A Project Report on

ONLINE BUS TICKET RESERVATION SYSTEM

Submitted in partial fulfilment of the
Requirements for the award of the degree of

BACHELOR OF COMPUTER APPLICATIONS

SUBMITTED BY

D.GOWTHAM BHASKAR (190377400029)

Under the Esteemed Guidance of

Mrs. G.SATYA,M.C.A,M.Tech
Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE

ADITYA DEGREE COLLEGE
(Affiliated to Adikavi Nannaya University)
Accredited by NAAC with B++Grade
Kakinada-533437, East Godavari District, Andhra Pradesh

2021-2022

ADITYA DEGREE COLLEGE

(Affiliated to Adikavi Nannaya University) Accredited by NAAC with B++)

Kakinada533437, East Godavari District, Andhra Pradesh.

2021-2022

DEPARTMENT OF COMPUTER SCIENCE

CERTIFICATE

This is to certify that the project work entitled, **"ONLINE BUS TICKET RESERVATION SYSTEM"** is a bonafide work of **D.GOWTHAM BHASKAR** bearing Regd.No**:190377400029** submitted to the faculty of Bachelor of Computer Applications**,** in partial fulfilment of the requirements for the award of the degree of **BACHELOR OF COMPUTER APPLICATIONS** from Aditya Degree College, Kakinada for the academic year 2021-2022.

Project Guide

Mrs. G.SATYA, M.C.A,M.Tech

Assistant Professor,

Department Of Computer Science

Aditya Degree College, KKD

Head of the Department

Mrs. D.Beulah, M.Tech(Ph.D)

Associate Professor,

Department Of Computer Science

Aditya Degree College, KKD

DECLARATION

I hereby declare that the project entitled **"ONLINE BUS TICKET RESERVATION SYSTEM"** done by my own and submitted to **Aditya Degree College, Kakinada** has been carried out by me alone under the guidance of **Mrs. G.SATYA.**

Place: Kakinada
Date: **(D.GOWTHAM BHASKAR)**

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of people who made it possible, whose constant guidance and encouragement crowned our efforts with success. It is a pleasant aspect that I have now the opportunity to express my gratitude for all of them.

The first person I would like to thank is my guide **Mrs.G.Satya,M.C.A,M.Tech, Assistant Professor, Aditya Degree College, Kakinada,** Her wide knowledge and logical way of thinking have made a deep impression on me. Her understanding, encouragement and personal guidance have provided the basis for this project. She is a source of inspiration for innovative ideas and her kind support is well known to all her students and colleagues.

I wish to thank **Mrs. D.Beulah, M.Tech(Ph.D.), Head of the Department, Aditya Degree College, Kakinada,** who has extended her support for the success of this project.

I also wish to thank **Dr.B.V.L.Naidu, Principal**, **Aditya Degree College, Kakinada,** who has extended his support for the success of this project.

I would like to thank all the **Management &Technical Supporting Staff** for their timely help throughout the project.

ABSTRACT

ABSTRACT

The Online Bus Reservation System is a web-based application that allows visitors to check bus ticket availability, buy bus ticket and pay the bus ticket online. This system is established for all the home/office users after gaining access from the administrator. Online Bus Reservation System provides bus transportation system, a facility to reserve seats, cancellation of seats and different types of enquiries which need an instant and quick reservation. This system can be used by the users in performing online reservation via internet for their all-business purposes. Users can use this program directly on their websites and no need to install it. The use of bus traveling is a large growing business in India and other countries; hence bus reservation system deals with maintenance of records of each passenger who had reserved a seat for a journey.

In these modern days the population has been increased and the travelling has become as one of their daily routines. With the increased number of customers or travellers it would require more man power. When we want to travel by bus, we usually go to the reservation centre and book a ticket for particular day. It is a time-consuming process and now we make this process easier and faster by creating an online web application that is used to book tickets by using digital method. Here the people can easily use the web application by simply clicking on the URL that we have provided. After clicking on it the user needed to enter the details about the location that user wanted to go along with that some of their personal details like the user's name, contact number, address etc., The user can easily book ticket from anywhere and after a successful booking the registration details will be displayed on the owner's desktop. Using simple components and programming techniques the web application is created. The users can easily travel from one place to another. It is a time saving process and user-friendly web application.

TABLE OF CONTENTS

CONTENTS **Page No**

4. TECHNOLOGY DESCRIPTION

LIST OF FIGURES

LIST OF SCREENS

CHAPTER-1

INTRODUCTION

1. INTRODUCTION

1.1 Brief Information about the project

Electronic Bus Ticketing

The ticket machines would end the use of the hefty 1.5-kg ticket racks carried by conductors. It would also end the practice of tearing out tickets and marking fare stages. The Conductor would just have to key in facilitate inspection by the corporation's checking inspectors. The ticket machines would help prevent loss on account of malpractice. It would also help in providing adequate data to the corporation, particularly with the details about the fare stage and the ticket machine would print out the ticket. The machine weighs only 800 grams and is convenient to carry. The parameters are almost like that of a railway ticket, the only difference being that the machine is portable. The machine can print out 2,300 tickets, including the journey report in order to regard to the boarding of passengers from fare stages and important points. This would help the corporation prepare and organize its schedules more efficiently on the basis of traffic demand. Besides, it would provide data on concessions given to various sections. Another additional feature is that the data in the ticket machine could be fed into the computer. More over the depots of the corporation would be fully computerized so we want to add some other modules in our domain for depot's verification.

Purpose of the project

The online bus reservation system is currently maintaining the project Transport company's process which is a very time consuming process. It deals with transport industry ticket booking and transport maintenance, so it becomes a very tedious job for the et booking transporter to look after these particulars to complete the task at right time. the online Bus reservation system not only deals with transporters owned vehicles but also takes with consideration about the other type project of the system transport vehicles available with other transporters To develop a software application that support specific to the project travel agency automation that can solve all tedious tasks related to ticket booking in a travel agency This system project is made as user friendly as possible so that any one can use it with little knowledge This system will lead to increase in the ticket booking efficiency of the project staff and members of the ticket booking agency with title throughout of computers .The ticket booking project will reduce the ticket booking tedious job of the system paperwork by keeping all the project details of bus ticket booking, cancelling tickets are stored in the form database in computers hard disk Up-to-date information of the system performance status and enquiry We provide up to dale information that is not possible manually.

The object of project is to make easy to the online Bus reservation system of ticket Booking Agency Simple, reliable ,user friendly, and corrective more over less time consuming as compared to manual work.

Report the project past several decades personal function has been transform from relatively obscure record keeping staff to central and top level management function. There are many factors that have influenced this System transformation like technological Advances, professionalism, and general recognition of human beings as most important resources. A computer based management system is designed to handle all the project primary. Information required to maintain the ticket booking project library management system. The searching of system record has been made the quite simple as all the ticket booking details of the customer and can be obtained by simply, keying in the reservation system ticket booking. Similarly record maintenance and updation can also be accomplished by using the identification of the ticket booking system customer with all the details being the automatically generated. These details are also being promptly automatically the updates in the master file thus keeping the record absolutely up-to-date.

The project entire information has maintained in the project database or files and who ever wants to retrieve can't retrieve only authorization user can retrieve the necessary information which can be easily be accessible from the file . The main purpose for the project entire activity is to automate the ticket booking process of the day today activates of system library like

- Ticket activities
- Creation of customer id
- Assign a bus tickets according to customers demand
- Advance bookings
- Bus ticket cancellation 6) Feedback

Scope

- The scope of the online bus ticket reservation of system is a person should be able to
- Login in the system through the first page of the applications.
- Change the password after logging into the system.

Should be able to create a new login for the accessing the reservation facility.

Query the buses for two weeks only two weeks advance reservation is available.

No reservation before two days can be done.

Able to choose the seats which are available for a certain class.

Give details about credit card.

- Improve and optimized services.
- Freight revenue enhancement.

As mentioned above, although our system had been completed but it is not perfect, we had planned to make some enhancement in the future

The system still has potential to grow. Besides we will include more functions and introduce more widgets to the system like.

Mobile apps Mobile version

Home delivery

Call centre support

Also planning to enhance the interface so that it looks more attractive and interactive

1.2 Motivation of project

This Project Is Suggested by Travel Agent. This Project Is Taken for The Purpose That There Is Lot Of

Problem Related to Buses Route, Timing, Ticketing, Trip Details, Bus Details, Bus Stops in Local Areas. More Importantly to Know the Running Time of A Particular Bus And We Want To Manipulate And Stores These Information Successful.

1.3 Objective of the project

The Bus ticket reservation system is currently maintaining the project Transport Company's process manually which is a very time-consuming process. It deals with transport industry's ticket booking and transport maintenance, so it becomes a very tedious job for the ticket booking transporter to look after these particulars to complete the task at right time. The bus ticket booking system not only deals with transporters owned vehicles but also takes into consideration about the other types project of system transport vehicles available with other transporters.

To develop a software application that supports Specific to the project Travel Agency Automation that can solve all tedious tasks related to ticket booking in a travel agency.

- This system will lead to an increase in the ticket booking efficiency of the project **Staff** and members of the Ticket Booking Agency with little throughput.
- This system project is Mauser-friendly as possible so anyone one can use it with little knowledge of system computers.

- The ticket booking project will reduce the ticket booking tedious job of system paperwork by keeping all the project details of bus ticket booking, cancelling tickets are stored in the form database in the computer's hard disk.
- The objective of my project is to make easy the ticket booking project system of Ticket Booking Agency simple, reliable, user friendly, and corrective. Moreover, less time-consuming as compared to manual work.

CHAPTER-2
SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

2.1 Existing System

Existing system refers to the system that is being followed till now. The existing system requires more computational time, more manual calculations, and the complexity involved in Selection of features is high. The other disadvantages are lack of security of data, Deficiency of Data accuracy, Time consuming etc. To avoid all these limitations and make the working more accurately the system needs to be computerized. Here in the Electronic bus ticketing, a detailed study of existing system is carried along with all the steps in system analysis.

Draw backs of existing system

Here in the Electronic bus ticketing, a detailed study of existing system is carried along with all the steps in system analysis. An idea for creating a better project was carried and the next steps were followed.

- Lack of security of data.
- More manpower.
- Time consuming.
- Consumes large volume of pare work.
- Needs manual calculations.
- No direct role for the higher officials.
- Damage of machines due to lack of attention.

To avoid all these limitations and make the working more accurately the system needs to be computerized.

2.2 Proposed System

The aim of proposed system is to develop a system of improved facilities. The proposed system can overcome all the limitations of the existing system. The system provides proper security and reduces the manual work. The existing system has several disadvantages and many more difficulties to work well. The proposed system tries to eliminate or reduce these difficulties up to some extent. The proposed system will help the user to reduce the workload and mental conflict. The proposed system helps the user to work user friendly and he can easily do his jobs without time lagging.

Advantages of Proposed System

The system is very simple in design and to implement. The system requires very low system resources and the system will work in almost all configurations. It has got following features

- Ensure data accuracy.
- Minimize manual data entry.

- Minimum time needed for the various processing Greater efficiency
- Better Service

- Minimum time required

The ticket machines would help prevent loss on account of malpractice
It would also help in providing adequate data to the corporation, particularly with regard to boarding of passengers from fare stages and important points

This would help the corporation prepare and organize its schedules more efficiently on the basis of traffic demand.

It would provide data on concessions given to various sections.
Another additional feature is that the data in the ticket machine could be fed into the computer.

2.3 Feasibility Study

All the projects are feasible, given unlimited resources and infinite time. So feasibility study is a test of system proposal, regarding its workability, impact on the organization, ability to meet user needs and effective use of resources. An important outcome of the investigation was the determination that the system requested was feasible. If the feasibility study is to serve as a decision document, it must answer the key questions:

- Is there a new and better way to do the job that will benefit the user?
- What are the costs and savings of the alternatives?
- What is recommended?

Three key considerations are involved in the feasibility study: Economic, Operational, Technical Feasibility.

