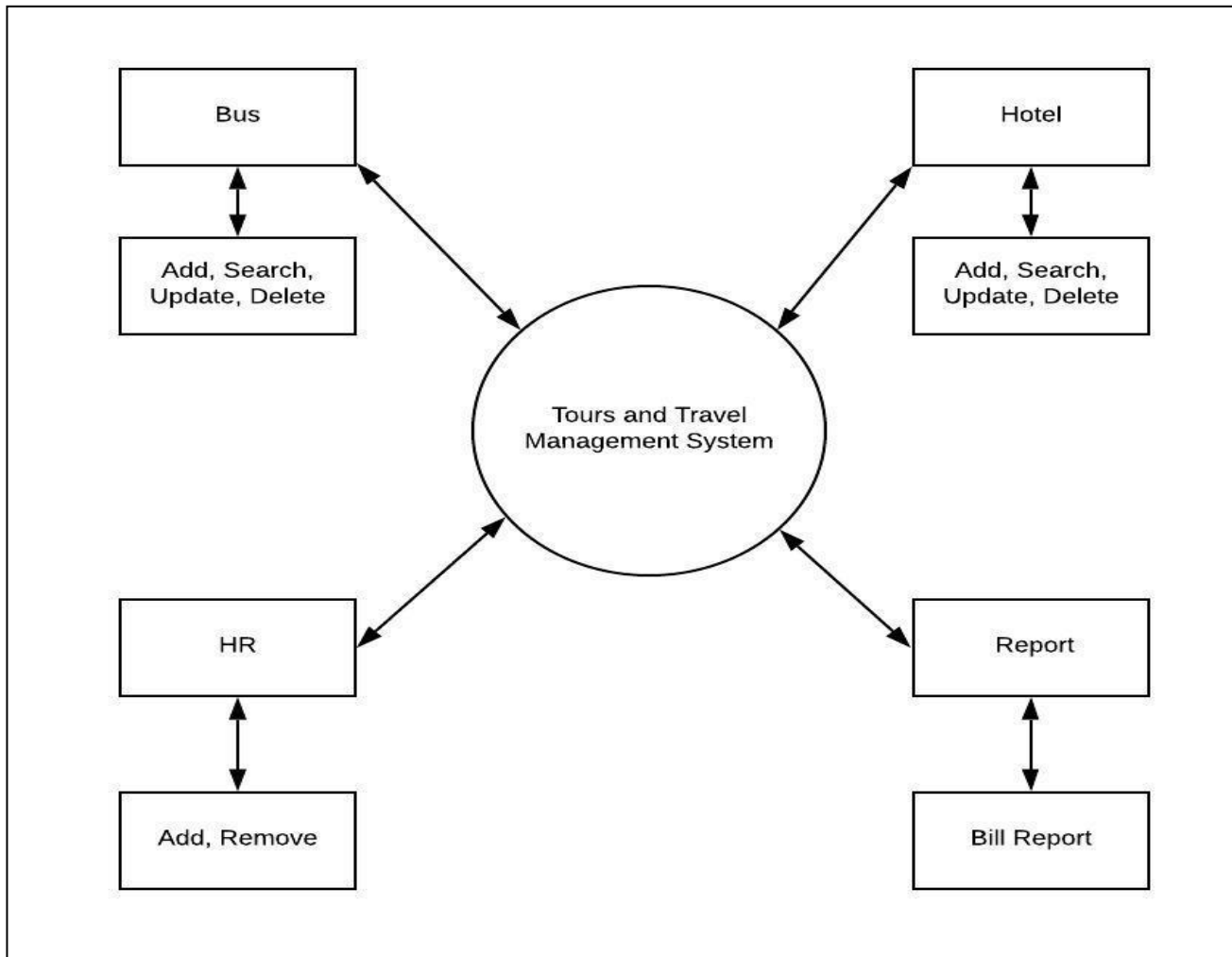


Diagram 0

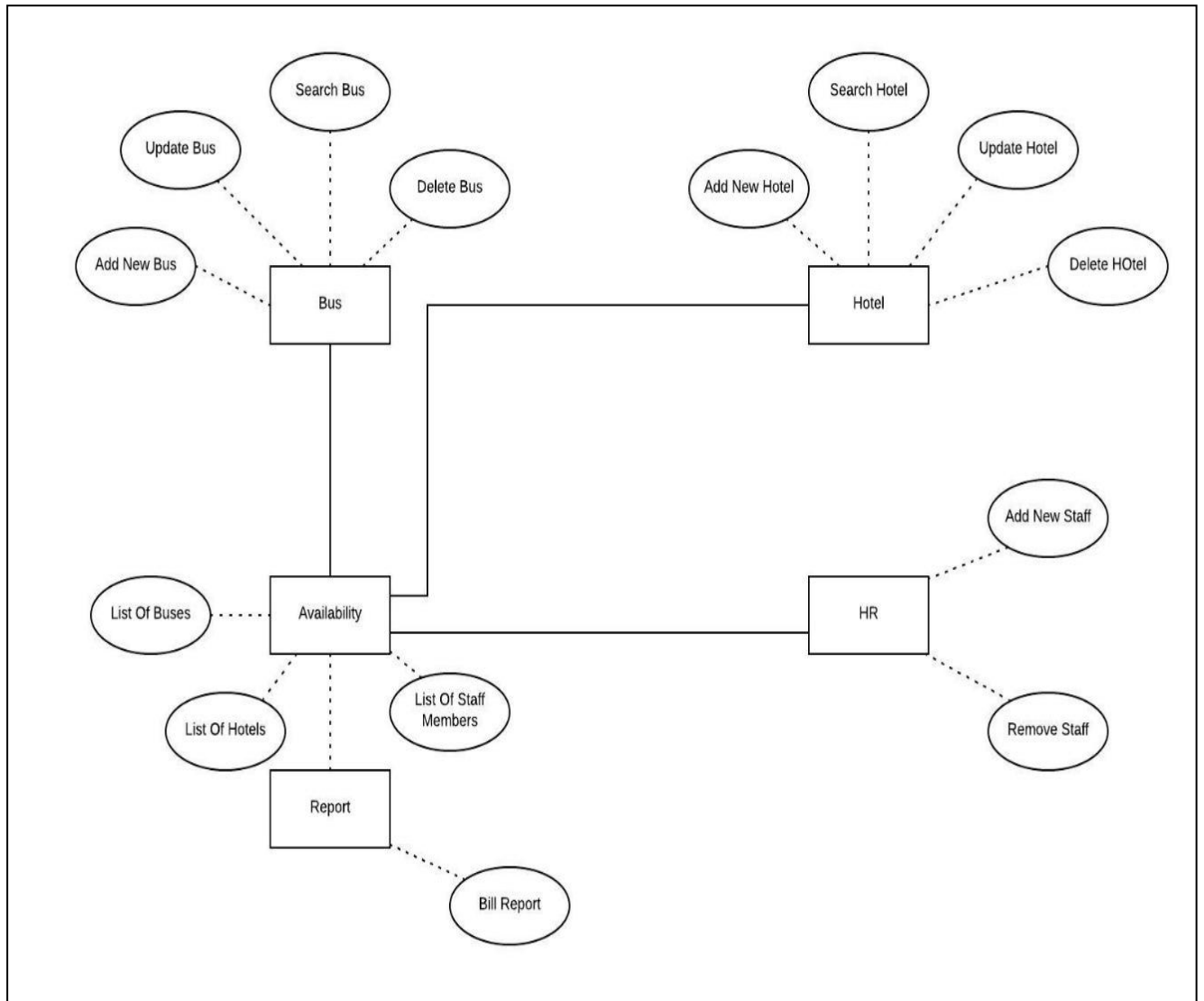


DFD Fragments

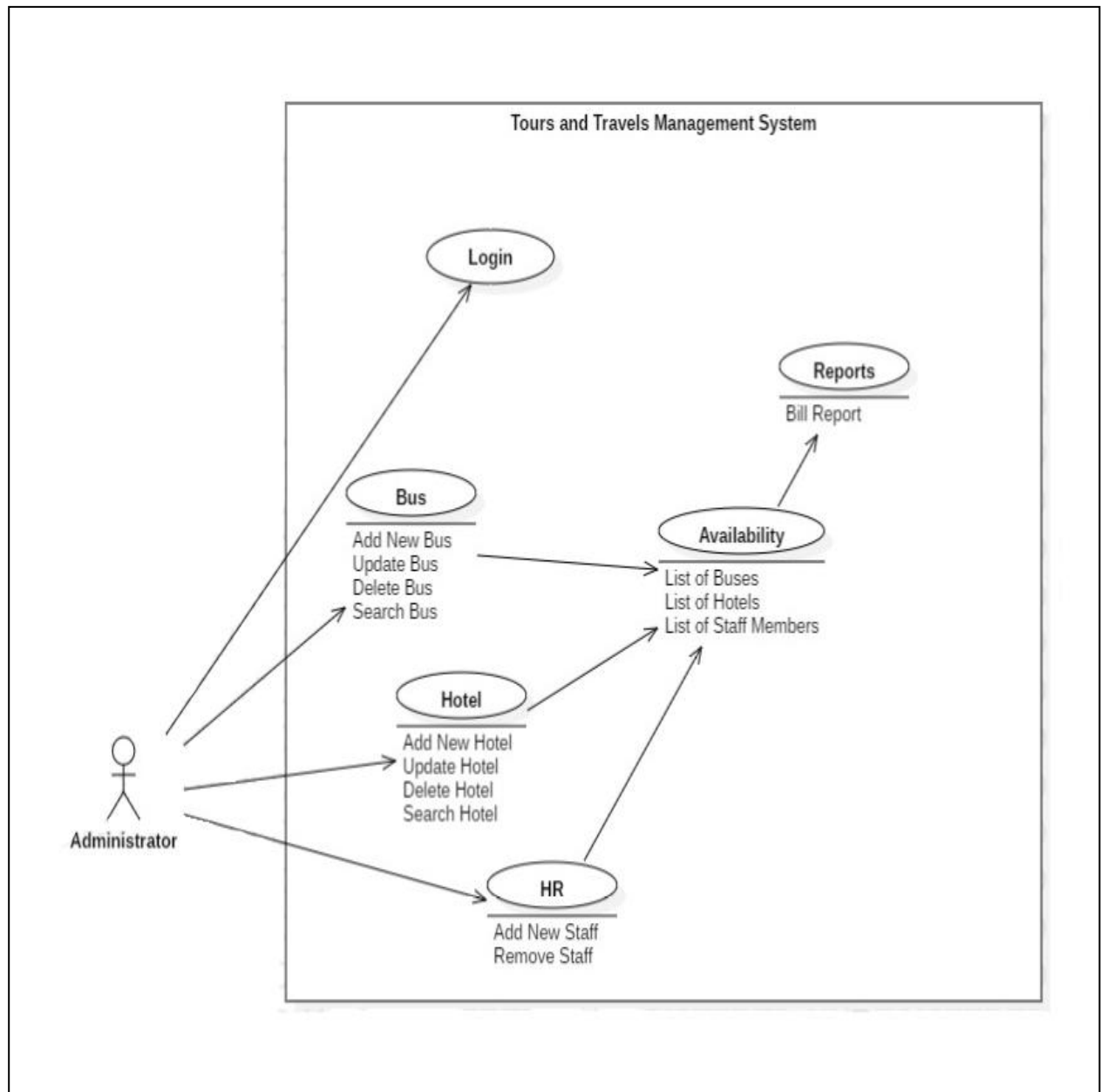
First level:



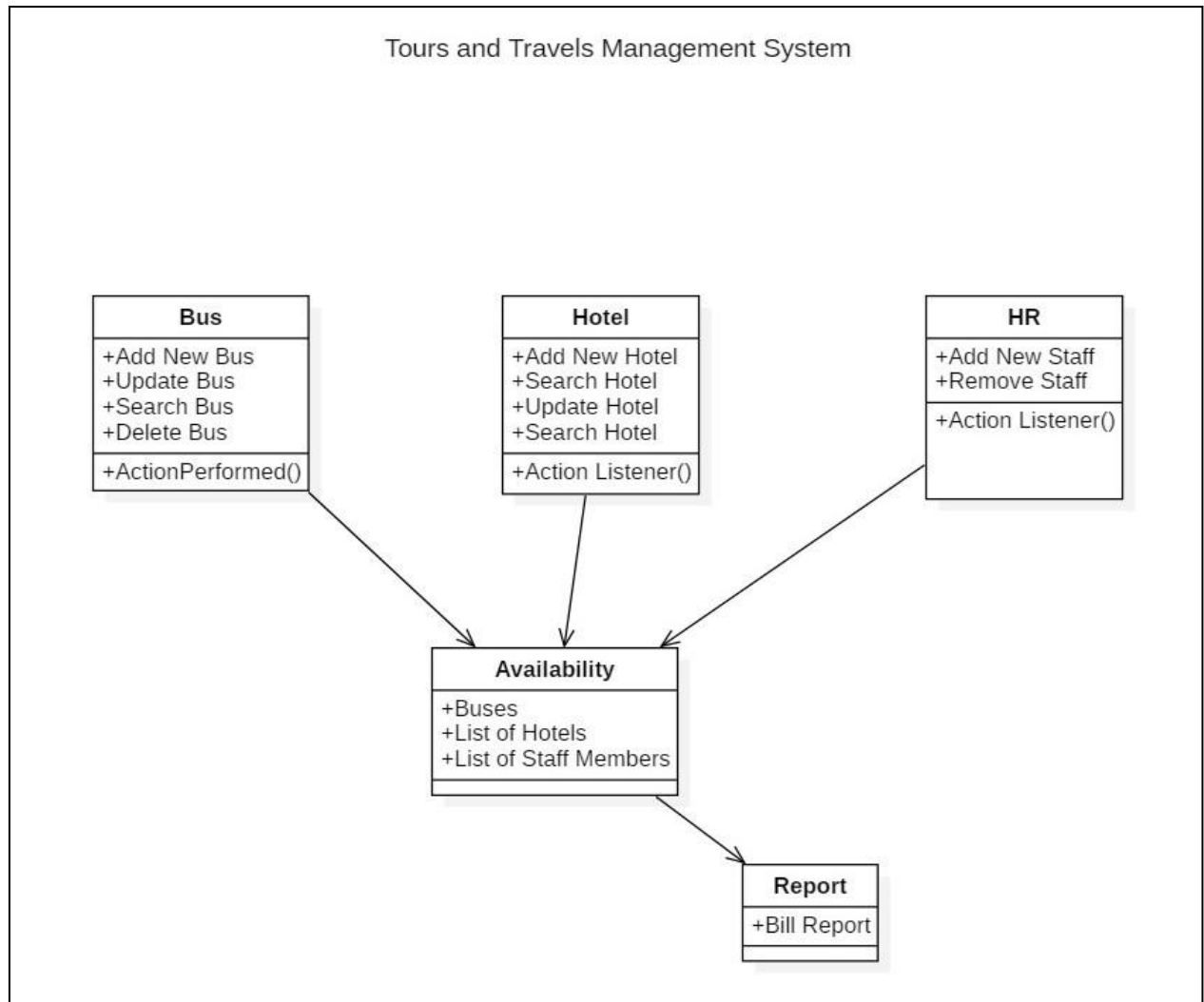
ER Diagram



Use-Case diagram



Class Diagram



EVENT TABLE

Event	Trigger	Source	Activity	Response	Destination
1.Bus 1.1 Add New Bus 1.2 Search Bus 1.3 Update Bus 1.4 Delete Bus	Add, Search, Update, Delete	Client	Manages the Buses	Bus Availability Details	Client
2.0 Hotel 2.1 Add New Hotel 2.2 Search Hotel 2.3 Update Hotel 2.4 Delete Hotel	Add, Search, Update, Delete	Client	Manages the Hotel Details	Hotel details	Client Customer
3.0 HR 3.1 Add New Staff 3.2 Remove Staff	Add, Remove	Client	Keeps records of Staff	Details of Staff	Administrator
4.0 Availability 4.1 List of Buses 4.2 List of Hotels 4.3 List of Staff Members	List				
4.0 Reports 4.1 Bill Report			Assembles all the reports	Transaction summary	Management

INPUT OUTPUT SCREEN DESIGN

Input Design

In an information system, input is the raw data that is processed to produce output. During the input design, the developers must consider the input devices such as PC, MICR, OMR, etc.

