

BUS TICKET BOOKING SITE

Done by

ANAND KRISHNA S - Reg no:190021089426

GREGEO GEO – Reg no:190021089452

Under the guidance of

Ms. NEENU JOSE

In partial fulfillment of the requirements for the award of the

Degree of

Bachelor of Computer Applications

Of

Mahatma Gandhi University, Kottayam, Kerala



**Department of Computer Science
NIRMALA COLLEGE, MUVATTUPUZHA**
(Affiliated to Mahatma Gandhi University)
2019-2022

NIRMALA COLLEGE

(Affiliated to Mahatma Gandhi University)

MUVATTUPUZHA



CERTIFICATE

Certified that this is a bonafide report on the project and work entitled

.....Bus Ticket Booking Site..... done by.....

Anand Krishna S. George. Gero.....

Reg.No. 190021089426 190021089452 during the year 2021-22.

in the partial fulfilment of the requirements for the award of the degree of Bachelor of Computer Applications of Mahatma Gandhi University, Kottayam, Kerala.

[Signature]
Internal guide



[Signature]
Head of the department

Submitted on the viva-voce held on the.....

[Signature]
External Examiner
21/03/22

ACKNOWLEDGEMENT

We give all honour and praise to the Lord who gave wisdom and enable to complete our project successfully.

We express our sincere and heartful thanks to respected **Dr. Thomas K.V**, our beloved principal for providing necessary facilities for the completion of our project successfully.

We think profusely **Ms. Arya S Nair**, Course Coordinator of Computer Science Department for her guidance and inspiration throughout our course of study.

We express our sincere gratitude to our project guide, **Ms. Neenu Jose** Assistant Professor in Computer Science Department for the valuable advices and guidance throughout the completion of our project.

We also express our gratitude and thanks to all my teachers and friends for their sincere and friendly cooperation in the successful completion of our project.

1. Project Design
2. Project Design
3. Coding Design
4. Database Design
5. User Interface
6. User Testing
7. Database Implementation
8. Quality Testing and Review
9. Maintenance
10. Computer Protection Measures
11. Documentation
12. Budgeting
13. Appraisals
14. Final Report Writing

CONTENTS

1. Introduction	1
1.1 Objective of the project	1
2. System analysis	2
2.1 Existing system	2
2.2 Proposed system	3
2.3 System requirement specification	3
2.3.1 Hardware specification	5
2.3.2 Software specification	5
2.3.3 Front End	7
2.3.4 Back End	8
2.4 Feasibility Analysis	8
2.5 Data Flow Diagram	10
2.5.1 Introduction to Data Flow Diagram	10
3. System Design	15
3.1 Input Design	15
3.2 Output Design	15
3.3 Table Design	16
3.4 process design	16
4. System Testing and Implementation	20
4.1 System Testing	20
4.2 System Implementation	23
5. Security Technology and Policies	25
6. Maintenance	27
7. Scope for Future Enhancement	29
8. Conclusion	30
9. Bibliography	31
10. Appendix	32
10.1 Screen Shots	32

INTRODUCTION

INTRODUCTION

1. INTRODUCTION

The project entitled "Bus ticket booking site" is an offline project for booking bus tickets. The main objective of developing this project is to book the bus tickets. This application will greatly simplify and speed up the booking process. The main aim of the project is to provide the bus tickets to the customers in a simple and accurate way.

This project is useful for customers for reserving their tickets in simple manner. By this system customer can easily get the tickets by entering their destination and their current location.

1.1 OBJECTIVE OF THE PROJECT

The main objective of the project is to provide bus tickets to customers for travelling from one place to another. This project is useful for customers for getting their tickets in simple manner. By online booking reservation one can consume their time and ticket will be get at that moment. The system is intended for public. One can book their tickets by entering the page.

Earlier, people used to avoid bus journey due to poor road infrastructure, lack of luxurious buses, and inefficient bus booking system. Now the Online bus booking system facilitates the online bus bookings and instant payments which is convenient for the booker as well as the facility provider.

There were long queues to book tickets that were time-consuming as well as tiring. Booking tickets via telephone was a difficult task because the telephone line remains either out of service or busy most of the time. But the time has changed. There are plenty of luxurious buses available now, providing best and efficient services to the customers. Online bus booking systems ensure convenience and comfort to guests and a peace of mind. No last minute rush to find a hotel and fear of all rooms being filled up.