Technical Feasibility

In an ever-changing software world, selecting one tool set and platform is a very difficult task. We should be extremely careful in the selection of the software platform and their tools become obsolete. If we ever select a platform or tool set of a company which are not there years to come, the major setback will be the service, and we will be left with no options other than abandoning the system. Then the next problem will be migration to a better system. Always we should be able to select a tool set platform, which can seamlessly integrate into other software platforms and the support for the future, should be ensured.

Here the proposed system is considered as technically feasible, because only the initial cost of buying a computer is needed. The considerations that are normally involved technical feasibility include development resources, availability and technology.

Operational Feasibility

So, we should select the platform and tools in such a way that, once the system is up and running. Getting the right manpower for that system to function successfully, should not be a cause to worry. The manpower should be easily Success of the new system pivots on its acceptance or non-acceptance by the organization. People are inherently resistant to change and computers have been known to facilitate change. The computer installations have something to do with turnover, transfer, retraining and changed to employee job status. Therefore, the introduction of a candidate system requires special effort to educate, sell and train the staff on new ways of conducting business.

The proposed system is user friendly and easy to use. The site could be accessed by both the admin and the customer. Work load is reduced for the admin and the total efficiency is improved.

Economic Feasibility

Economic analysis is the most frequently used method of evaluating the effectiveness of the candidate system and compare with the cost. More commonly known as cost-benefit analysis, the procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs.

If benefits outweigh costs, then a decision is made to design and implement the system.

The proposed system is cost effective in the sense; the number of computers, accessories and the software which have to be purchased, has to be optimized. If we could do the same task with a less powerful system with lesser price, the system study report should contain that information and the management can decide a very cos effective hardware purchase. Now to software side we have a client server based system to be implemented, of that the essential purchase of operating systems, database server and the front-end tool, we can go for so many options without compromising on the effectiveness and robustness of the proposed system. Computerized system reduces the manpower, thus the organization can save the salary for the employees also.

The proposed system is more economic than the existing system, since it uses GUI that reduces the operating time of the system and also it is interactive and user-friendly, so the chance of error is low and cost needed to correct the error is reduced.

Here the proposed system is considered to be economically feasible because a single system is only needed for the admin to upload the data to the site. Thus, it reduces the manual work. A person from anywhere in the world could access the information easily the same time instantaneously with much cost.

2.4 Functional Requirements

- Every online booking needs to be associated with an account.
- One account cannot be associated with multiple users.
- Search results should enable users to find the most recent and relevant booking options.
- System should enable users to book / pay for their tickets only in a timeboxed manner after tickets being added to the cart.
- System should only allow users to move to payment only when mandatory fields such as date, time, location has been mentioned.
- System should consider time zone synchronisation when accepting bookings from different time zones.
- Booking confirmation should be sent to user to the specified contact details.

2.5 Non-Functional Requirement

The major non-functional Requirements of the system are as follows

- Use of captcha and encryption to avoid bots from booking tickets
- System should accept payments via different payment methods, like PayPal, wallets, cards, vouchers, etc
- System should visually confirm as well as send booking confirmation to the user's contact

2.6 Minimum Hardware Requirements

- Processor: Intel Core i5
- Speed : 2.2 Ghz
- RAM : 8GB
- Hard Disk: 500GB

Software Requirements

- Operating system: Windows 11
- Back End : PHP
- Front End : JS,Html & CSS
- Data Base : MySql

CHAPTER-3
SYSTEM DESIGN

3.SYSTEM DESIGN

3.1 Introduction

The purpose of the design phase is to plan a solution of the problem specified by the requirement document. This phase is the first step in moving from the problem domain to the solution domain. In other words, starting with what is needed, design takes us toward how to satisfy the needs. The design of a system is perhaps the most critical factor affection the quality of the software; it has a major impact on the later phase, particularly testing, maintenance. The output of this phase is the design document. This document is similar to a blueprint for the solution and is used later during implementation, testing and maintenance. The design activity is often divided into two separate phases System Design and Detailed Design. System Design also called top-level design aims to identify the modules that should be in the system, the specifications of these modules, and how they interact with each other to produce the desired results. At the end of the system design all the major data structures, file formats, output formats, and the major modules in the system and their specifications are decided. During, Detailed Design, the internal logic of each of the modules specified in system design is decided. During this phase, the details of the data of a module is usually specified in a high level design description language, which is independent of the target language in which the software will eventually be implemented. In system design the focus is on identifying the modules, whereas during detailed design the focus is on designing the logic for each of the modules.

In other works, in system design the attention is on what components are needed, while in detailed design how the components can be implemented in software is the issue. Design is concerned with identifying software components specifying relationships among components. Specifying software structure and providing blueprint for the document phase. Modularity is one of the desirable properties of large systems. It implies that the system is divided into several parts. In such a manner, the interaction between parts is minimal clearly specified During the system design activities, Developers bridge the gap between the requirements specification, produced during requirements elicitation and analysis, and the system that is delivered to the user. Design is the place where the quality is fostered in development. Software design is a process through which requirements are translated into a representation of software.

3.2 Data dictionary

A data dictionary contains metadata and data about the database. The data dictionary is very important as it contains information such as what is in the database, who can access it, where is the database physically stored etc. The users of the database normally don't interact with the data dictionary, it is only handled by the database administrators. The data dictionary in general contains information about the following Names of all the database tables and their schemas.

Details about all the tables in the database, such as their owners, them security constraints, when they were created etc.

- Physical information about the tables such as where they are stored and how.

- Table constraints such as primary key attributes, foreign key information etc.
- Information about the database views that are visible.

Booking Table

Field Name	Data Type	Size	Description
From	VARCHAR	255	From Address
To	VARCHAR	255	To Address
Return Date	Date	Short	Return Journey
Departure Date	Date	Short	Departure Journey

Contact Table

Field Name	Data Type	Size	Description
Your Name	VARCHAR	255	Enter your name
Your Email	VARCHAR	255	Enter your email
Message	VARCHAR	255	Enter your Message

3.3 Data Flow Diagram

Data flow-oriented techniques advocate that the major data items handled by a system must be first identified and then the processing required on these data items to produce the desired outputs should be determined. The DFD (also called as bubble chart) is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on these data, and the output generated by the system. It was introduced by De Macro (1978), Gane and Sur son (1979). The primitive symbols used for constructing

DFD's are:

Symbols used in DFD

A circle represents a  process

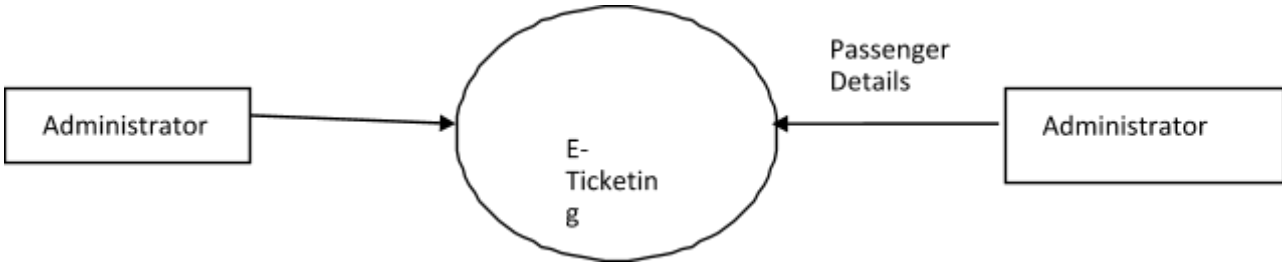
Rectangle represent external entity

A square defines a source or destination of the system data.

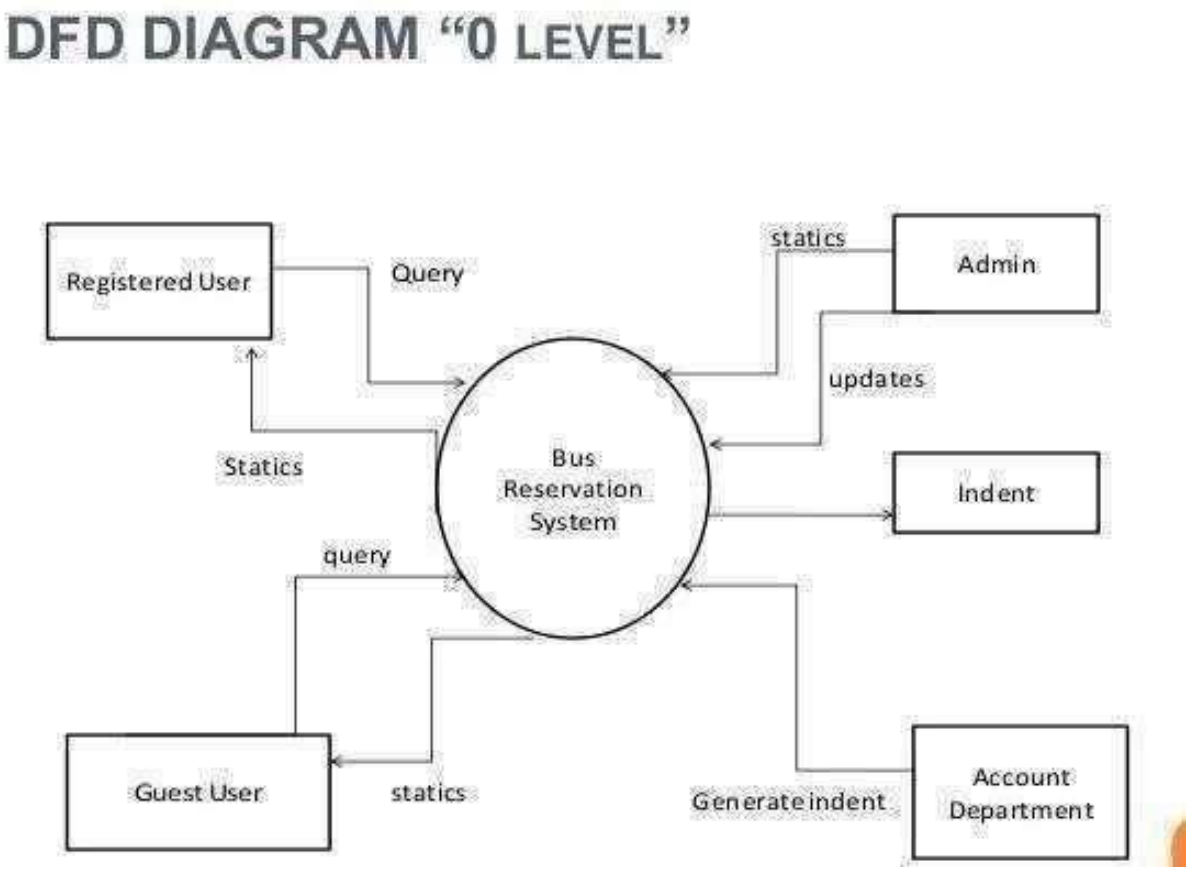
An arrow identifies dataflow.