Therefore, the quality of system input determines the quality of system output. Well designed input forms and screens have following properties –

It should serve specific purpose effectively such as storing, recording, and retrieving the information.

It ensures proper completion with accuracy.

It should be easy to fill and straightforward.

It should focus on user's attention, consistency, and simplicity.

All these objectives are obtained using the knowledge of basic design principles regarding –

- **Objectives for Input Design**

The objectives of input design are –

To design data entry and input procedures

To reduce input volume

To design source documents for data capture or devise other data capture methods

To design input data records, data entry screens, user interface screens, etc.

To use validation checks and develop effective input controls.

Data Input Methods

It is important to design appropriate data input methods to prevent errors while entering data. These methods depend on whether the data is entered by customers in forms manually and later entered by data entry operators, or data is directly entered by users on the PCs.

Batch input method (Offline data input method)

Online data input method

Computer readable forms

Interactive data input

Input Integrity Controls

Input integrity controls include a number of methods to eliminate common input errors by end-users. They also include checks on the value of individual fields; both for format and the completeness of all inputs.

• Output Design

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

• Objectives of Output Design:

The objectives of input design are –

To develop output design that serves the intended purpose and eliminates the production of unwanted output.

To develop the output design that meets the end users requirements.

To deliver the appropriate quantity of output.

To form the output in appropriate format and direct it to the right person.

To make the output available on time for making good decisions.

Let us now go through various types of outputs –

- External Outputs

Manufacturers create and design external outputs for printers. External outputs enable the system to leave the trigger actions on the part of their recipients or confirm actions to their recipients.

Some of the external outputs are designed as turnaround outputs, which are implemented as a form and re-enter the system as an input.

- Internal outputs

Internal outputs are present inside the system, and used by end-users and managers. They support the management in decision making and reporting.

There are three types of reports produced by management information –

PROCESS INVOLVED

Introduction

Project management is one of the critical processes of any project. This is due to the fact that project management is the core process that connects all other project activities and processes together.

When it comes to the activities of project management, there are plenty. However, these plenty of project management activities can be categorized into five main processes.

Let's have a look at the five main project management processes in detail.

1 - Project Initiation

Project initiation is the starting point of any project. In this process, all the activities related to winning a project takes place. Usually, the main activity of this phase is the pre-sale.

During the pre-sale period, the service provider proves the eligibility and ability of completing the project to the client and eventually wins the business. Then, it is the detailed requirements gathering which comes next.

During the requirements gathering activity, all the client requirements are gathered and analysed for implementation. In this activity, negotiations may take place to change certain requirements or remove certain requirements altogether.

Usually, project initiation process ends with requirements sign-off.

2 - Project Planning

Project planning is one of the main project management processes. If the project management team gets this step wrong, there could be heavy negative consequences during the next phases of the project.

Therefore, the project management team will have to pay detailed attention to this process of the project.

In this process, the project plan is derived in order to address the project requirements such as, requirements scope, budget and timelines. Once the project plan is derived, then the project schedule is developed.

Depending on the budget and the schedule, the resources are then allocated to the project. This phase is the most important phase when it comes to project cost and effort.

3 - Project Execution

After all paperwork is done, in this phase, the project management executes the project in order to achieve project objectives.

When it comes to execution, each member of the team carries out their own assignments within the given deadline for each activity. The detailed project schedule will be used for tracking the project progress.

During the project execution, there are many reporting activities to be done. The senior management of the company will require daily or weekly status updates on the project progress.

In addition to that, the client may also want to track the progress of the project. During the project execution, it is a must to track the effort and cost of the

project in order to determine whether the project is progressing in the right direction or not.

In addition to reporting, there are multiple deliveries to be made during the project execution. Usually, project deliveries are not onetime deliveries made at the end of the project. Instead, the deliveries are scattered through out the project execution period and delivered upon agreed timelines.

4 - Control and Validation

During the project life cycle, the project activities should be thoroughly controlled and validated. The controlling can be mainly done by adhering to the initial protocols such as project plan, quality assurance test plan and communication plan for the project.

Sometimes, there can be instances that are not covered by such protocols. In such cases, the project manager should use adequate and necessary measurements in order to control such situations.

Validation is a supporting activity that runs from first day to the last day of a project. Each and every activity and delivery should have its own validation criteria in order to verify the successful outcome or the successful completion.

When it comes to project deliveries and requirements, a separate team called 'quality assurance team' will assist the project team for validation and verification functions.

5 - Closeout and Evaluation

Once all the project requirements are achieved, it is time to hand over the implemented system and closeout the project. If the project deliveries are in par with the acceptance criteria defined by the client, the project will be duly accepted and paid by the customer.

Once the project closeout takes place, it is time to evaluate the entire project. In this evaluation, the mistakes made by the project team will be identified and will take necessary steps to avoid them in the future projects.

During the project evaluation process, the service provider may notice that they haven't gained the expected margins for the project and may have exceeded the timelines planned at the beginning.

In such cases, the project is not a 100% success to the service provider. Therefore, such instances should be studied carefully and should take necessary actions to avoid in the future.

Conclusion

Project management is a responsible process. The project management process connects all other project activities together and creates the harmony in the project.

Therefore, the project management team should have a detailed understanding on all the project management processes and the tools that they can make use for each project management process.

SYSTEM TESTING AND CONCLUSION

1. Unit Testing: -

This is the testing process which we can do manually because in this testing program is a tested individually using dummy record to see whether that program produce satisfied output as the company and validation also.

2. Validation Testing: -

In this requirements established as part of software requirements analysis are validated against the software that has been constructed. Validation testing provides final Assurance that software meets all functional, behavioural and performance requirements. Validation can be define in many ways but a simple definition is that validation succeeds when software Function in a manner that can be reasonably by the customer.

1. Validation test criteria
2. Configuration review
3. Alpha and Beta testing (conducted by end user)

3. System Testing:-

System testing is actually a series different test whose primary purpose is to full exercise the computer base system .Where the software and other system elements are tested as whole .To test computer software, we spiral out along streamlines that broadens the scope of testing with each turn. The last higher-order testing step falls outside the boundary of software Engineering and in to the broader context of computer system engineering, Software once validated, must be combining with order system elements (hardware, databases). System testing verifies that all the elements Mesh properly and that overall system function/performance is achieved.

1. Recovery Testing
2. Security Testing
3. Stress Testing

4. White Box Testing:-

WHITE BOX TESTING (also known as Clear Box Testing, Open Box Testing, Glass Box Testing, Transparent Box Testing, Code-Based Testing or Structural Testing) is a software testing method in which the internal structure/design/implementation of the item being tested is known to the tester.

In my project I have performed white box testing ,I have checked that entire application is working properly. I have also performed white box testing in front-end netbeans as well as back-end mysql. In this I have checked each and every section and internal working of project.

5. Black Box Testing:-

BLACK BOX TESTING, also known as Behavioural Testing, is a software testing method in which the internal structure/design/implementation of the item being tested is not known to the tester. I have performed Black box testing for my project, in this I have checked that all the records that are inputs and the outputs from those inputs are correct or not. I have mainly focused on inputs and outputs of the application.

TEST REPORTS

Any report is an important and laconic form of information transfer from the executor to the customer. Let me remember you about software testing process. Here we have the following stages:

- Project creating
- Test Plan preparing Execute testing. Find Bugs Make Reports
- Test Case execution
- Finding bugs
- Making reports

As you can see reports, which can be prepared, have to contain the information about the activities from the preceding stages.