SYSTEM ANALYSIS

2.SYSTEM ANALYSIS

System analysis is a step-by-step process used to identify and develop or acquire the software need to control the processing of specific application. System analysis is a continuing activity through the stages of the systems development. System analysis is the process of gathering and interpreting facts, diagnosing problems and using the facts to improve the system. The outputs from the organization are traced through the various processing that the input phases through in the organization. This involves gathering information and using structured tools for analysis. A detailed study of this process must be made by various techniques like interviews; questionnaires etc.

2.1 EXISTING SYSTEM

Customer need to go to the bus station for reserving their tickets. It would take time and availability of the ticket is not sure. And the customer doesn't know about time of arrival and departure. Customer cannot cancel their tickets on their wish .

Drawback of existing system

- Doesn't provide availability of bus
- Might become costly
- Might be slow

2.2 PROPOSED SYSTEM

In the proposed system, there are various controls to provide user friendliness. Details can be accessed over the internet, and huge amount of data, records and information can be stored. It provides high level of security, and there is no risk of data mismanagement. The overall result processing system is easier, flexible and requires less time

ADVANTAGES

- Less time consuming

- Highly cost effective
- Almost zero paper work
- Easy access

2.3 SYSTEM REQUIREMENT SPECIFICATION

A software requirements specification (SRS) is a comprehensive description of the intended purpose and environment for software under development. The SRS fully describes what the software will do and how it will be expected to perform. An SRS minimizes the time and effort required by developers to achieve desired goals and also minimizes the development cost. A good SRS defines how an application will interact with system hardware, other programs and human users in a wide variety of real-worked situations.

Customer requirements

- The system should be fast
- User friendly
- Maintaining security of data
- Efficiency in data retrieval and management

What the developer needs to know?

- Must know the existing system and its drawbacks
- Must know what will be needed in the proposed system

Business Requirements

The system should be feasible both to the developer and client. It should be effective and should be able to complete in time. Developer should be responsible for developing the system, install the software and update the software whenever necessary, conducting any user training that may be needed for using the system.

User Requirements

The user requirement(s) specification is a document usually that specifies the requirements the user expects from software to be constructed in a software project.

- Administrator has overall control in the system.
- Admin can view and edit user details and complaints.
- Faster processing

Functional Requirements

Functional requirements define what a system is supposed to do. The system should perform the following functionalities.

- Login- Login of admin
- Reservation-Customer can book their tickets.
- Bill details-customer can see their booked information.
- View-Admin can view user details.
- Logout- admin can logout from panel

2.3.1 Hardware Specifications

Processor	: Intel Pentium 4 or higher Processor
Speed	: 1.5GHz or higher
System bus	: 32bits
Memory	: 1GB RAM or Higher
Hard disk	: 40GB or Higher
Monitor	: 14" LCD Monitor
Keyboard	: 104 keys
Pointing Device	: Two or Three Button Mouse

2.3.2 Software Specifications

Operating System	: Windows 10
Front End	: PHP, HTML, CSS
Scripting Language	: JavaScript
Back End	: SQL Server Web Server WAMP 2.0
Browser	: Mozilla Firefox

2.3.3 Front end

PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the recursive backronym PHP: HypertextPre-processors.

PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management system and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

CSS

Cascading Style Sheets (CSS) is a style _sheet language used for describing the presentation of a document written in a mark-up language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media.

Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

CSS is designed primarily to enable the separation of document content from document presentation, including aspects such as the layout, colours, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification or presentation characteristics, enable multiple HTML pages to share formatting by specifying on all platforms except Windows. MySQL ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools or install MySQL Workbench via a separate download. Many third party GUI tools are also available.

2.3.4 Back end

MYSQL

MySQL server is powerful database and it requires limited programs and used has back end. It supports GUI and more application is developed by help this server. Collection of tables which holds the data is called database. A beginner can create their own database by click home page. ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools or install MySQL Workbench via a separate download. Many third party GUI tools are also available.

2.4 FEASIBILITY ANALYSIS

A feasibility study is an evaluation and analysis of the potential of the proposed project which is based on extensive investigation and research to give full comfort to the decision makers. Feasibility studies aim to objectively and rationally uncover the strength and weakness of existing business of proposed venture, opportunities and threads as presented by the environment, the resources required to carry through, and ultimately the process for success. In its simplest terms, the two criteria to judge feasibility are cost required and value to attain. As such, a well-designed feasibility study should provide a historical background of the business or project, description of the product or service, accounting statements, details of the operations and management, marketing research and policies, financial data, legal requirements and tax obligations.