Double line with one end closed indicates data store

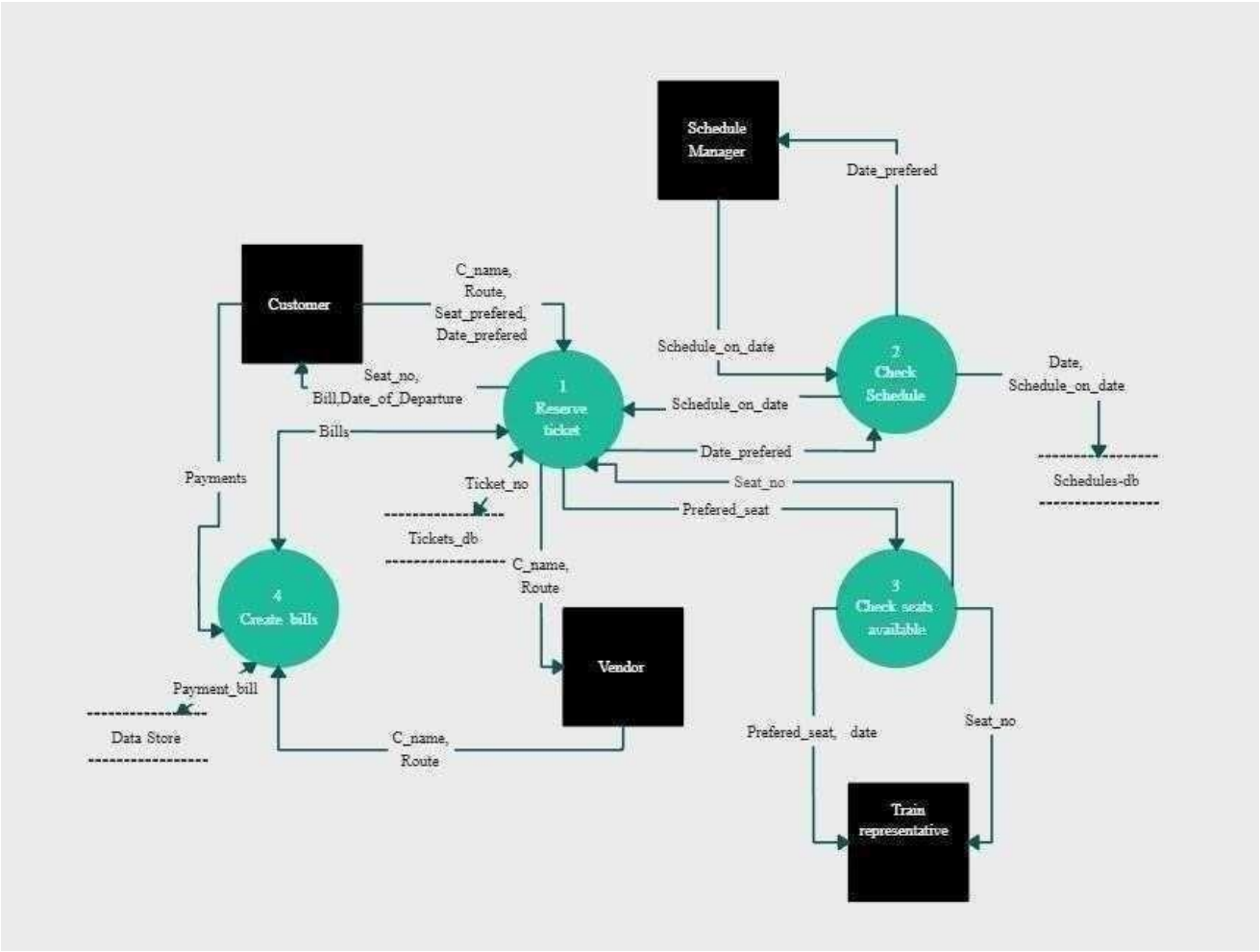
Data Flow Diagram



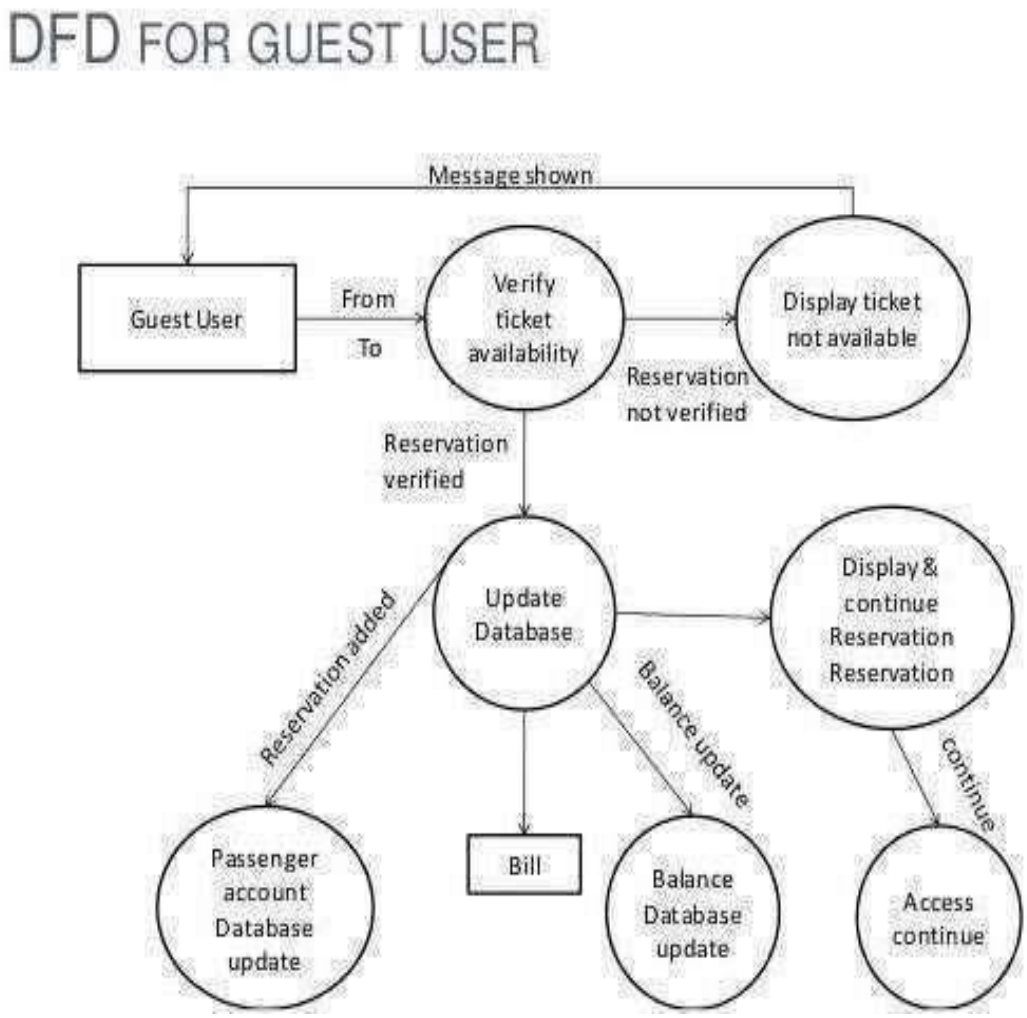
Context Diagram



Level 1


Schedule Manager
Date_prefered
Customer
C_name, Route, Seat_prefered, Date_prefered
Schedule_on_date
2 Check Schedule
Date, Schedule_on_date
Seat_no, Bill,Date_of_Departure
1 Reserve ticket
Schedule_on_date
Bills
Date_prefered
Schedules-db
Payments
Ticket_no
Seat_no
Prefered_seat
Tickets_db
C_name, Route
4 Create bills
3 Check seats available
Vendor
Payment_bill
Data Store
C_name, Route
Prefered_seat, date
Seat_no
Train representative

Level 2


DFD FOR GUEST USER
Message shown
Guest User
From
To
Verify ticket availability
Display ticket not available
Reservation not verified
Reservation verified
Update Database
Display & continue Reservation Reservation
Reservation added
Balance update
continue
Passenger account Database update
Bill
Balance Database update
Access continue

3.4 UML Diagrams

The unified modeling language allows the software engineer to express an analysis model using the modeling notation that is governed by a set of syntactic semantic and pragmatic rules. A UML system is represented using five different views that describe the system from distinctly different perspective. UML is specifically constructed through two different domains UML Analysis modeling, this focuses on the user model and structural model views of the system UML design modeling, which focuses on the behavioral modeling, implementation modeling and environmental model views. These are divided into the following types.

- **Use case diagram**
- **Class diagram**

3.4.1 Use Case Diagram

Use Case diagrams identify the functionality provided by the system (use cases), the users who interact with the system (actors), and the association between the users and the functionality. Use Cases are used in the Analysis phase of software development to articulate the high-level requirements of the system. The primary goals of Use Case diagrams include: ● Providing a highlevel view of what the system does.

- Identifying the users ("actors") of the system.
- Determining areas needing human-computer interfaces.

Graphical Notation The basic components of Use Case diagrams are the Actor, the Use Case, and the

Association.

Actor

An Actor, as mentioned, is a user of the system, and is depicted using a stick figure. The role of the user is written beneath the icon. Actors are not limited to humans. If a system communicates with another application, and expects input or delivers output, then that application can also be considered an actor.

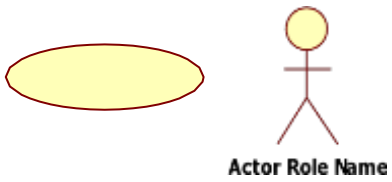
A Use Case is functionality provided by the system; Use Cases are depicted with an ellipse. The name of the use case is written within the

Use Case : ellipse These Associations are used to link

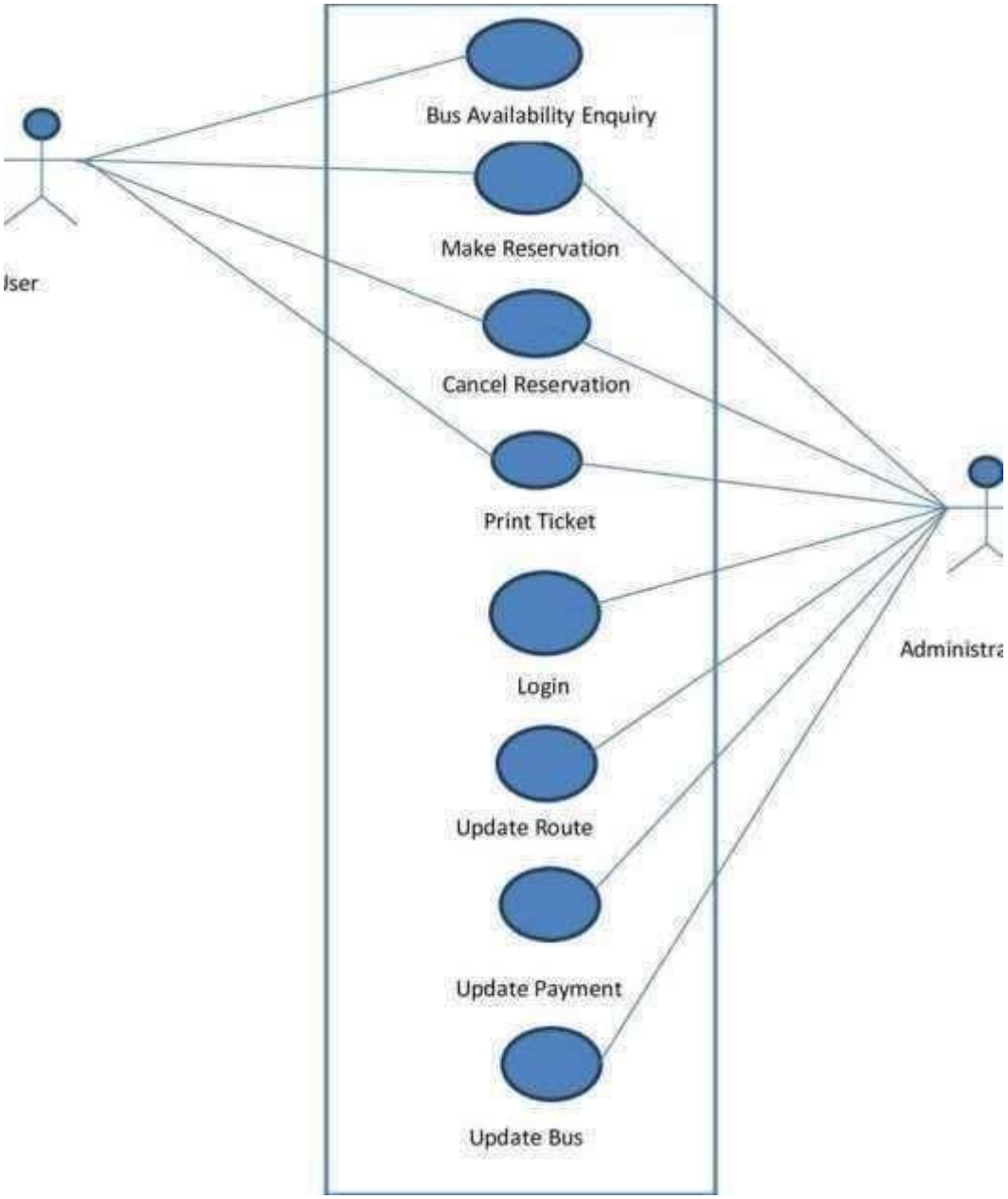
Actors with Use. **Association**

:

Cases, and indicate that an Actor participates in the Use Case in some form.