So, we can define test report as a document containing information about the performed actions (run test cases, detected bugs, spent time etc.) and the results of this performance (failed/passed test cases, the number of bugs and crashes etc.)

TEST CASES:

Sr. No	Name	Input	Expected Output	Actual output	Remark
1	Login Page	Valid credentials	Navigate to users Purchase Page	Navigate to users Purchase Page	Pass
2	Login Page	Invalid credentials	Popup showing Invalid password for user	Popup showing Invalid password for user	Pass
3	Login Page	No value in username or password	Popup showing Please Enter the username or password	Popup showing Please Enter the username or password	Pass
4	Bus Page	No value in Text Field	Pop Showing Value is Required	Value is Required	Pass
5	Hotel Page	No hotel in list	Pop Showing Required hotel is Unavailable	Pop Showing Required hotel is Unavailable	Pass
6	Staff Page	No value in Text Field	Value is Required	Value is Required	Pass
7	Report Page	No value in Text Field	Value is Required	Value is Required	Pass
8	About Page	No value in Text Field	Value is Required	Value is Required	Pass

FUTURE ENHANCEMENTS

- There is always a room for improvement in any software package, however good and efficient it may be.
- But the improvement thing is that the system should be flexible enough for further modifications.
- Considering this important factor, the system is designed in such a way that provisions can be given for further enhancement without affecting the system presently developed.
- In future, will try to improve our tour destinations as per the customers expectations.
- Also we will try to provide maximum best hotel packages to the customers.

SOURCE CODE:

1.Login Page

```
import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;

class InvalidException extends Exception
{
}

class Login extends JFrame implements ActionListener
{
    JFrame jf;
    JButton b1,b2,b3;
    JLabel l1,l2,l3,l4,l5,l6;
    JTextField t1,t2;
    JPasswordField p1;
    ResultSet rs,rs1;
    PreparedStatement ps,ps1;
    Connection con;
    Font f;
    int cnt=0,cnt1=0;
    Login()
    {
        //setVisible(true);
        jf=new JFrame();
```

```
f = new Font("Times New Roman",Font.BOLD,20);
jf.setLayout(new BorderLayout());
jf.setTitle("Login");
jf.setLocation(20,20);
jf.setSize(800,600);
jf.setResizable(true);
jf.setLayout(new FlowLayout());

jf.setContentPane(new JLabel(new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\log.jpg")));

l5 = new JLabel("ADMINISTRATOR");l5.setFont(new Font("Times New
Roman",Font.BOLD,30));

l5.setBounds(300,100,300,40);
jf.add(l5);

l3 = new JLabel(new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\log.jpg"));
l3.setBounds(150,250,50,25);
jf.add(l3);

l1 = new JLabel("User Name : "); l1.setFont(f);
l1.setBounds(200,250,200,25);
jf.add(l1);

t1 = new JTextField(20);
t1.setBounds(350,250,200,25);
t1.setToolTipText("Enter Username");
jf.add(t1);

l4 = new JLabel(new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\log.jpg"));
l4.setBounds(150,300,50,25);
jf.add(l4);

l2 = new JLabel("Password : ");
l2.setFont(f);
l2.setBounds(200,300,200,25);
```

```
jf.add(l2);

p1 = new JPasswordField(20);
p1.setBounds(350,300,200,25);
p1.setToolTipText("Enter Password");
jf.add(p1);

b1 = new JButton("Login",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\log.jpg"));
b1.setBounds(200,400,100,35);
jf.add(b1);

b1.addActionListener(this);

b2 = new JButton("Clear",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\log.jpg"));
b2.setBounds(320,400,100,35);
jf.add(b2);

b2.addActionListener(this);

b3 = new JButton("Exit",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\log.jpg"));
b3.setBounds(440,400,100,35);
jf.add(b3);

b3.addActionListener(this);

//background
jf.setVisible(true);
}

public void actionPerformed(ActionEvent ae)
{
if(ae.getSource()==b1)
{
try
{
String uname=t1.getText();
String pass=new String(p1.getPassword());
```

```
Class.forName("com.mysql.jdbc.Driver");
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/tours","root","root123"
);
System.out.println("Connected to database.");
ps1=con.prepareStatement("select * from login where uname=? and pass=?");
ps1.setString(1,uname);
ps1.setString(2,pass);
rs1=ps1.executeQuery();
if(rs1.next())
{
JOptionPane.showMessageDialog(null," Welcome !!!....."+uname+" ...!!!
","WELCOME",JOptionPane.INFORMATION_MESSAGE);
jf.setVisible(false);
new MainMenu();
}
else
{
throw new InvalidException();
}
}
catch(Exception e1)
{
cnt++;
JOptionPane.showMessageDialog(null," Sorry !!! You are not valid user
...!!!","WARNING",JOptionPane.ERROR_MESSAGE);
t1.setText("");
p1.setText("");
if(cnt==3)
{
JOptionPane.showMessageDialog(null,"Sorry !!! Your 3 attempts are over
...!!!","WARNING",JOptionPane.ERROR_MESSAGE);
```

```
System.exit(0);
}
}
}

else if(ae.getSource()==b2)
{
t1.setText("");
p1.setText("");
}
else if(ae.getSource()==b3)
{
System.exit(0);
}
}

public static void main(String args[])
{
new Login();
}
}
```


2.Main menu Page

```
import java.awt.*;
import java.sql.*;
import java.awt.event.*;
import javax.swing.*;

public class MainMenu extends JFrame implements ActionListener
{
    JFrame jf;
    JMenuBar mbar;
    JMenuItem add;
    JMenu m1,m2,m3,m4,m5,m6,m7;
    JMenuItem
    m1_1,m1_2,m1_3,m2_1,m2_2,m2_3,m2_4,m3_1,m3_2,m3_3,m3_4,m4_1,m4_2,m5_1,m6_
    1,m7_1;
    JLabel l1,LogoColl;
    GridBagLayout gbl;
    public MainMenu()
    {
        jf=new JFrame();
        jf.setLayout(new BorderLayout());
        jf.setTitle("main menu");
        jf.setLocation(20,20);
        jf.setSize(800,556);
        jf.setResizable(true);
        jf.setLayout(new FlowLayout());
        jf.setContentPane(new JLabel(new
        ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\mainmen
        u.jpg")));
        gbl=new GridBagLayout();
```

```
jf.setLayout(gbl);

l1=new JLabel("WELCOME TO TOURS AND TRAVELS MANAGEMENT SYSTEM");

l1.setFont(new Font("Times New Roman",Font.BOLD,26));

l1.setForeground(Color.white);

jf.add(l1);

mbar = new JMenuBar();

jf.setJMenuBar(mbar);

m1=new JMenu("Availability");

mbar.add(m1);

m1_1 = new JMenuItem("List of Buses",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\addnew.p
ng"));

m1.add(m1_1);

m1_2 = new JMenuItem("List of Hotels",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\search.pn
g"));

m1.add(m1_2);

m1_3 = new JMenuItem("List of Staff Members",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\update.pn
g"));

m1.add(m1_3);

m2=new JMenu("Bus");

JMenu add1 = mbar.add(m2);

m2_1 = new JMenuItem("Add new Bus",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\report.png
"));

m2.add(m2_1);

m2_2 = new JMenuItem("Update Bus",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\report.png
"));

m2.add(m2_2);

m2_3 = new JMenuItem("Delete Bus",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\report.png
"));

m2.add(m2_3);
```

```
m2_4 = new JMenuItem("Search Bus",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\report.png
"));
m2.add(m2_4);

m3=new JMenu("Hotel");
JMenu add = mbar.add(m3);

m3_1 = new JMenuItem("Add new Hotel",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\report.png
"));
m3.add(m3_1);

m3_2 = new JMenuItem("Update Hotel",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\report.png
"));
m3.add(m3_2);

m3_3 = new JMenuItem("Delete Hotel",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\report.png
"));
m3.add(m3_3);

m3_4 = new JMenuItem("Search Hotel",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\report.png
"));
m3.add(m3_4);

m4=new JMenu("HR");
mbar.add(m4);

m4_1 = new JMenuItem("Add New Staff",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\report.png
"));
m4.add(m4_1);
mbar.add(m4);

m4_2 = new JMenuItem("Remove Staff",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\report.png
"));
m4.add(m4_2);

m5=new JMenu("Report");
mbar.add(m5);
```

```
m5_1 = new JMenuItem("Bill Report",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\help.png"
));
m5.add(m5_1);

m6=new JMenu("About");
mbar.add(m6);

m6_1 = new JMenuItem("About",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\exit.png")
);
m6.add(m6_1);

m7=new JMenu("Exit");
mbar.add(m7);

m7_1 = new JMenuItem("Exit",new
ImageIcon("C:\\Users\\DELL\\Documents\\NetBeansProjects\\travels\\src\\travels\\exit.png")
);
m7.add(m7_1);

m1_1.addActionListener(this);
m1_2.addActionListener(this);
m1_3.addActionListener(this);
m2_1.addActionListener(this);
m2_2.addActionListener(this);
m2_3.addActionListener(this);
m2_4.addActionListener(this);
m3_1.addActionListener(this);
m3_2.addActionListener(this);
m3_3.addActionListener(this);
m3_4.addActionListener(this);
m4_1.addActionListener(this);
m4_2.addActionListener(this);
m5_1.addActionListener(this);
m6_1.addActionListener(this);
m7_1.addActionListener(this);
```

```
jf.setVisible(true);
}
public void actionPerformed(ActionEvent ae)
{
    if(ae.getSource()==m1_1)
    {
        ListOfBuses addNewBooking = new ListOfBuses();
        addNewBooking.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m1_2)
    {
        ListOfHotels cancelBooking = new ListOfHotels();
        cancelBooking.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m1_3)
    {
        ListOfStaff updateBooking = new ListOfStaff();
        updateBooking.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m2_1)
    {
        AddNewBus addnewbus = new AddNewBus();
        addnewbus.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m2_2)
    {
```