The four aspects in the feasibility study are:

- ✓ Technical feasibility
- ✓ Economic feasibility
- ✓ Operational feasibility
- ✓ Behavioural feasibility

✓ Technical Feasibility

The technical feasibility centres on the existing system and what extend it can support the proposed addition. The technical feasibility assessment is focused on gaining an understanding of the present technical resources of the organization and their applicability to the expected needs of the proposed system. The minimum requirements of the system are met by average user. The developer system has a modest technical requirement as only minimal or null changes are required for implementing system.

Normally associated with the technical feasibility includes:

- ❖ Development risk
- ❖ Resource availability
- ❖ Technology

The proposed system can work without any additional hardware or software support other than the computer system and networks. So, I analysed that the proposed system is much more technically feasible than other systems when comparing with the benefits of the new system.

✓ Economic Feasibility

Economic feasibility analysis is also known as cost/benefit analysis. The purpose of the economic feasibility assessment is to determine the positive economic benefits to the organization that the proposed system will provide. The proposed system reduces the operating cost in terms of time by automating the process. This system is economically feasible

✓ **Operational Feasibility**

Operational feasibility is a measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

✓ **Behavioural Feasibility**

People are inherently resistant to changes and computer is known for facilitating the chances. An estimate should be made to how strongly the users react towards the development of the system. The proposed system consumes less time. Thus, the people are made to engage in some other important work.

2.5 DATA FLOW DIAGRAM (DFD)

2.5.1 Introduction to data flow diagram

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. It differs from the flowchart as it shows the data flow instead of the control flow of the program. A data flow diagram can also be used for the visualization of data processing (structured design).

Data flow diagrams were invented by Larry Constantine, the original developer of structured design, based on Martin and Estrin's "data flow graph" model of computation.

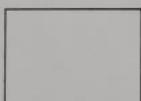
Data flow diagrams (DFDs) are one of the three essential perspectives of Structured System Analysis and Design Method SSADM. The sponsor of a project and the end users will need to be briefed and consulted throughout all stages of a system's evolution. With a data flow diagram, users are able to visualize how the system will operate, what the system will accomplish, and how the system will be implemented. The old system's data flow diagrams can be drawn up and compared with the new system's data flow diagrams to draw comparisons to implement a more efficient system. Data flow diagrams can be used to provide the end user with physical idea of where the data they input ultimately has an effect upon the structure of

the whole system from order to dispatch to report. How any system is developed can be determined through a data flow diagram.

Developing a data flow diagram helps in identifying the transaction data in the data model. There are different notations to draw data flow diagrams, defining different visual representation for process, data stores, data flow, and external entities. The first step is to draw a data flow diagram (DFD). A DFD also known as "bubble chart" has the purpose of clarifying system requirements and identifying major transformation that will become program in system design. So, it is starting point of the design phase that functionally decompose the requirements specification down to the lowest level of details DFD consists of series of bubbles joined by lines. The bubbles represent data transformation and the lines represent data flow in the system.

DFD Symbols

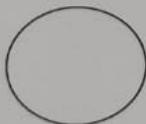
- Square- Defines source or destination of system.



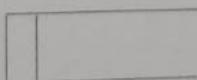
- Data flow - Identifies data flow Circle



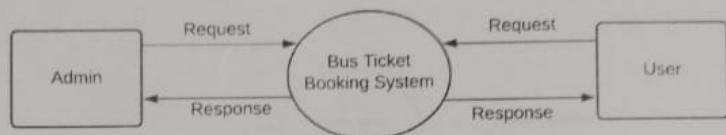
- Bubble - Represents a process that transforms incoming data to outgoing data.