Behind each Use Case is a series of actions to achieve the proper functionality, as well as alternate paths for instances where validation fails, or errors occur. These actions can be further defined in a Use Case description. Because this is not addressed in UML, there are no standards for Use Case descriptions. However, there are some common templates can follow, and whole books on the subject writing of Use Case description.



Use case diagram

3.4.2 Class Diagram

Class diagrams identify the class structure of a system, including the properties and methods of each class. Also depicted are the various relationships that can exist between classes, such as an inheritance relationship. Part of the popularity of Class diagrams stems from the fact that many

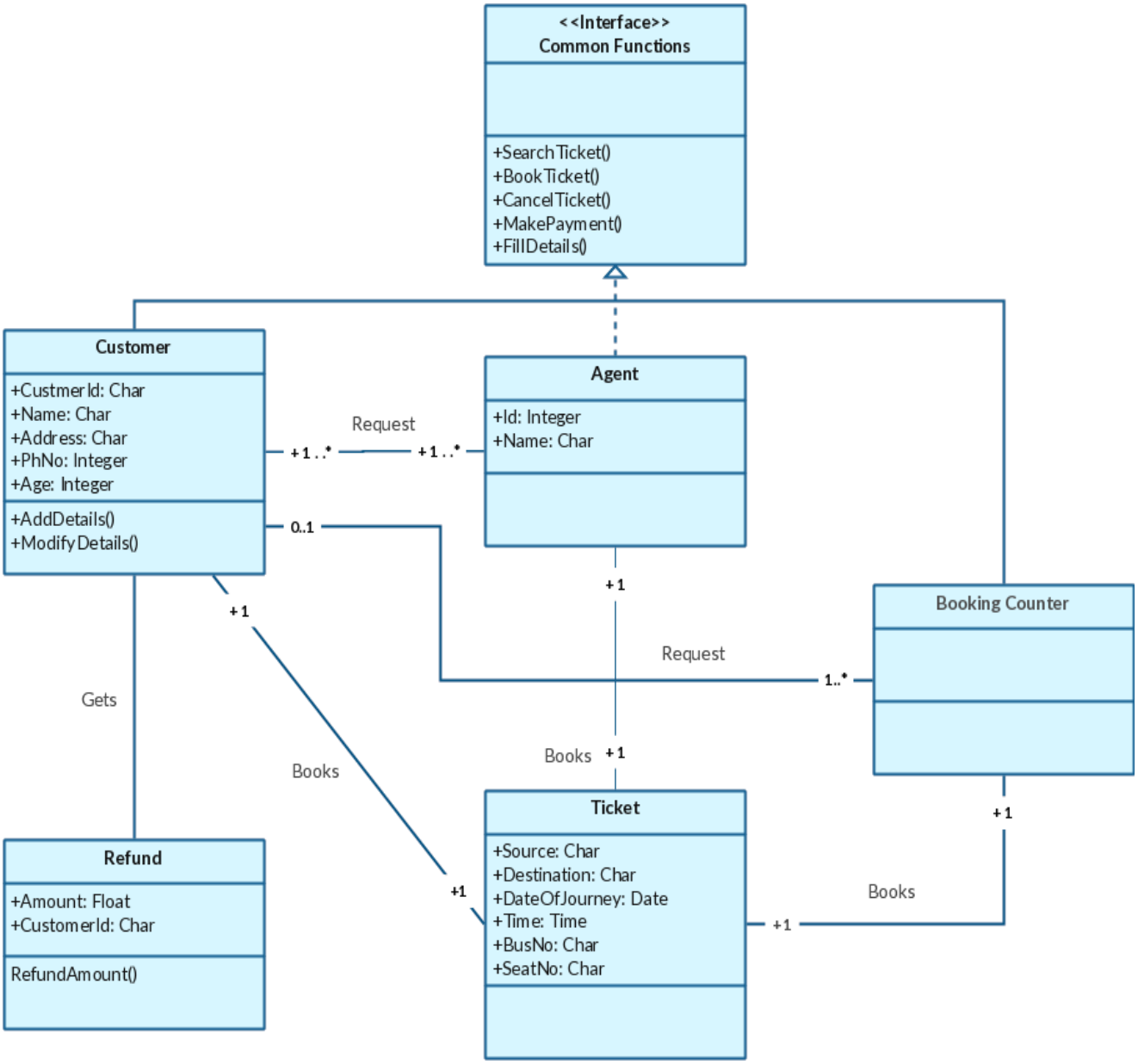
CASE tools, such as Rational XDE, will autogenerate code in a variety of languages, these tools can synchronize models and code, reducing the workload, and can also generate Class diagrams from object-oriented code. **Graphical Notation** The elements on a Class diagram are classes and the relationships between them.

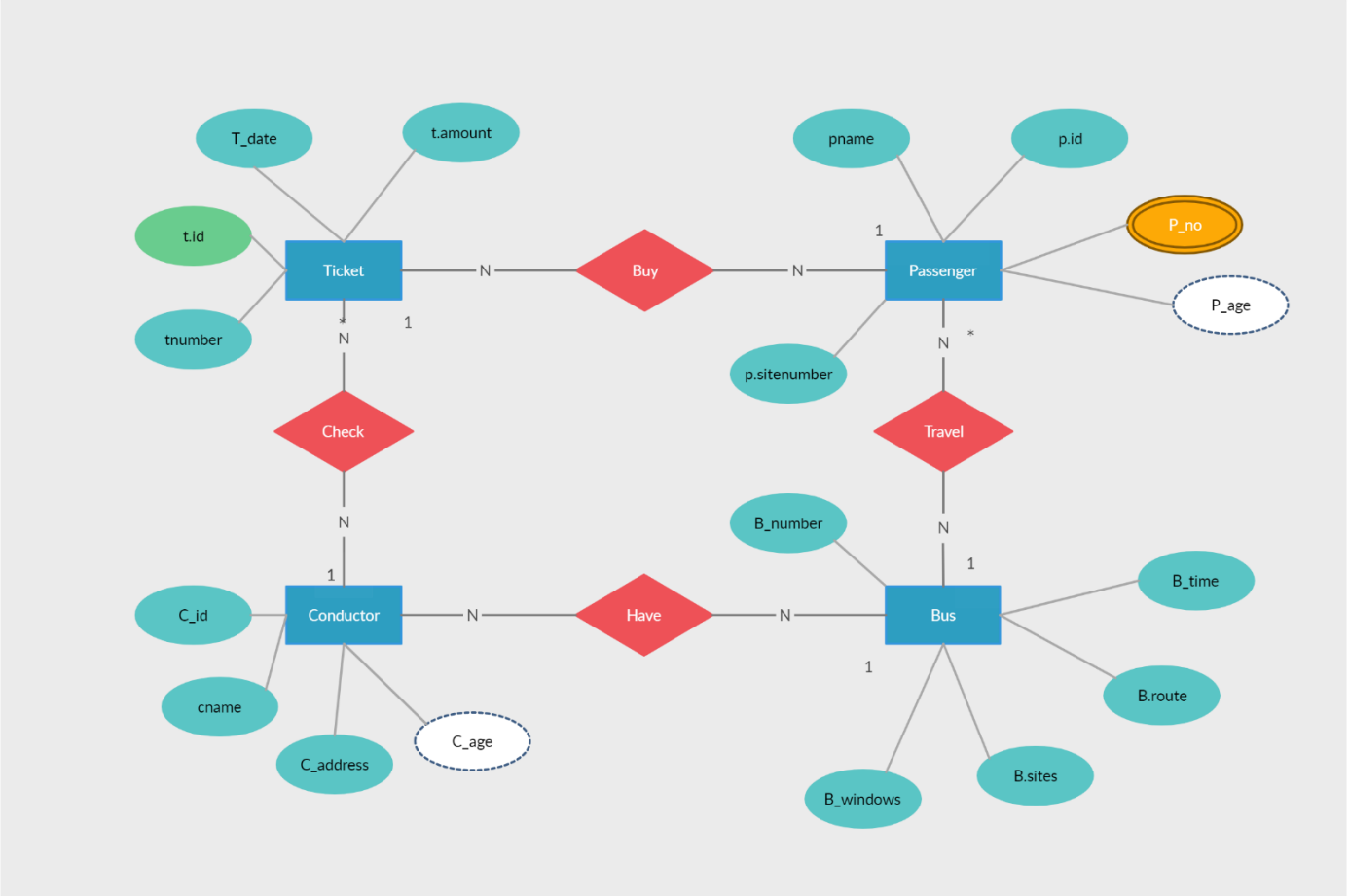
Class

Classes are the building blocks in object-oriented programming. A Class is depicted using a rectangle divided into three sections. The top section is the name of the Class. The middle section defines the properties of the Class. The bottom section lists the methods of the class.

Association An Association is a generic relationship between two classes, and is modeled by a line connecting the two classes. This line can be qualified with the type of relationship, and can also feature multiplicity rule (e.g. one-to-one, one-to-many, many-to-many) for the relationship.

Class diagram


<<Interface>>
Common Functions
+SearchTicket()
+BookTicket()
+CancelTicket()
+MakePayment()
+FillDetails()
Customer
+CustmerId: Char
+Name: Char
+Address: Char
+PhNo: Integer
+Age: Integer
+AddDetails()
+ModifyDetails()
Agent
+Id: Integer
+Name: Char
Request
+1..*
+1..*
0..1
+1
+1
Request
1..*
Booking Counter
Gets
Books
Books
+1
+1
+1
Refund
+Amount: Float
+CustomerId: Char
RefundAmount()
Ticket
+Source: Char
+Destination: Char
+DateOfJourney: Date
+Time: Time
+BusNo: Char
+SeatNo: Char
+1
Books


T_date
t.amount
t.id
Ticket
tnumber
N
1
Buy
N
pname
p.id
1
P_no
Passenger
P_age
p.sitenumber
N
*
Check
Travel
N
B_number
N
1
1
C_id
Conductor
N
Have
N
Bus
B_time
1
B.route
cname
C_age
C_address
B_windows
B.sites

ER diagram

CHAPTER- 4
TECHNOLOGY DESCRIPTION

4.TECHNOLOGY DESCRIPTION

4.1 HTML Introduction:

Hypertext Mark-up Language (HTML), the languages of the world wide web (WWW), allows users to produce web pages that included text, graphics and pointer to other web pages (Hyperlinks).HTML is not a programming language but it is an applications of ISO standard 8879, SGML (Standard Generalized Mark -up Language), but specialized to hypertext and adapted to the web. The idea behind Hypertext one point to another point. We can navigate through the information based on out interest and preference. A mark-up language is simply a series of enclosed within the elements should be displayed.

Hyperlinks are underlined or emphasized works that load to other documents or some portions of the same document. Html can be used to display any type of document on the host computer, which can be geographically at a different location. It is a versatile language and can be used on any platform or desktop. HTML provides tags (special code) to make the document look attractive. HTML provides or not case –sensitive. Using graphics, fonts, different sizes, colour, etc.. can enhance the presentation of the document. Anything that is not a tag is part of the document itself.

4.1.1 Basic HTML Tags:

< ! ---->	Specific comments.
<A>….</A>	Creates Hypertext links.
<B>….</B>	Creates Hypertext links.
<Big>….</Big>	Formatted text in large-font.