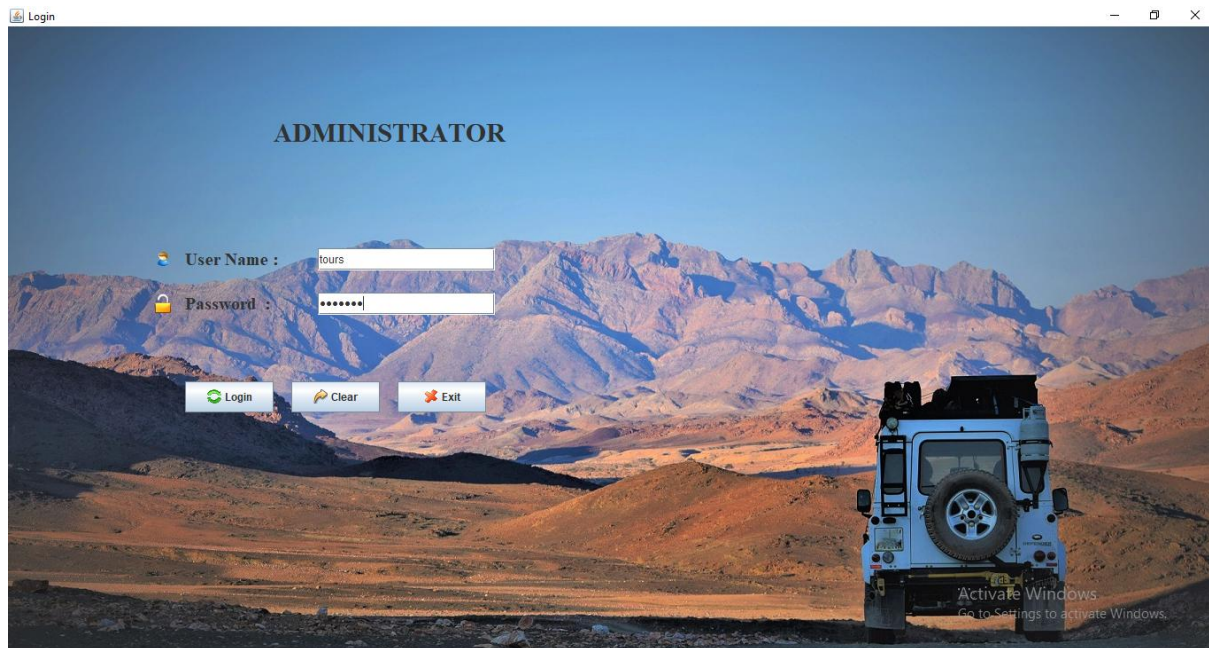
```
        UpdateBus updatebus = new UpdateBus();
        updatebus.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m2_3)
    {
        DeleteBus deletebus = new DeleteBus();
        deletebus.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m2_4)
    {
        SearchBus searchbus = new SearchBus();
        searchbus.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m3_1)
    {
        AddNewHotel addnewhotel = new AddNewHotel();
        addnewhotel.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m3_2)
    {
        UpdateHotel updatehotel = new UpdateHotel();
        updatehotel.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m3_3)
    {
```

```
        DeleteHotel deletehotel = new DeleteHotel();
        deletehotel.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m3_4)
    {
        SearchHotel searchhotel = new SearchHotel();
        searchhotel.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m4_1)
    {
        AddNewStaff addStaff = new AddNewStaff();
        addStaff.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m4_2)
    {
        RemoveStaff removeStaff = new RemoveStaff();
        removeStaff.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m5_1)
    {
        BillReport customerDetails = new BillReport();
        customerDetails.setVisible(true);
        this.dispose();
    }
    else if(ae.getSource()==m6_1)
    {
```

```
        About about = new About();  
  
    }  
    else if(ae.getSource()==m7_1)  
    {  
        System.exit(0);  
    }  
    }  
    public static void main(String args[])  
    {  
        MainMenu mainMenu = new MainMenu();  
    }  
    }
```


SCREENSHOTS OF THE JAVA PAGES

1. Administrator login:



2. Main Menu:



3. List of buses:

List of Buses					
B_ID	B_NAME	B_From	B_Destination	B_Seats	B_Price
a4	shruti	mumbai	delhi	45	900.0
b1	shivneri	mumbai	nashik	50	240.0
d3	abc	mumbai	goa	50	600.0
e2	praveer	mumbai	bangalore	44	1200.0
f3	saibaba	pune	chennai	60	800.0
g5	shivshakti	mumbai	nagpur	40	500.0
h7	shivshahi	mumbai	pune	55	450.0
j8	anuj	mumbai	matheran	60	300.0
o9	aditya	mumbai	rajasthan	40	1000.0
s7	tejas	mumbai	mahabaleshwar	50	550.0

4. List of hotels:

List of Hotels					
H_Name	H_Ratings	H_Location	H_Type	H_Charges	H_Rooms
kumar	3	mumbai	Classic	3700	140
oberoi	4	pune	Royal Suite	14500	180
pathan	4	nagpur	Premium	4500	175
prithwi	3	chennai	Standard	1600	100
raj	2	goa	Premium	1000	80
rudra	4	kolkata	Classic	4000	190
samudra	4	rajasthan	Classic	4555	300
shaurya	4	matheran	Premium	4000	155
shilpa	5	bangalore	Royal Suite	10000	400
taj	5	mahabaleshwar	Superior	15000	200
trident	4	delhi	Premium	1500	250

5. List of Staff members:

List of Staff							
ID	NAME	GENDER	PHONENO	AGE	DOB	EMAIL_ID	ADDRESS
1	tejas	MALE	8097346150	32	25/2/2013	tej@gmail.com	Thane
2	rohit	MALE	8097346150	30	25/2/2012	tej@gmail.com	Thane
3	pranil	MALE	8369275205	32	24/2/2019	pra@gmail.com	Rabodi
4	pranali	MALE	9586755753	37	15/4/2017	t@gmail.com	thane
5	tom	MALE	9874561232	39	14/4/2018	g2D@d.com	mulund
6	jerry	FEMALE	8497451554	39	24/2/2014	g2D@d.com	kalyan
7	tashvi	FEMALE	8487595784	29	25/5/2013	cd@gmail.com	mulund