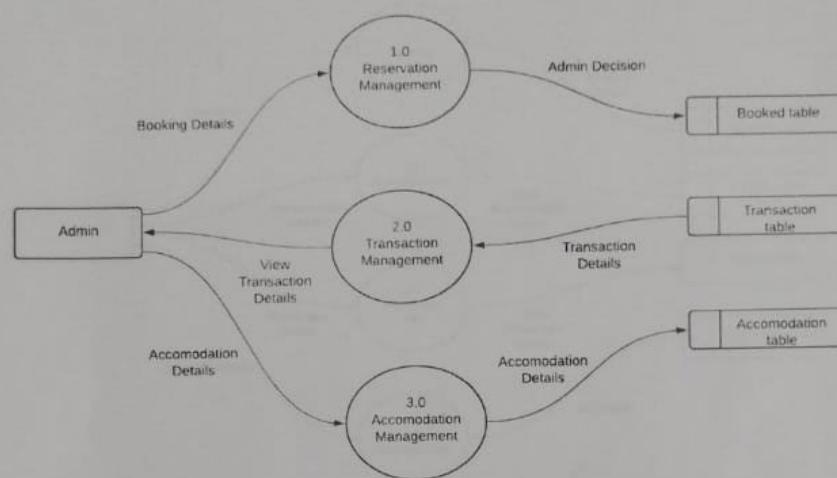


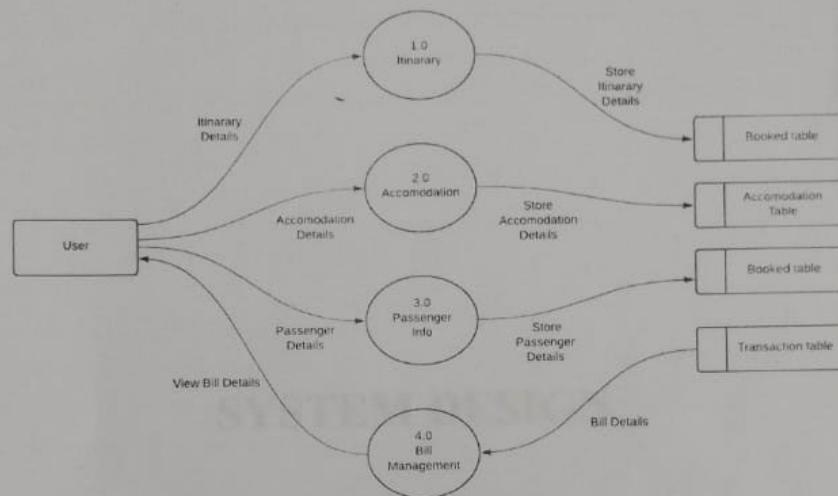
- Open rectangle- Data store



LEVEL 0



LEVEL 1 - Admin

Level 1 - User

1. SYSTEM DESIGN

The system design stage is concerned with the overall architecture of the system. This includes the choice of hardware and software components, their interconnection, and the overall system performance. The main objective of system design is to ensure that the system meets the requirements of the user.

System design involves several steps, starting from the analysis of the requirements, followed by the selection of appropriate hardware and software components, and finally the integration and testing of the system.

SYSTEM DESIGN

The system design stage is concerned with the overall architecture of the system. This includes the choice of hardware and software components, their interconnection, and the overall system performance. The main objective of system design is to ensure that the system meets the requirements of the user.

2. INPUT DESIGN

The input design stage is concerned with the design of the input devices used to enter data into the system. These devices include keyboards, mice, touchscreens, and other sensors. The goal of input design is to ensure that the input devices are easy to use, reliable, and accurate. In addition, the input design stage also considers the user interface, which is the way the user interacts with the system. The user interface should be intuitive, easy to learn, and provide feedback to the user.

The input design stage is concerned with the design of the input devices used to enter data into the system. These devices include keyboards, mice, touchscreens, and other sensors. The goal of input design is to ensure that the input devices are easy to use, reliable, and accurate. In addition, the input design stage also considers the user interface, which is the way the user interacts with the system. The user interface should be intuitive, easy to learn, and provide feedback to the user.

3. SYSTEM DESIGN

3.1 INPUT DESIGN

The quality of the system input determines the quality of the system output. Input specification describes the manner in which data enter the system for processing. Input design features can ensure the reliability of the system and produce result from accurate data, or they can result in the production of erroneous information. The input design also determines whether the user can interact efficiently with the system.

In our system almost all inputs are being taken from the databases. To provide adequate inputs we have to select necessary values from the databases and arrange it to the appropriate controls.

Admin

Admin is the one who controls the whole system. The administrator is the super user of this application. Only admin have access into this admin page. Admin can access the page using their login id and password. Admin can create and delete elections. Admin can review candidate applications and remove users.

User

The user of the system is the students. Users can cast their vote for their respective candidates. Submit application to become a candidate.