<Body>…..</Body> contains all tags and text in the Html-document <Center>…..</Center> Creates Text.

<DD>….</DD>	Definition of a term.
<TABLE>…</TABLE>	Creates table
<Td>……</Td>	indicates table data in a table.
<Tr>……..</Tr>	designates a table row.
<Th>……..<Th>	creates a heading in a table.

Advantages

- A HTML document is a small and hence easy to send over the net .It is a small because it does not includes formatted information
- HTML is a platform independent .

4.2 CSS Introduction

CSS is used to control the style of a web document in a simple and easy way.**CSS** is the acronym for **"Cascading Style Sheet"**. This tutorial covers both the versions CSS1,CSS2 and CSS3, and gives a complete understanding of CSS, starting from its basics to advanced concepts.**Cascading Style Sheets**, fondly referred to as **CSS**, is a simple design language intended to simplify the process of making web pages presentable.**CSS** is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. I will list down some of the key advantages of learning CSS:

- **Create Stunning Web site** - CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs,variations in display for different devices and screen sizes as well as a variety of other effects.

- **Become a web designer** - If you want to start a carrer as a professional web designer, HTML and CSS designing is a must skill.

- **Control web** - CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

Learn other languages - Once you understands the basic of HTML and CSS then other related technologies like javascript, php, or angular are become easier to understand.

Applications of CSS :

As mentioned before, CSS is one of the most widely used style language over the web. I'm going to list few of them here:

CSS saves time - You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.

Pages load faster - If you are using CSS, you do not need to write HTML tag attributes every time.

Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code

Easy maintenance - To make a global change, simply change the style, and all elements in all the web
pages will be updated automatically.

Superior styles to HTML - CSS has a much wider array of attributes than HTML, so you can give a
far better look to your HTML page in comparison to HTML attributes.

Multiple Device Compatibility - Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.

Global web standards - Now HTML attributes are being deprecated and it is being recommended to use CSS. So its a good idea to start using CSS in all the HTML pages to make them compatible to future browsers

4.3 PHP INTRODUCTION

The term PHP is an acronym for *PHP: Hypertext Preprocessor*. PHP is a server-side scripting language designed specifically for web development. It is open-source which means it is free to download and use. It is very simple to learn and use. The files have the extension ".php".Rasmus Lerdorf inspired the first version of PHP and participating in the later versions. It is an interpreted language and it does not require a compiler.

- PHP code is executed in the server.
- It can be integrated with many databases such as Oracle, Microsoft SQL Server, MySQL, PostgreSQL, Sybase, Informix.
- It is powerful to hold a content management system like WordPress and can be used to control user access.
- It supports main protocols like HTTP Basic, HTTP Digest, IMAP, FTP, and others.
- Websites like www.facebook.com, www.yahoo.com are also built on PHP.
- One of the main reasons behind this is that PHP can be easily embedded in HTML files and HTML codes can also be written in a PHP file.
- The thing that differentiates PHP from the client-side language like HTML is, PHP codes are executed on the server whereas HTML codes are directly rendered on the browser. PHP codes are first executed on the server and then the result is returned to the browser.

- The only information that the client or browser knows is the result returned after executing the PHP script on the server and not the actual PHP codes present in the PHP file. Also, PHP files can support other client-side scripting languages like CSS and JavaScript.

Other characteristics of PHP are as follows.

- Simple and fast
- Efficient
- Secured
- Flexible
- Cross-platform, it works with major operating systems like Windows, Linux, MacOS.

Syntax:

```
<?php
    PHP code goes here
?>
```

4.4 MODULES Module Description

This project is modularized as the following:

1. Management of Routes Module:

This module include information about how we can Manage the routes for a particular bus services so In the case of Route management module we must know the details about route number, number of stops, fare stages and running time of the particular bus more over we want to manipulate and stored these information successfully.

2. Trip information Module:

Each journey is identified as a trip. Each ticket must contain the trip no so that calculation of passenger can be done easily. Here in this section we want to know start time and route no of the bus this information can be manipulate and stored successfully.

3. Bus Detail Module:

In this module all bus details are stored and manipulated, in bus detail module contains minimum charge, type, depot, fare increment, bus number, and passenger's states (child or adult) are manipulated and stored.

4. Bus Stops Module:

Bus Stops module includes information about what are the main bus stops of a particular bus. This module connected to the route of the bus and it is used to store stop number, stop name and fare stages and Route number.

5. Bus Ticketing Module:

Ticketing is the most important module in this Project which uses all the table together and calculates fare for the pass, beginning stop, end stop, ticket number persons(Adult/child)rate, date and time also we want to print the all these information. In order to do the calculation data has to be pulled out from stops, bus, trip and route. Number of passengers & the states are entered by the Venter and to produce the tickets.

CHAPTER-5
SAMPLE CODE

5. SAMPLE CODE

HOME PAGE

```
<!DOCTYPE html>
<html lang="">
 <head>
      <meta charset="utf-8">
      <meta http-equiv="X-UA-Compatible" content="IE=edge">
      <meta name="viewport" content="width=device-width, initial-scale=1">
      <title>Online Ticket Reservation</title>

      <!-- Bootstrap CSS -->
   <link rel="stylesheet" type="text/css" href="../assets/css/bootstrap.min.css">
   <link rel="stylesheet" type="text/css" href="../assets/css/bootstrap-theme.min.css">

 </head>
<body>

<nav class="navbar navbar-inverse">
 <div class="container-fluid">
      <a class="navbar-brand" href="#"> Online Ticketing</a>
      <ul class="nav navbar-nav">
               <li>
                        <a href="#"></a>
               </li>
               <li>
                        <a href="#"></a>
               </li>
      </ul>
      <ul class="nav navbar-nav navbar-right">
     <li><a href="../"><span class="glyphicon glyphicon-backward"></span> Return Home</a></li>
</ul>
 </div>
</nav>



<div class="col-md-3"></div>
<div class="col-md-6">
 <div class="panel panel-success">
```

```
    <div class="panel-heading">
            <h3 class="panel-title">Please Login Here</h3>
    </div>
    <div class="panel-body">
            <form class="form-horizontal" role="form" id="form-login">
              <div class="form-group">
                <label class="control-label col-sm-2" for="un">Username:</label>
                <div class="col-sm-10">
     <input type="text" class="form-control" id="un" placeholder="Enter Username" autofocus="" required="">
                </div>
              </div>
              <div class="form-group">
                <label class="control-label col-sm-2" for="pwd">Password:</label>
                <div class="col-sm-10">
                 <input type="password" class="form-control" id="pwd" placeholder="Enter password" required="">
                </div>
              </div>
              <div class="form-group">
                <div class="col-sm-offset-2 col-sm-10">
                  <div class="checkbox">
                    <label><input type="checkbox"> Remember me</label>
                  </div>
                </div>
              </div>
              <div class="form-group">
                <div class="col-sm-offset-2 col-sm-10">
                  <button type="submit" class="btn btn-default">Login
                  <span class="glyphicon glyphicon-check" aria-hidden="true"></span>
                  </button>
                </div>
              </div>
            </form>
    </div>
 </div>
</div>
<div class="col-md-3"></div>


<?php require_once('modal/message.php'); ?>

<script type="text/javascript" src="../assets/js/jquery-3.1.1.min.js"></script> <script
type="text/javascript" src="../assets/js/bootstrap.min.js"></script>

<script type="text/javascript">
```

```
 $(document).on('submit', '#form-login', function(event)
{     event.preventDefault();          /* Act on the event */
      // console.log('test');
      var un = $('#un').val();
      var pwd = $('#pwd').val();

      $.ajax({                              url:
'../data/login.php',                        type:
'post',                           dataType:
'json',
                                  data: {
                                  un: un,
                                  pwd : pwd
                                  },
                        success: function (data) {
                                  // console.log(data);
                                  if(data.valid == true){
                                            window.location = data.url;
                                  }else{
                                            $('#modal-message').find('#body-cont').text(data.msg);
                                            $('#modal-message').modal('show');
                                            $('#un').val("");
                                            $('#pwd').val("");
                                            $('#un').focus();
                                  }
                        },
                        error: function(){
                                  alert('Error: L99+');
                        }//
                });
 });

</script>

</body>
</html>
```

RESERVATION PAGE

```
<?php
require_once('session_login.php');
 ?>

 <!DOCTYPE html>

<html lang="">
 <head>
      <meta charset="utf-8">
      <meta http-equiv="X-UA-Compatible" content="IE=edge">
      <meta name="viewport" content="width=device-width, initial-scale=1">
      <title> Admin Panel </title>

      <!-- Bootstrap CSS -->
      <link rel="stylesheet" type="text/css" href="../assets/css/bootstrap.min.css">
   <link rel="stylesheet" type="text/css" href="../assets/css/bootstrap-theme.min.css">

    <!-- Custom CSS -->
   <link href="../assets/css/simple-sidebar.css" rel="stylesheet">
   <link href="../assets/css/dataTables.bootstrap.min.css" rel="stylesheet">

 </head>
<body>

<nav class="navbar navbar-inverse">
 <div class="container-fluid">
      <a class="navbar-brand" href="#">Online Ticketing Admin Panel</a>
      <ul class="nav navbar-nav">
               <li class="active"><a href="reservation.php">Reserved
               <span class="glyphicon glyphicon-list" aria-hidden="true"></span>
               </a></li>
               <li class=""><a href="transaction.php">Transaction
               <span class="glyphicon glyphicon-usd" aria-hidden="true"></span>
               </a></li>
      </ul>
      <ul class="nav navbar-nav navbar-right">
    <li><a  href="../admin/logout.php"><span  class="glyphicon  glyphicon-log-out"></span>  Logout</a></li>
</ul>
 </div>
</nav>
```

```
<br />
<div class="container-fluid">
 <div class="col-md-1"></div>
<div class="col-md-10">
      <div id="book-table"></div>
 </div>
 <div class="col-md-1"></div>
</div>


<?php require_once('modal/view_passenger.php'); ?>
<?php require_once('modal/message.php'); ?>
<?php require_once('modal/confirmation.php'); ?>

<script type="text/javascript" src="../assets/js/jquery-3.1.1.min.js"></script>
<script type="text/javascript" src="../assets/js/bootstrap.min.js"></script>
<script type="text/javascript" src="../assets/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" src="../assets/js/dataTables.bootstrap.min.js"></script>

<script type="text/javascript">
 function             showAllBook(){
      $.ajax({
                        url: '../data/get_all_book.php',
                        type: 'post',
      // data: {},
                        success: function (data) {
                                 // console.log(data);
                                 $('#book-table').html(data);
                        },
                        error: function(){
                                 alert('Error: L54+');
                        }
                });
 }//end showAllBook
 showAllBook();

 var book_tracker;  function
deleteBook(tracker){        //
console.log(tracker);
      book_tracker = tracker;
      $('#modal-confirmation').modal('show');
 }//end deleteBook

 $('.del-book').click(function(event) {
      /*   Act   on   the   event   */
      $.ajax({
```

```
                url: '../data/deleteBook.php',
                type: 'post',
                dataType: 'json',
                data: {
                        tracker : book_tracker
                },
                success: function (data) {
                        // console.log(data);
                        $('#modal-confirmation').modal('hide');
                        showAllBook();
                        $('#modal-message').find('.modal-body').text(data.msg);
                        $('#modal-message').modal('show');
                },
                error: function(){
                        alert('Error: L87+');
                }
        });
});//del


function displayPassenger(tracker){
    //          console.log(tracker);
    $.ajax({
                url:        '../data/getPassengers.php',
                type: 'post',
                // dataType: 'json',
                data: {
            tracker : tracker
                },
                success: function (data) {
                        // console.log(data);
                        $('#passenger-list').html(data);
                },
                error: function(){
                        alert('Error: L113+');
                }
        });
}//end displayPassenger

function viewBook(tracker){
    // $('#modal-view-pass').modal('show');

    $.ajax({
```

```
                    url: '../data/getBookBy.php',
                    type: 'post',      dataType:
'json',
                    data: {
                            tracker : tracker
                    },
                    success: function (data) {
                            // console.log(data);
                            $('#book-by').text(data.book_by);
                            $('#date').text(data.book_departure);
                            $('#contact').text(data.book_contact);
                            $('#address').text(data.book_address);
                            $('#modal-view-pass').modal('show');
                    },
                    error: function(){
                    alert('Error: L113+');
                    }
            });
      displayPassenger(tracker);
 }//end viewBook

 function acceptPayment(){          var
counter = $('#i').val();       var tot = 0;
      for(var i = 0; i < counter; i++){
      var name = $('#name'+i).val();
      var disc = $('#disc'+i).val();
      var price = $('#price'+i).val();
      var discounted = price * disc;
      $('#pri'+i).text(discounted);
      tot += Number(discounted);

      }//for loop
      $('#total').text(tot);

 }//end acceptPayment

 function addTransaction(){          var
counter = $('#i').val();       var tot = 0;
      for(var i = 0; i < counter; i++){
      var name = $('#name'+i).val();
      var disc = $('#disc'+i).val();
      var price = $('#price'+i).val();
      var discounted = price * disc;
      $('#pri'+i).text(discounted);
      tot += Number(discounted);
                    $.ajax({
```

```
                url: '../data/save_transaction.php',
    type: 'post',               dataType:
'json',         data: {                 bid
: name,
                        disc : disc
                },
    success: function (data) {             console.log(data);
                $('#modal-view-pass').modal('hide');
    showAllBook();
                        $('#modal-message').find('.modal-body').text('Transaction Save Successfully!');
                        $('#modal-message').modal('show');
                },
                error: function(){
                        alert('Error: L162+');
                }
            });
    }//for loop
    $('#total').text(tot);
 }
</script>

</body>
</html>
```