6. Add new bus:

Add New Bus

New Bus details

Bus ID*

Bus name*

From

Destination

No. of seats

Ticket price

B_ID	B_NAME	B_From	B_Destination	B_Seats	B_Price
a4	shruti	mumbai	delhi	45	900.0
b1	shivneri	mumbai	nashik	50	240.0
d3	abc	mumbai	goa	50	600.0
e2	praveer	mumbai	bangalore	44	1200.0
f3	saibaba	pune	chennai	60	800.0
g5	shivshakti	mumbai	nagpur	40	500.0
h7	shivshahi	mumbai	pune	55	450.0
j8	anuj	mumbai	matheran	60	300.0
o9	aditya	mumbai	rajasthan	40	1000.0
s7	tejas	mumbai	mahabaleshwar	50	550.0

7. Update bus:

Update Bus

Update Bus

Bus ID*

Bus name*

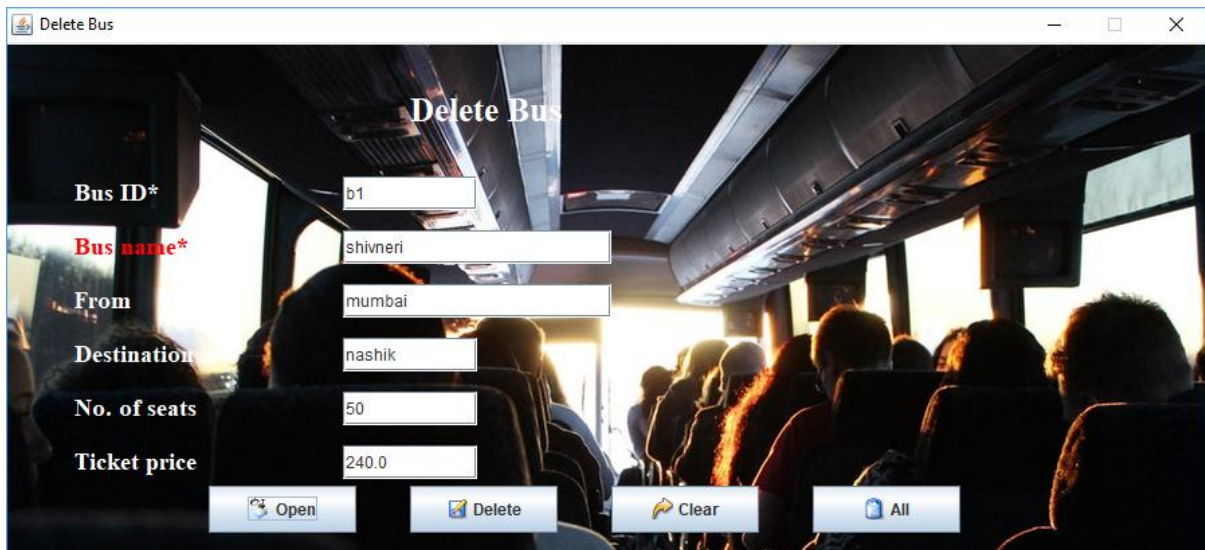
From

Destination

No. of seats

Ticket price

8. Delete bus:



Delete Bus

Bus ID*

Bus name*

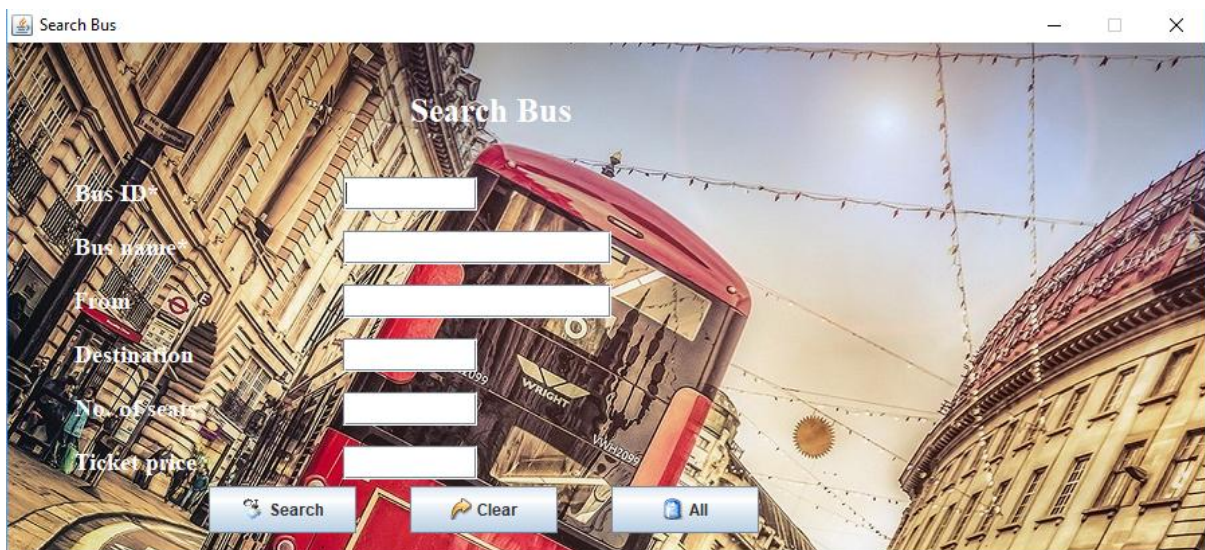
From

Destination

No. of seats

Ticket price

9. Search bus:



Search Bus

Bus ID*

Bus name*

From

Destination

No. of seats

Ticket price

10. Add new hotel:

Add New Hotel

Hotel name*

Ratings*

Location

Hotel type*

Per Day Charges

Availability of rooms

H_Name	H_Ratings	H_Location	H_Type	H_Charges	H_Rooms
kumar	3	mumbai	Classic	3700	140
oberoi	4	pune	Royal Suite	14500	180
pathan	4	nagpur	Premium	4500	175
prithwi	3	chennai	Standard	1600	100
raj	2	goa	Premium	1000	80
rudra	4	kolkata	Classic	4000	190
samudra	4	rajasthan	Classic	4555	300
shaurya	4	matheran	Premium	4000	155
shilpa	5	banglore	Royal Suite	10000	400
taj	5	mahabaleshwar	Superior	15000	200
trident	4	delhi	Premium	1500	250