3.2 OUTPUT DESIGN

One of the important features of an information system for users is the output produced. Output is the information delivered to users through the information system. Without quality of the output, the entire system appears to be unnecessary that users will avoid using it. Users generally merit the system solely by its output. In order to create the most useful output possible. One works closely with the user through an interactive process, until the result is considered to be satisfactory.

Admin

Admin can view all the reservation details, transaction details and accomodation details.

User

User can choose any destination and date, then select the corresponding accomodation and can view the result of the reservation.

3.3 TABLE DESIGN

The data design transforms the information domain model created during analysis into the data structures that will be required to implement the software. The data objects and relationships defined in the entity relationship diagram and the detailed data content depicted in the data dictionary provide the basis for the data design activity.

The overall objective in the development of database technology has been to treat data as an organizational resource and as an integrated whole. Database Management System allows data to be protected and organized separately from other resources. Database is a integrated collection of data. This is the difference between logical and physical data.

The organization of data in the database aims to achieve three major objectives:

- Data integration
- Data integrity
- Data independence

The databases are implemented using a DBMS package. Each particular DBMS has unique characteristics and general techniques for database design. There are 6 major steps in design process. The first 5 steps are usually done on paper and finally the design is implemented.

- Identify the table and relationships
- Identify the data that is needed for each table and relationship.
- Resolve the relationship
- Verify the design
- Implement the design

The database uses tables for storage. A table also contains records, which is a set of fields. All records, in a table have the same set of fields with different information. Uses 7 tables.

Each table contains key fields that establish relationships in the database and how the records are stored. There are primary key fields that uniquely identify a record in a table. There are also fields that contain the primary key from another table called foreign keys.

The various database tables that are used in this project are the following:

1) Tablename: accomadation

Description: Store accomadation information

Primary Key: acc_id

Field Name	Datatype	Size	Description
acc_id	Int		Accomadation Id
acc_type	Varchar	30	Type of accomadation
acc_price	Int		Price of accomadation
acc_slot	Int		Available slot

2) Tablename: origin

Description: Store the details of origin

PrimaryKey: origin_id

Field Name	Datatype	Size	Description
Origin_id	Int		Origin's Id
Origin_desc	varchar	10	Name of origin

3)

Tablename: destination

Description: Stores the details of destination

PrimaryKey: dest_id

Field Name	Datatype	Size	Description
Dest_Id	Int		Destination id
Dest_desc	varchar	30	Name of Destination

4)

Tablename:booked

Description:Store the details of tickets booked

PrimaryKey: book_id

ForeignKey: acc_id,dest_id,origin_id

Field Name	Datatype	Size	Description
Book_Id	Int		Booking id
Book_by	varchar	30	name of person who booked
Book_contact	varchar	30	Contact of person who booked
Book_address	varchar	30	Address of person who booked
Book_name	varchar	20	Passenger name
Book_age	int		Passenger age
Book_gender	varchar	15	Passenger gender
Book_departure	date		Date of departure
Dest_id	int		Destination id
Acc_id	int		Accomadation id
Origin_id	int		Origin id

5)

Tablename :transaction

Description: Stores the information of transaction

PrimaryKey: trans_id

ForeignKey:acc_id,dest_id,origin_id,stat_id

Field Name	Datatype	Size	Description
Trans_Id	Int		Transaction id
Trans_time	varchar	30	Transaction time
Trans_payment	Int		Transaction payment
Trans_passenger	varchar	30	Details of passenger
Trans_age	Int	10	Age of passenger
Trans_gender	Varchar	30	Gender of passenger
Acc_id	int		Accomadation id
Origin_id	int		Origin id
Dest_id	int		Destination id
Stat_id	int		Status id
Trans_refunded	int		Transaction refund

6)

Tablename:status

Description: Stores the details of the payment status

PrimaryKey: status_id

Field Name	Datatype	Size	Description
Status_Id	Int		Status ID
Status_desc	Varchar	30	Details of status

7)

Tablename: admin

Description: Stores the information of user

Primary Key: admin_id

Field Name	Datatype	Size	Description
Admin_id	Int		Admin ID
Admin_account	varchar	30	Admin account
Admin_password	varchar	30	Admin password

**SYSTEM TESTING
&
IMPLEMENTATION**

4. SYSTEM TESTING AND IMPLEMENTATION

4.1 SYSTEM TESTING

Testing is the process of examining the software to compare the actual behaviour with that of the expected behaviour. The major goal of software testing is to demonstrate that faults are not present. In order to achieve this goal, the tester executes the program with the intent of finding errors. Though testing cannot show absence of errors but by not showing their presence it is considered that these are not present.