LOGIN SESSION

```
<?php
if(session_status() == PHP_SESSION_NONE)
{
 session_start();//start session if session not start
}

if(!isset($_SESSION['logged'])){
header('location: index.php');
}
```

LOGOUT SESSION

```
<?php
if(session_status() == PHP_SESSION_NONE)
{
 session_start();//start session if session not start
}

unset($_SESSION['logged']);
```

```
header('Location: index.php');
```

TRANSACTION CLASS

```
<?php
require_once('../database/Database.php'); require_once('../interface/iTransaction.php');
class Transaction extends Database implements iTransaction
 {

 public function addTransaction($pay, $pass, $age, $gen, $acc_id, $orig_id, $dest_id)
 {
     $sql = "INSERT INTO transaction (trans_payment, trans_passenger, trans_age, trans_gender,
             acc_id, origin_id, dest_id)
                     VALUES(?,?,?,?,?,?,?);
     ";
     return $this->insertRow($sql, [$pay, $pass, $age, $gen, $acc_id, $orig_id, $dest_id]);  }//end
addTransaction

 public function getAllTransaction()
 {
     $sql = "SELECT *
                     FROM transaction t
                     INNER JOIN accomodation a
                     ON t.acc_id = a.acc_id
             ";
     return $this->getRows($sql);
 }//end getAllTransaction

 public function getTransData($t_id)
 {
     $sql = "SELECT *
                     FROM transaction
                     WHERE trans_id = ?;
     ";
     return $this->getRow($sql, [$t_id]);
 }//end getTransData

 public function updateTrans($trans_id, $payment)
 {
     $refunded = 1;
     $sql = "UPDATE transaction
                     SET trans_payment = ?, trans_refunded = ?
```

```
                    WHERE trans_id = ?;
        ";
        return $this->updateRow($sql, [$payment, $refunded, $trans_id]);

 }//end updateTrans

}//end class Transaction

$transaction = new Transaction();
```

TRANSACTIONS PAGE

```
<?php
require_once('session_login.php');
 ?>
 <!DOCTYPE html>
<html lang="">
 <head>
      <meta charset="utf-8">
      <meta http-equiv="X-UA-Compatible" content="IE=edge">
      <meta name="viewport" content="width=device-width, initial-scale=1">
      <title> Admin Panel</title>

      <!-- Bootstrap CSS -->
      <link rel="stylesheet" type="text/css" href="../assets/css/bootstrap.min.css">          <link rel="stylesheet"
type="text/css" href="../assets/css/bootstrap-theme.min.css">

    <!-- Custom CSS -->
   <link href="../assets/css/simple-sidebar.css" rel="stylesheet">
   <link href="../assets/css/dataTables.bootstrap.min.css" rel="stylesheet">

 </head>
<body>

<nav class="navbar navbar-inverse">
 <div class="container-fluid">
      <a class="navbar-brand" href="#">Online Ticketing Admin Panel</a>
      <ul class="nav navbar-nav">
              <li class=""><a href="reservation.php">Reserved
              <span class="glyphicon glyphicon-list" aria-hidden="true"></span>
              </a></li>
              <li class="active"><a href="transaction.php">Transaction
              <span class="glyphicon glyphicon-usd" aria-hidden="true"></span>
              </a></li>
      </ul>
```

```
      <ul class="nav navbar-nav navbar-right">
     <li><a  href="../admin/logout.php"><span  class="glyphicon  glyphicon-log-out"></span>  Logout</a></li>
</ul>
 </div>
</nav>
<br />
<div class="container-fluid">
 <div class="col-md-1"></div>
 <div class="col-md-10">
      <div id="trans-table"></div>
 </div>
 <div class="col-md-1"></div>
</div>


<?php require_once('modal/view_passenger.php'); ?>
<?php require_once('modal/message.php'); ?>
<?php require_once('modal/confirmation.php'); ?>

<script type="text/javascript" src="../assets/js/jquery-3.1.1.min.js"></script>
<script type="text/javascript" src="../assets/js/bootstrap.min.js"></script>
<script type="text/javascript" src="../assets/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" src="../assets/js/dataTables.bootstrap.min.js"></script>

<script type="text/javascript">
 function showAllTransaction()
 {
      $.ajax({
                        url: '../data/get_all_transaction.php',
                        type:    'post',
                        // data: {},
                        success: function (data) {
      $('#trans-table').html(data);
                        },
                        error: function(){
                                  alert('Error: L70+');
                        }
                });
 }//end transaction
 showAllTransaction();



 var t_id,per;  //10 percent  function
tenPercent(trans_id, perc){          //
```

```
console.log(trans_id);        t_id =
trans_id;     per = perc;
      $('#modal-confirmation').modal('show');
 }//end tenPercent

 $('.tenPercent').click(function(event) {
      /* Act on the event */
      $.ajax({
                        url: '../data/refundTen.php',
                        type: 'post',
      dataType: 'json',                         data:
{                                 t_id : t_id,
                                  per : per
                        },
                        success: function (data) {
                        // console.log(data);
               if(data.valid == false){
                                    $('#modal-message').find('.modal-body').text('Passenger is Already Refunded!');

                                  }else if(data.valid == true){
                                          showAllTransaction();
                                          $('#modal-message').find('.modal-body').text('Passenger                    Refunded
Successfully!');
                                  }
                                          $('#modal-confirmation').modal('hide');
                                          $('#modal-message').modal('show');
                        },
                        error: function(){
                                  alert('Error: L90+');
                        }
               });
 });//end ten%


 //20
</script>

</body>
</html>
```

CHAPTER-6
TESTING

6. TESTING

Testing is vital to the success of the system. System testing makes a logical assumption that if all parts of the system are corrected, the goal will be successfully achieved. Inadequate testing non testing leads to errors that may not appear until months later.

The testing of this project ensures that the data received by the user is accurate. The project gives details of different books available in the book stall according to the customer wish. This is ensured in the testing. Testing should systematically uncover different classes of errors in a minimum amount of time with a minimum amount of efforts. Two classes of inputs are provided to test the process

A software configuration that includes a software requirement specification, a design specification and source code. A software configuration that includes a test plan and procedure, any testing tool and test cases and their expected results.

Testing is divided into several distinct operations:

TYPES OF TESTING

Unit Testing

In computer programming, a unit test is a procedure used to validate that a particular module of source code is working properly. The procedure is to write test cases for all functions and methods so that whenever a change causes a regression, it can quickly be identified and fixed. Ideally, each test case is separate from others.

This project has so many modules like new book insertion, deletion, update of the book details. All the individual modules are tested and validated for checking whether it gives the desired output. After validating these modules we can say that all the modules of these system is working perfectly and giving the desired output required by the administrator.

White Box Testing

Integration testing (sometimes called integration and testing and abbreviated I&T) is the phase of software testing in which individual software modules are combined and tested as a group. It follows unit testing and precedes system testing. Integration testing takes as its input values that have been unit tested, groups them in larger aggregates, applies tests defined in and integrated test plan to those aggregates and delivers as its output the integrated system ready for system testing The purpose of integration testing is to verify functional performance and reliability requirements placed in major design items.

Form validation involves checking all the form constraints like arithmetic, syntax, logical errors. Database validation involves checking constraints like primary key, foreign key, and all the database validations avoiding data redundancy. All the modules of the system are tested and now the modules are combined together and the integrated module is tested and validated for checking whether the combined modules are working perfectly or not.

Basic Path Testing

Validation testing is a concern which overlaps with integration testing ensuring that the application fulfills its specification, is a major criterion of integration testing. Validation, testing also overlaps to a large extend with system testing, where the application is tested with respect to its typical working environment.

Consequently for many processes no clear division between validation and system testing can be made validation testing.

Specific tests can be performed in either or both stages include the following:

Conditional Testing

where this version of the software is tested with the automated test harnesses, used with previous versions to ensure that the required features of the previous versions are still working in the new version.

Data Flow Testing

where the software is deliberately interrupted in a number of ways, for example taking its hard disk off line or even turning the computer off, to ensure that the appropriate techniques for restoring any lost data will function. It is a system that forces the software to fail in a variety of ways and verifies that the recovery is properly performed.

where unauthorized attempts to operate the software or part of it, are attempted. It might also include attempts to obtain access the data or harm the software installations or even the system software. A with all types of security it is recognized that someone sufficiently determined will

be able to obtain unauthorized access and the best that can be achieved is to make this process as difficult as possible .It attempts to verify that protection mechanisms built into a system will in fact protect it from improper penetration. The system's security must of course be tested from in vulnerability form frontal attack.

Loop Testing

where abnormal demands are made upon the software by increasing the rate at which it is asked to produce information. More complex tests may attempt to create very large data sets or cause the software to make excessive demands on the operating system. Stress tools are designed to confront programs with abnormal situations. Stress testing executes a system in a manner that demands resources in abnormal quantity and volume.

Performance Testing

performance requirements, if any, are checked. These may include the size of the software when

installed the amount of main memory and/or secondary storage it requires and the demands made

of the operating system when running within normal limits of the response

time

Black Box Testing

Black box testing is done to find out the following information as shown in below:

1. Incorrect or missing functions.

2. Interface errors.

3. Errors or database access.

4. Performance error.

5. Termination error.

The mentioned testing is carried out successfully for this application according to the user's requirement specification.

Test Data Output

After preparing test data, the system under study is tested using the test data. While testing the system using test data, errors are again uncovered and corrected by using above testing and corrections are also noted for future use.

Implementation

Implementation is the process of converting the system design into code, testing the system and giving the user training. Implementation of a new system design is a crucial phase in the system development life cycle.

There are three types of implementations:

Implementation of computer system to replace manual system

In this type of implementation, the existing manual system is replaced to create a computerized system which manages the same work but in an efficient way. In this project "Book Stall Management System" the existing manual system is to be replaced with the fully automated system. The problem encountered are converting files, creating accurate files and verifying printouts for integrity.

Implementation system to replace existing system

In this type of implementation, the existing system is replaced by the new system which deals the problems and characteristics of the system in a very different way. The only advantage of this implementation is that detailed study about the current system is not required. This is usually a difficult conversion. If not properly planned there can be many problems.