11. Update hotel:

Update Hotel

Hotel name*

Ratings*

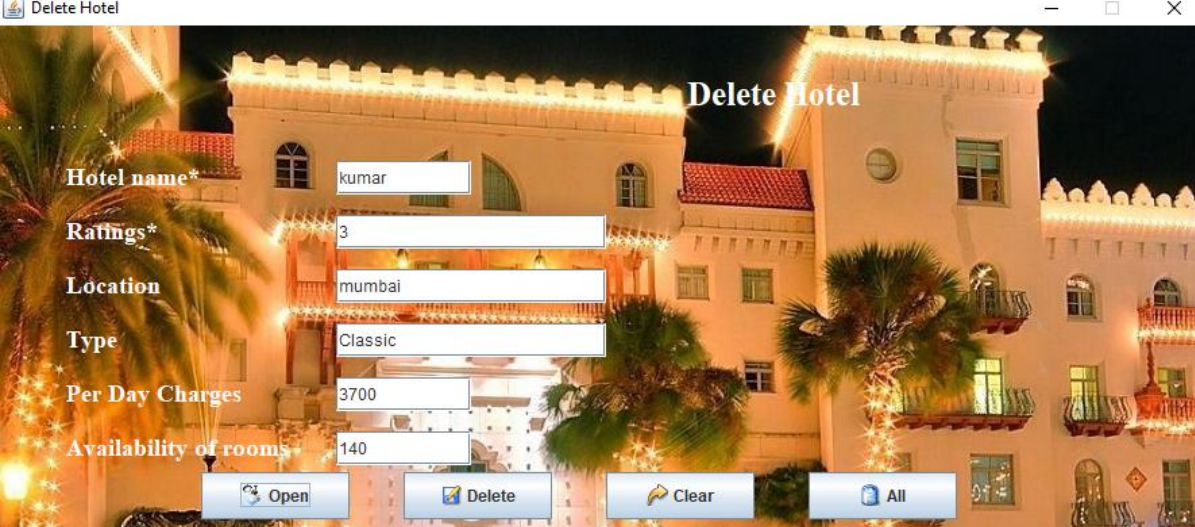
Location

Type

Per Day Charges

Availability of rooms

12. Delete hotel:



Delete Hotel

Hotel name* kumar

Ratings* 3

Location mumbai

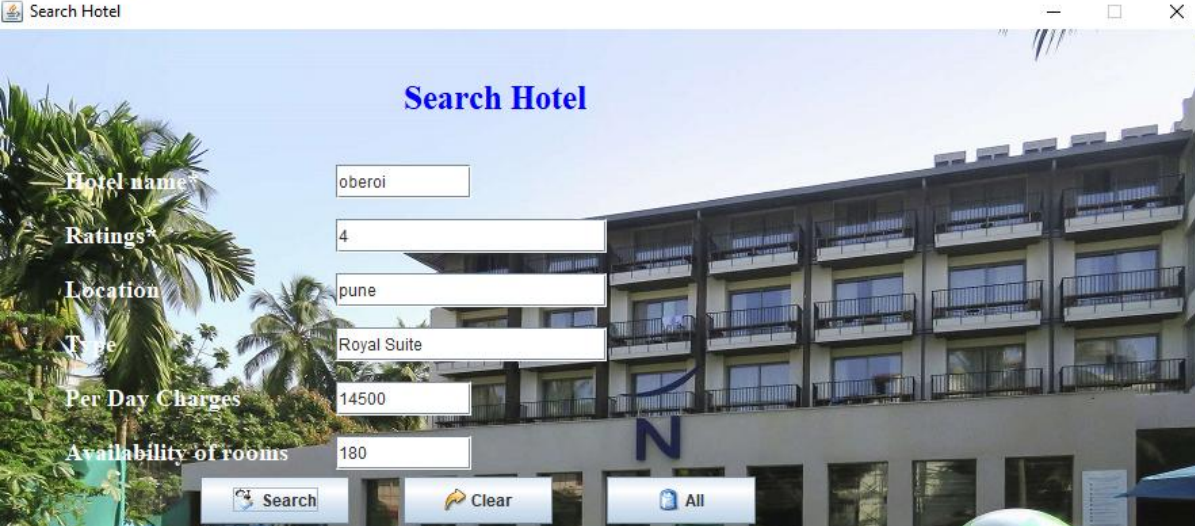
Type Classic

Per Day Charges 3700

Availability of rooms 140

Open Delete Clear All

13. Search hotel:



Search Hotel

Hotel name* oberoi

Ratings* 4

Location pune

Type Royal Suite

Per Day Charges 14500

Availability of rooms 180

Search Clear All

14. Add new Staff:

Add New Staff

ID	<input type="text" value="2"/>	Email ID	<input type="text" value="tejas@gmail.com"/>
Name	<input type="text" value="tejas"/>	Address	<input type="text" value="thane"/>
Gender	<input type="text" value="MALE"/>	Post	<input type="text" value="trainer"/>
Phone Number	<input type="text" value="9773709511"/>		
Age	<input type="text" value="25"/>		
Date Of Birth	<input type="text" value="27/2/2019"/>		

15. Remove staff:

Remove Staff

ID	<input type="text" value="2"/>	Email ID	<input type="text" value="tej@gmail.com"/>
Name	<input type="text" value="rohit"/>	Address	<input type="text" value="Thane"/>
Gender	<input type="text" value="MALE"/>	Post	<input type="text" value="manager"/>
Phone Number	<input type="text" value="8097346150"/>		
Age	<input type="text" value="30"/>		
Date Of Birth	<input type="text" value="25/2/2012"/>		

16. Bill Report:

Bill Report

Bill no	<input type="text" value="22"/>	Hotel Rooms	<input type="text" value="2"/>
Customer Name*	<input type="text" value="tejas"/>	No. of seats	<input type="text" value="4"/>
Bus From	<input type="text" value="mumbai"/>	Hotel Price	<input type="text" value="14500"/>
Bus Destination	<input type="text" value="pune"/>	Bus Price	<input type="text" value="180"/>
Hotel Location	<input type="text" value="pune"/>	Tax	<input type="text" value="2972.0"/>
Hotel name	<input type="text" value="oberoi"/>	Total Price	<input type="text" value="32692.0"/>

17. About:

Tours and Travels Database Manangement System

This System developed by,

Mr. Tejas Argade and Mr. Rohit Narwade

Under the guidance of Mr. Tejas Jadhav

In this system we can add details of the whole database as per Client's requirements.

We can also update,delete & search the existing details.

It is helpfull for keeping details of client & also booking the tours.

Validation

1.

The screenshot shows a web application window titled "Add New Bus". Inside, there's a form titled "New Bus details" with a background image of a red bus. The form fields are: Bus ID* (b1), Bus name* (shivba), From (mumbai), Destination (kolhapur), No. of seats (50), and Ticket price (jisj). Below the fields are "Save", "Clear", and "All" buttons. A "Warning!!!" dialog box is open, displaying a yellow warning icon and the message "Price should be in Integer!!!". At the bottom of the form, there's a table with headers: B_ID, B_NAME, B_From, B_Dest, B_Seats, and B_Price.

B_ID	B_NAME	B_From	B_Dest	B_Seats	B_Price
------	--------	--------	--------	---------	---------

2.

The screenshot shows the same "Add New Bus" form. The fields are: Bus ID* (b3), Bus name* (tejas), From (mumbai), Destination (goa), No. of seats (hjhjh), and Ticket price (20). The "Warning!!!" dialog box now displays the message "Seats should be in Integer!!!". The table at the bottom remains the same.

B_ID	B_NAME	B_From	B_Dest	B_Seats	B_Price
------	--------	--------	--------	---------	---------

3.

The screenshot shows a web application window titled "Add New Staff". The form contains the following fields and values:

ID	Name	Gender	Phone Number	Age	Date Of Birth	Email ID	Address	Post
8	praveer	MALE	jkjkj	29	27/5/1993	p@gmail.com	thane	manager

Buttons: Save, Clear, All.

A warning dialog box is displayed with the message: "Warning!!! Invalid phone no!". The dialog has an "OK" button.

The form is displayed over a background image of people's hands raised in a high-five gesture.

4.

The screenshot shows the same "Add New Staff" form, but with the following values:

ID	Name	Gender	Phone Number	Age	Date Of Birth	Email ID	Address	Post
8	praveer	MALE	1234567890	20	27/5/1993	jjjjjjjj	thane	manager

Buttons: Save, Clear, All.

A warning dialog box is displayed with the message: "Warning!!! Invalid email id!". The dialog has an "OK" button.

The form is displayed over a background image of people's hands raised in a high-five gesture.

5.

Add New Staff

ID **Email ID**

Name **Address**

Gender **Post**

Phone Number

Age

Date Of Birth

Warning!!!
Invalid Date use format [1/1/2019]

ID	NAME	GENDER	PHONENO	AGE	POST
----	------	--------	---------	-----	------

Screenshots of database tables

1.

The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left displays the 'tours' database. The 'Table: bill' is selected, showing its columns: bno, cname, bfrom, bdest, hloc, hname, rooms, seats, hprice, bprice, btax, and total. The 'Query Editor' contains the query: `SELECT * FROM tours.bill;`. The 'Result Grid' displays the following data:

bno	cname	bfrom	bdest	hloc	hname	rooms	seats	hprice	bprice	btax	total
1	TEJAS	mumbai	pune	pune	dc	7	5	100	5	73	798
4	rf	mumbai	pune	pune	dc	3	3	100	100	60	660
21	qd	mumbai	pune	pune	dc	3	3	100	95	59	644
22	ads	mumbai	pune	pune	dc	2	2	100	97	39	433

The 'Action Output' pane shows the execution of the query, returning 4 rows in 0.078 seconds.