System testing is defined as the process by which one detects the defects in the software. Any software development organization or team has to perform several processes. Software testing is one among them. It is the final opportunity of any programmer to detect and rectify any defects that may have appeared during the software development stage. Testing is a process of testing a program with the explicit intention of finding errors that makes the program fail. In short system testing and quality assurance is a review in software products and related documentation for completion, correctness, reliability and maintainability.

System testing is the first stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct and the goal will be successfully achieved. A series of testing are performed for the proposed system before the proposed system is ready for user acceptancetesting.

The testing steps are,

- Unit testing
- Integration testing
- Validation
- Output testing
- Acceptance testing

System Testing provides the file assurance that software once validated must combined with all other system elements. System testing verifies whether all elements have been combined properly and that overall system function and performance is achieved. At the integration of modules, the validation test was carried out over the system. It was that all the modules work well together and meet the overall system function and performance.

a. Unit Testing

Unit testing is carried out screen-wise, each screen being identified as an object. Attention is diverted to individual modules, independently to one another to locate errors. This has enabled the detection of errors in coding and logic.

Various test cases are prepared. For each module these test cases are implemented and it is checked whether the module is executed as per the requirements and outputs the desired result. In this test each service input and output parameters are checked.

In unit testing

- Module interface was tested to ensure that information properly flows into and out of the program under test.
- Boundary condition was tested to ensure that module operates properly at boundaries established to limit or restrict processing.
- All independent paths through the control structures were executed to ensure that all statements in the modules have been executed at least once.
- Error handling paths were also tested.

b. Integration Testing

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing.

Unit tested module were taken and a single program structure was built that has been dictated by the design. Incremental integration has been adopted here.

The modules are tested separately for accuracy and modules are integrated together using bottom up integration i.e., by integrating from moving from bottom to the top the system is checked and errors found during integration are rectified. In this testing individual modules were combined and the module wise shifting was verified to be alright.

The entire software was developed and tested in small segments, where errors were easy to locate and rectify. Program builds (group of modules) were constructed corresponding to the successful testing of user interaction, data manipulation analysis, and display processing and database management.

c. Validation Testing

Validation testing is done to ensure complete assembly of the error-free software. Validation can be termed successful only if it functions in manner. Reasonably expected by the student under validation is alpha and beta testing. The student-side validation is done in this testing phase. It is checked whether the data passed to each student is valid or not. Entering incorrect values does the validation testing and it is checked whether the errors are being considered. Incorrect values are to be discarded. The errors are rectified.

In "Bus Ticket Booking System" verifications are done correctly. So, there is no chance for users to enter incorrect values. It will give error messages by using different validations. The validation testing is done very clearly and found it is error free.

d. Output Testing

After performing the validation testing the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in a specific format.

The output format on the screen was found to be correct as the format was designed in the system design phase according to the user needs. For the hard copy also, the output comes out as specified requirement by the user. Hence output testing does not result in any Correction in the system output .This project is developed based on the user choice. It is user friendly. The output format is very clear to user. Output testing is done on Smart builders correctly.

e. Acceptance testing

Acceptance involves running a suite of tests on the completed system. Each individual test, known as a Case, exercise particular operating condition of the operating condition of the user's environment or feature of the system, and will result in a pass fail, or Boolean outcome.

4.2 SYSTEM IMPLEMENTATION

The implementation is the final state and it is an important phase. It involves the invalid programming system testing, user training and the operational running of developed proposed system that constitutes the application subsystems. A major task of preparing for implementation is education of users, which should really have been taken place much earlier in the project when they were involved in the investigation and design work. During the implementation phase system actually take physical shape. In order to develop a system implemented planning is very essential.

The implementation phase of the software development is concerned with translating design specification into source code. The user tests the developed system and changes are made according to their needs. Our system has been successfully implemented.

Before implementation several tests have been conducted to ensure that no errors are encountered during the operation. The implementation phase ends with an evaluation of the system after placing into the operation for a period of time.

The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from old system to new system. The system can be implemented only after testing is done and is found to be working to specifications. The implementation stage is a systems project in its own right.

The implementation stage involves following tasks:

- Careful planning.
- Investigation of system and constraints.
- Design of method to achieve change over
- Evaluation of the changeover method.