Implementation of modified application to replace an existing one

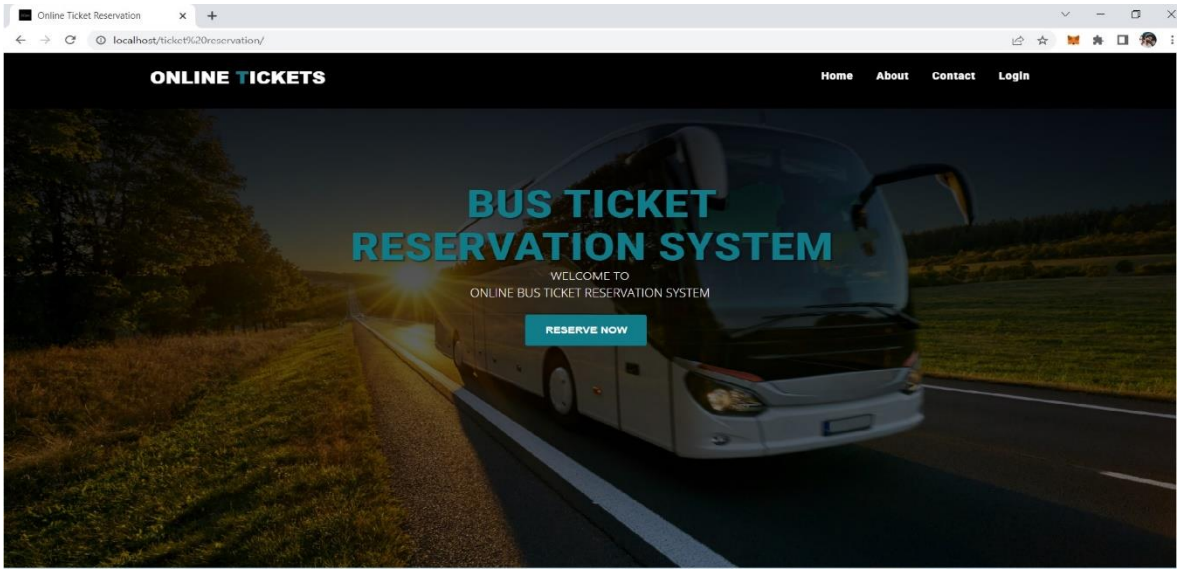
In this implementation the current system is modified in the proposed system to include new modules or to edit the bugs in the current system. The advantage is that there need not to be much effort to be made in the design phase. The important point is in understanding the current system and understanding its deficiencies accurately. This type of conversion is relatively easy to handle provided there are no major changes in the files.

TESTING TABLE

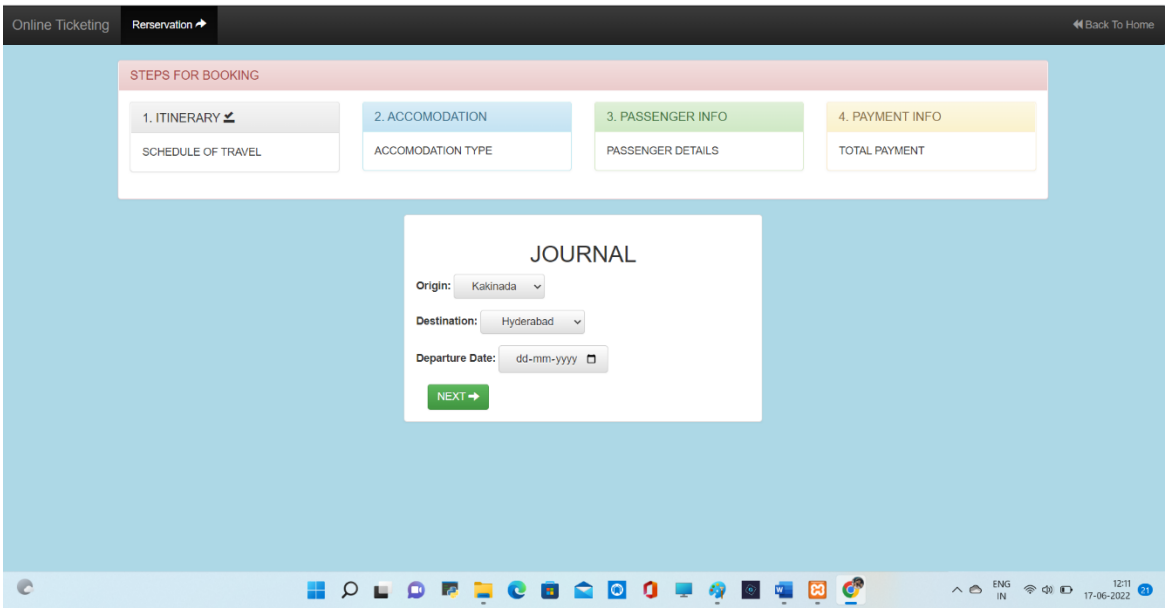
ID	Test Case	Expected Output	Actual Output	Conclusion
01	To show main page.	Any user can access this page and view the list of main sections provided on the platform.	User can access this page and view the list of main sections provided on the platform properly.	This function works effectively.
02	To give the users permission to view the login page and enter their details.	Any user can view the login page and input their details based on their user type/role.	Users can view the login page and input their details properly.	This function works effectively.
03	To give the users permission to view the registration page and enter their details.	Any user can view the registration page.	Users can view the registration page and input their details properly.	This function works properly.
04	To give the users permission to see the “About Us” Page	Any user can view the About Us Page.	Any user can view the login page and input their details based on their user type/role.	This function works properly.
05	To give the users permission to see the “Contact Us” Page	Any User can view the Contact us Page.	Any User can view the Contact Us Page	This function works Properly.