2.

The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left displays the 'tours' database. The 'Table: bus' is selected, showing its columns: bid, bname, start, destination, bseats, and bprice. The 'Query Editor' contains the query: `SELECT * FROM tours.bus;`. The 'Result Grid' displays the following data:

bid	bname	start	destination	bseats	bprice
a4	shruti	mumbai	delhi	45	900
b1	shivneri	mumbai	nashik	50	240
d3	abc	mumbai	goa	50	600
e2	praveer	mumbai	bangalore	44	1200
f3	saibaba	pune	chennai	60	800
g5	shivshakti	mumbai	nagpur	40	500
h7	shivshakti	mumbai	pune	55	450
j8	anuj	mumbai	matheran	60	300

The 'Action Output' pane shows the execution of the query, returning 10 rows in 0.047 seconds.

3.

The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left displays the 'sakila' database structure, including tables like 'bill', 'bus', 'hotel', 'login', 'staff', and 'views'. The 'Table: hotel' structure is detailed in the 'Columns' pane, showing fields like 'hname', 'hrating', 'hlloc', 'htype', 'hcharge', and 'hroom'. The 'Query Editor' contains the SQL query: `SELECT * FROM tours.hotel;`. The 'Result Grid' displays the query results, showing columns: 'hname', 'hrating', 'hlloc', 'htype', 'hcharge', and 'hroom'. The 'Output' pane shows the execution log with three rows of results.

hname	hrating	hlloc	htype	hcharge	hroom
kumar	3	mumbai	Classic	3700	140
oberoi	4	pune	Royal Suite	14500	180
pathan	4	nagpur	Premium	4500	175
prithvi	3	chennai	Standard	1600	100
raj	2	goa	Premium	1000	80
rudra	4	kolkata	Classic	4000	190
samudra	4	rajasthan	Classic	4555	300
shaurya	4	matheran	Premium	4000	155

Output:

#	Time	Action	Message
1	20:59:27	SELECT * FROM tours.bill LIMIT 0, 1000	4 row(s) returned
2	20:59:40	SELECT * FROM tours.bus LIMIT 0, 1000	10 row(s) returned
3	21:00:19	SELECT * FROM tours.hotel LIMIT 0, 1000	11 row(s) returned

4.

The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left displays the 'sakila' database structure, including tables like 'bill', 'bus', 'hotel', 'login', 'staff', and 'views'. The 'Table: login' structure is detailed in the 'Columns' pane, showing fields like 'uname', 'pass', 'rolname', and 'travels'. The 'Query Editor' contains the SQL query: `SELECT * FROM tours.login;`. The 'Result Grid' displays the query results, showing columns: 'uname', 'pass', 'rolname', and 'travels'. The 'Output' pane shows the execution log with four rows of results.

uname	pass	rolname	travels
tejas	rohit	travels	travels
tejas	rohit	travels	travels
tejas	rohit	travels	travels
tejas	rohit	travels	travels

Output:

#	Time	Action	Message
1	20:59:27	SELECT * FROM tours.bill LIMIT 0, 1000	4 row(s) returned
2	20:59:40	SELECT * FROM tours.bus LIMIT 0, 1000	10 row(s) returned
3	21:00:19	SELECT * FROM tours.hotel LIMIT 0, 1000	11 row(s) returned
4	21:00:24	SELECT * FROM tours.login LIMIT 0, 1000	2 row(s) returned

5.

The screenshot displays the MySQL Workbench interface. The left sidebar shows the 'SCHEMAS' tree with 'sakila' selected. The 'Tables' list under 'sakila' includes 'bill', 'bus', 'hotel', 'login', 'staff', and 'views'. The 'Table: hotel' is selected, showing its columns: 'hname' (varchar(45) PK), 'hrating' (varchar(45)), 'hloc' (varchar(45)), 'htype' (varchar(45)), 'hcharge' (varchar(45)), and 'hroom' (varchar(45)).

The main query editor shows the query: `SELECT * FROM tours.staff;`. The 'Result Grid' displays the following data:

id	name	gender	phone no	age	dob	email id	address	post
1	tejas	MALE	8097346150	32	25/2/2013	tej@gmail.com	Thane	manager
2	rohit	MALE	8097346150	30	25/2/2012	tej@gmail.com	Thane	manager
3	pranil	MALE	8369275205	32	24/2/2019	pra@gmail.com	Rabodi	employee
4	pranali	MALE	9586755753	37	15/4/2017	t@gmail.com	thane	employee
5	tom	MALE	9674561232	39	14/4/2018	g2d@gmail.com	mulund	employee
6	jerry	FEMALE	8497451554	39	24/2/2014	g2d@gmail.com	kalyan	peon
7	tashvi	FEMALE	8487595784	29	25/5/2013	cd@gmail.com	mulund	trainer

The 'Output' tab shows the execution log:

#	Time	Action	Message
1	20:59:27	SELECT * FROM tours.bill LIMIT 0, 1000	4 row(s) returned
2	20:59:40	SELECT * FROM tours.bus LIMIT 0, 1000	10 row(s) returned
3	21:00:19	SELECT * FROM tours.hotel LIMIT 0, 1000	11 row(s) returned
4	21:00:24	SELECT * FROM tours.login LIMIT 0, 1000	2 row(s) returned
5	21:00:31	SELECT * FROM tours.staff LIMIT 0, 1000	7 row(s) returned

A 'Screenshot Added' notification is visible in the bottom right corner, stating: 'A screenshot was added to your Dropbox. Dropbox'.

Security Aspects

Java's security model is one of the language's key architectural features that makes it an appropriate technology for networked environments. Security is important because networks provide a potential avenue of attack to any computer hooked to them. This concern becomes especially strong in an environment in which software is downloaded across the network and executed locally, as is done with Java applets, for example. Because the class files for an applet are automatically downloaded when a user goes to the containing Web page in a browser, it is likely that a user will encounter applets from untrusted sources. Without any security, this would be a convenient way to spread viruses. Thus, Java's security mechanisms help make Java suitable for networks because they establish a needed trust in the safety of network-mobile code.

Java's security model is focused on protecting users from hostile programs downloaded from untrusted sources across a network. To accomplish this goal, Java provides a customizable "sandbox" in which Java programs run. A Java program must play only inside its sandbox. It can do anything within the boundaries of its sandbox, but it can't take any action outside those boundaries. The sandbox for untrusted Java applets, for example, prohibits many activities, including:

- Reading or writing to the local disk
- Making a network connection to any host, except the host from which the applet came
- Creating a new process
- Loading a new dynamic library and directly calling a native method

By making it impossible for downloaded code to perform certain actions, Java's security model protects the user from the threat of hostile code.

Because of the safety features built into the Java virtual machine, running programs can access memory only in safe, structured ways. This helps make Java programs robust, but also makes their execution more secure. Why? There are two reasons.

First, a program that corrupts memory, crashes, and possibly causes other programs to crash represents one kind of security breach. If you are running a mission-critical server process, it is critical that the process doesn't crash. This level of robustness is also important in embedded systems such as a cell phone, which people don't usually expect to have to reboot.

The second reason unrestrained memory access would be a security risk is because a wiley cracker potentially could use the memory to subvert the security system. If, for example, a cracker could learn where in memory a class loader is stored, it could assign a pointer to that memory and manipulate the class loader's data. By enforcing structured access to memory, the Java virtual machine yields programs that are robust -- but also frustrates crackers who dream of harnessing the internal memory of the Java virtual machine for their own devious plots.

REFERENCES AND BIBLIOGRAPHY

Information and data related to the project has been taken from the sources below, special thanks to the editors for making the task easier: -

Reference books

- 1.Beginning on Java - Herbert Schildt.
- 2.SQL- The Complete Reference -James R. Groff & Paul N. Weinberg

- **Websites**

1. www.youtube.com
2. www.google.com
3. <http://www.java.com/beginners/>
4. <https://www.tutorialspoint.com/java>
5. www.w3schools.com