CHAPTER – 7
SCREEN SHOTS

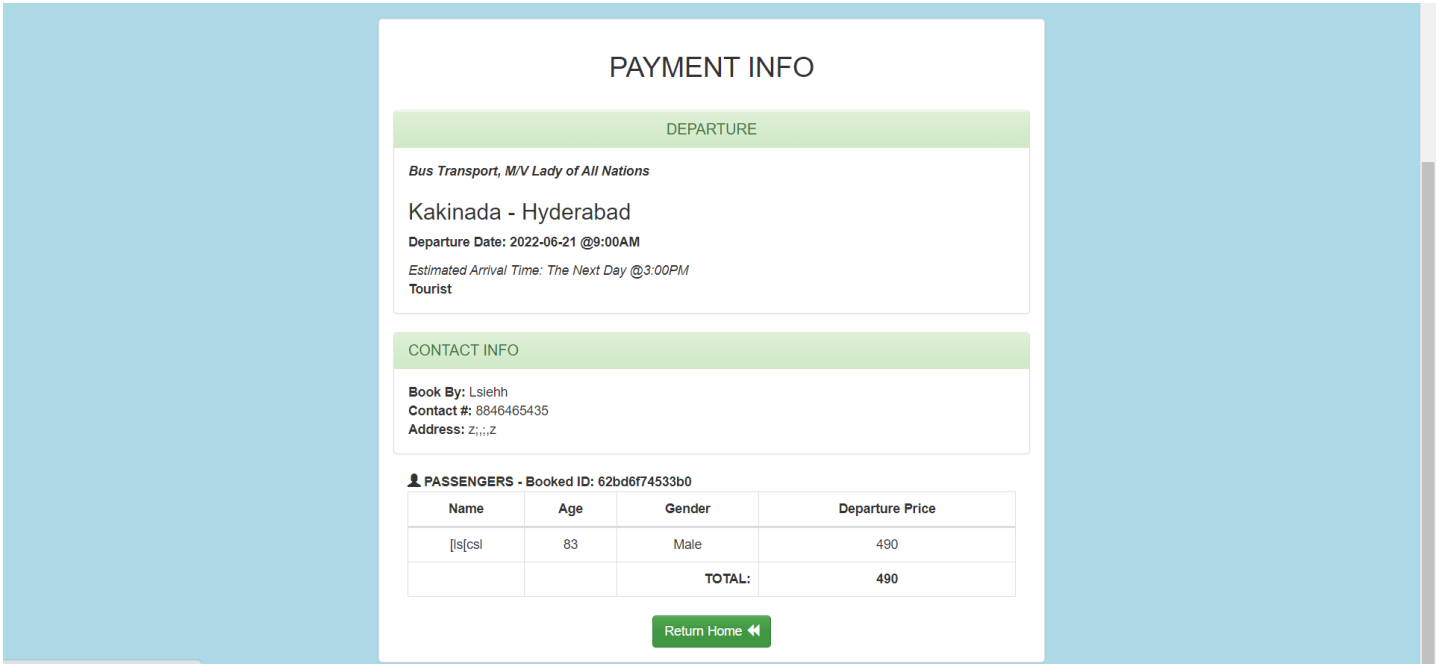
HOME PAGE



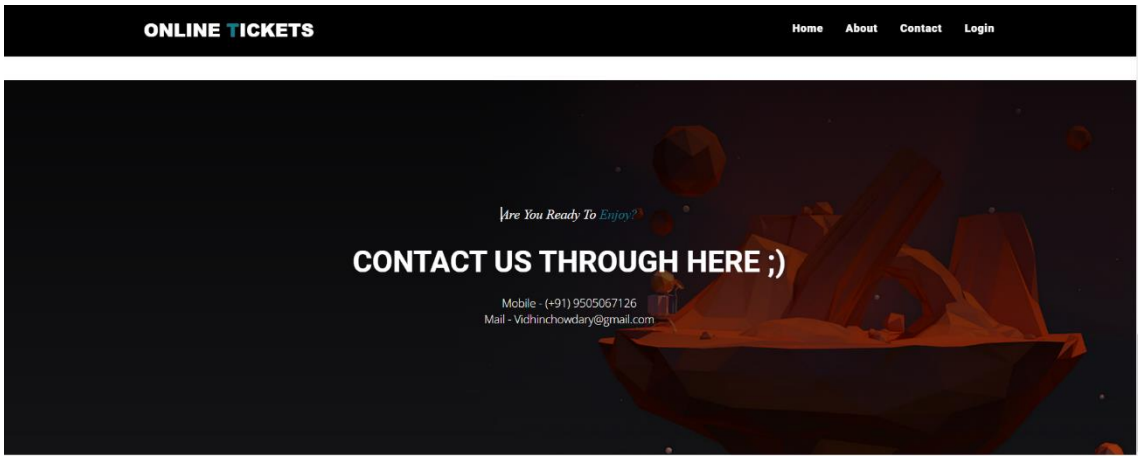
RESERVATION PAGE



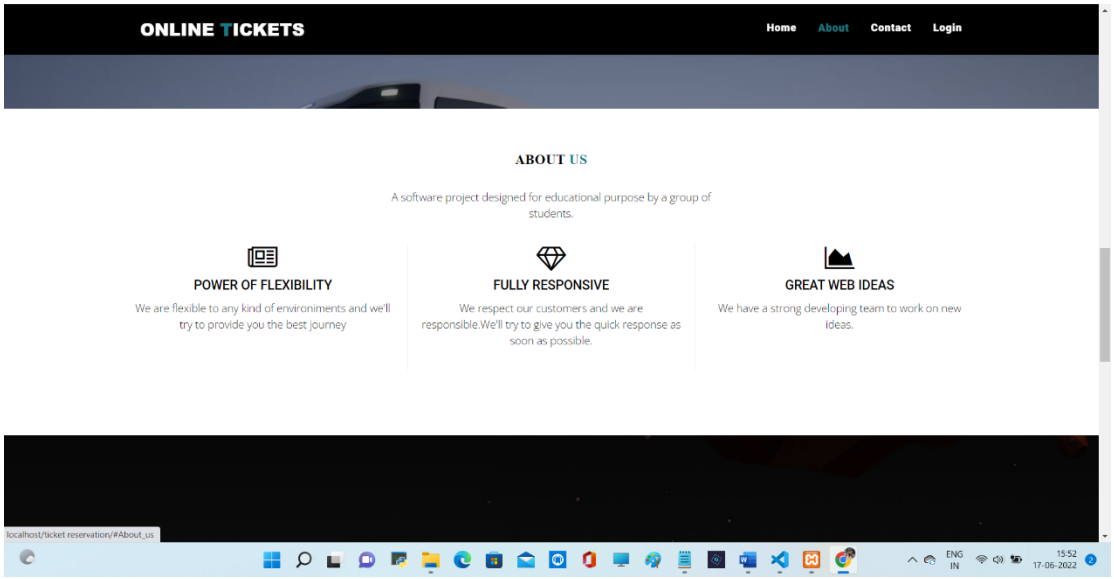
PAYMENTS PAGE



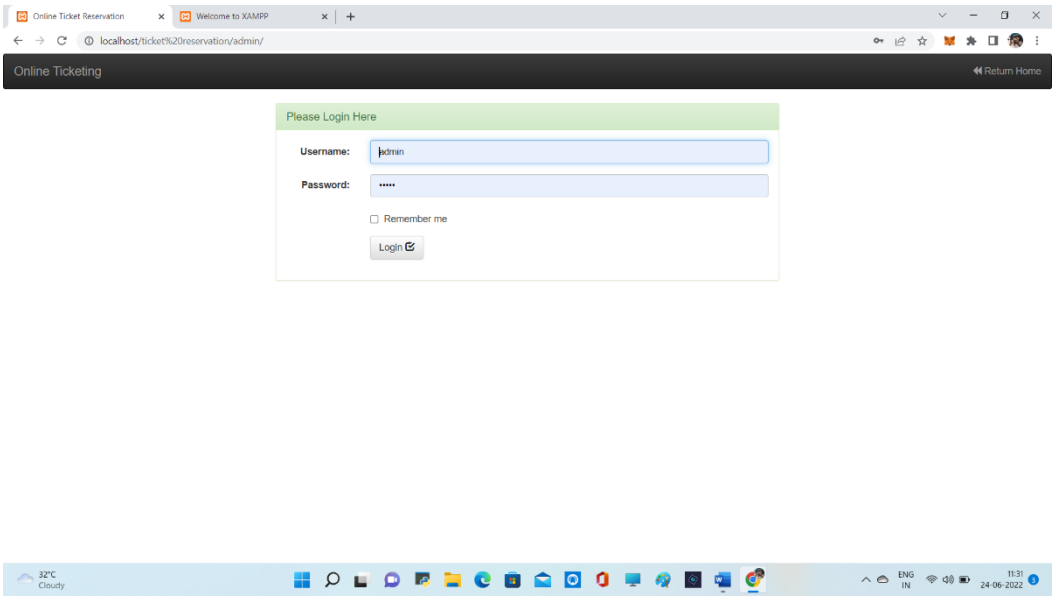
CONTACT PAGE



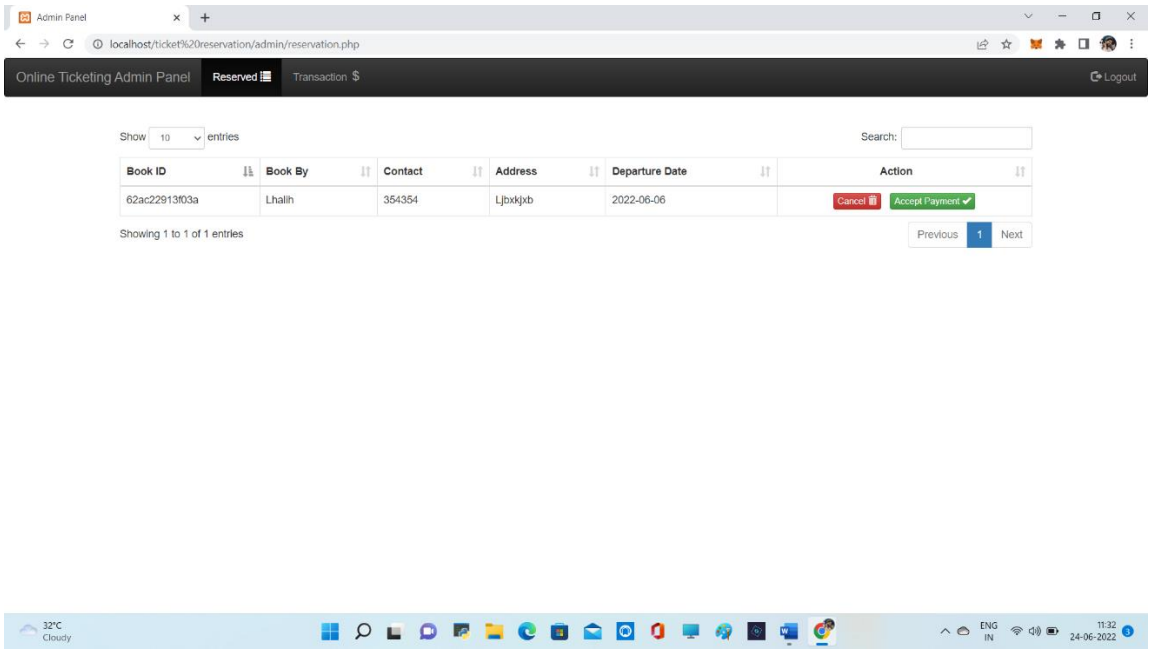
ABOUT US PAGE



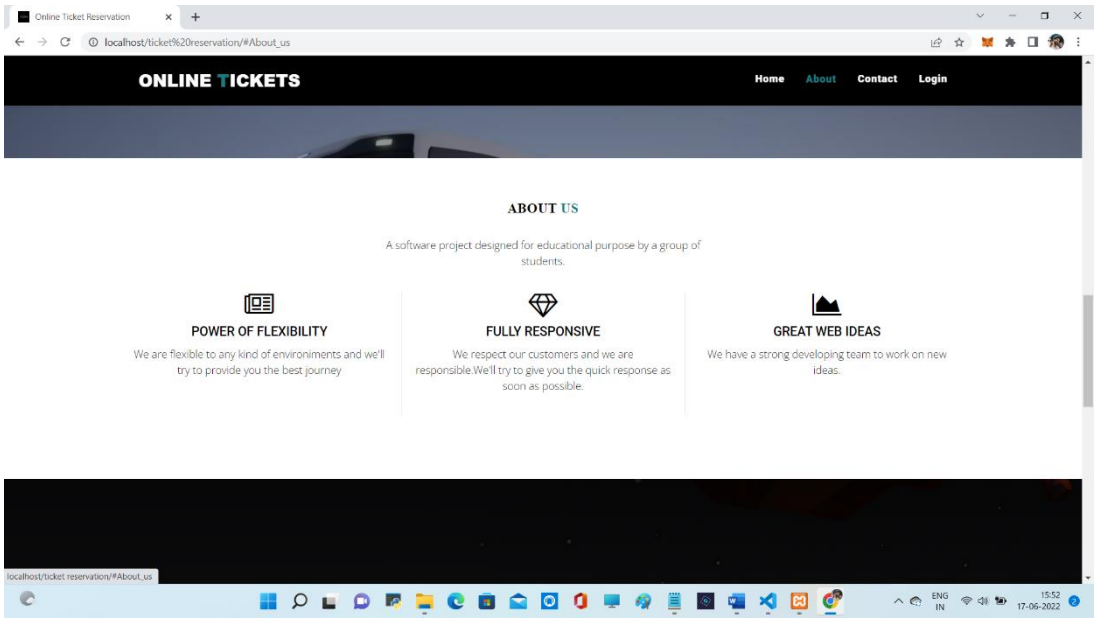
LOGIN PAGE



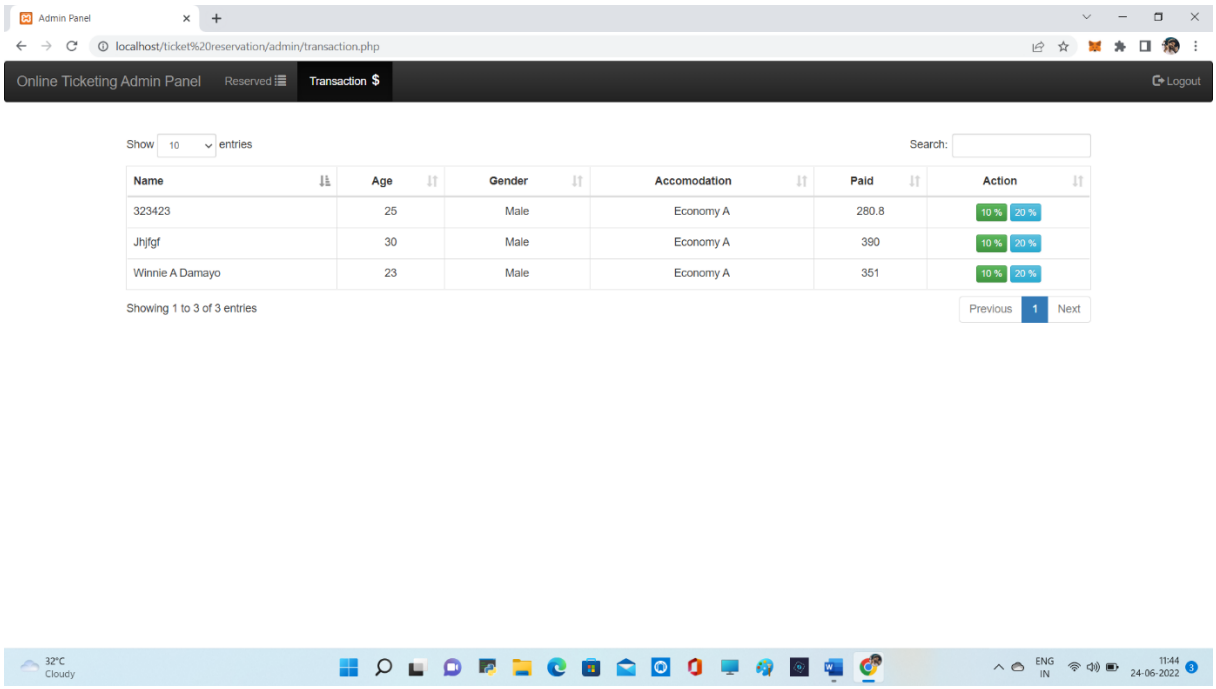
BOOKINGS PAGE



DETAILS PAGE



TRANSACTIONS PAGE



CONCLUSION

Although the overall system will bring some advantages for EZB, some of
the facts needed to be considered. As we said earlier the system is not going to available for 24*7 and there will be a limited operation hour, EZB may lose some customers in future due to the booking time restriction. If the internet connection become down then the system will halt and no offline data can be accessed through it. Changing customers mind wont effect the system as there will be no refund after booking.

It can be observed that computer applications are very important in every field of human endeavour. Here all the information about customer that made reservation can be gotten just by clicking a button with this new system, some of the difficulties encountered with the manual system are overcome. It will also reduce the workload of the staff, reduce the time used for making reservation at the bus terminal and also increase efficiency. The application also has the ability to update records in various files automatically thereby relieving the company's staff the stress of working from file security of data.

This project, as a whole, will give a new way in bus reservations and ticketing processes. The automation and management of seats and reservations will be done online. However, this project does not limit the walkin passengers that is passengers who visit the company's counter because it also caters for them. This also lessens the use of papers like in the traditional way of ticketing.

BIBLIOGRAPHY

BIBLIOGRAPHY

Books Reference

- Design Patterns (1994) 'Erich Gamma, Richard Helmralph, Johnson John and Vlissides'.
- Soft Skills: The Software Developer's Life Manual (2014). 'John Sonmez'.
- Clean Code: A Handbook of Agile Software Craftmanship. 'Robert C. Martin'.

Online Reference

•https://www.codeproject.com/script/Content/Tag.as px?tags=PHP