In the case of this project all the screens are designed first. For making it to be executable, codes are written on each screen and performs the implementation by creating the database and connecting to the server. After that the system is Checked, whether it performs all the transactions Correctly. Then databases are cleared and made it to be usable to the technicians.

Implementation Plans

The following are the step involved in the implementation plan of "Smart Builders":

- ✓ Test system with sample data
- ✓ Detection and correction of errors
- ✓ Make the necessary changes in the system
- ✓ Check the existing system
- ✓ Installation of hardware and software utilities
- ✓ Training and involvement of user personals

**SECURITY TECHNOLOGIES
&
POLICIES**

5. SECURITY TECHNOLOGIES & POLICIES

The protection of computer-based resources that includes hardware, software, data procedures and people against unauthorized use or natural.

Disaster is known as System Security.

System Security can be divided into four related issues:

- Security
- Integrity
- Privacy
- Confidentiality

SYSTEM SECURITY refers to the technical innovations and procedures applied to the hardware and operation systems to protect against deliberate or accidental damage from a defined threat.

DATA SECURITY is the protection of data from loss, disclosure, modification and destruction.

SYSTEM INTEGRITY refers to the proper functioning of hardware and programs, appropriate physical security and safety against external threats such as eavesdropping and wiretapping.

PRIVACY defines the rights of the user or organizations to determine what information they are willing to share with or accept from others and how the organization can be protected against unwelcome, unfair or excessive dissemination of information about it.

CONFIDENTIALITY is a special status given to sensitive information in a database to minimize the possible invasion of privacy. It is an attribute of information that characterizes its need for protection.

SECURITY IN SOFTWARE System security refers to various validations on data in form of checks and controls to avoid the system from failing. It is always important to ensure that only valid data is entered and only valid operations are performed on the system.

The system employs two types of checks and controls:

CLIENT-SIDE VALIDATION Various client-side validations are used to ensure on the client side that only valid data is entered. Client-side validation saves server time and load to handle invalid data. Some checks imposed are:

- ❖ Forms cannot be submitted without filling up the mandatory data so that manual mistakes of submitting empty fields that are mandatory can be sorted out at the client side to save the server time and load.
- ❖ Tab-indexes are set according to the need and taking into account the ease of user while working with the system.

SERVER-SIDE VALIDATION Some checks cannot be applied at client side. Server-side checks are necessary to save the system from failing and intimating the user that some invalid operation has been performed or the performed operation is restricted. Some of the server-side checks imposed is:

- ❖ Server-side constraint has been imposed to check for the validity of primary key and foreign key. A primary key value cannot be duplicated. Any attempt to duplicate the primary value results into a message intimating the user about those values through the forms using foreign key can be updated only of the existing foreign key values.
- ❖ User is intimating through appropriate messages about the successful operations or exceptions occurring at server side.
- ❖ Various Access Control Mechanisms have been built so that one user may not agitate upon another. Access permissions to various types of users are controlled according to the organizational structure. Only permitted users can log on to the system and can have access according to their category. User- name, passwords and permissions are controlled o the server side.
- ❖ Using server-side validation, constraints on several restricted operations are imposed.

MAINTENANCE

6. MAINTENANCE

Software maintenance is the modification of a software product after delivery to correct faults, to improve performance or other attributes. Maintenance is the ease with which a program can be corrected if any error is encountered, adapted if its environment changes or enhanced if the customer desires a change in requirement. Maintenance follows conversation to extend that changes are necessary to maintain satisfactory operations relative to changes in the user's environment.

Maintenance often includes minor enhancements or corrections to problems that surface in the system's operation. Maintenance is also done based on fixing the problems reported, changing the interface with other software or hardware enhancing the software.

CATEGORIES OF MAINTENANCE

a. Corrective Maintenance

Corrective maintenance is the most commonly used maintenance approach, but it is easy to see its limitations. When equipment fails, it often leads to downtime in production, and sometimes damages other parts. In most cases, this is expensive. Also, if the equipment needs to be replaced, the cost of replacing it alone can be substantial. Reliability of systems maintained by this type of maintenance is unknown and cannot be measured. Corrective maintenance is possible since the consequences of failure or wearing out are not significant and the cost of this maintenance is not great.

b. Perfective Maintenance

Modification of a software product after delivery to improve performance or maintainability. This term is used to describe changes undertaken to expand the existing requirements of the system. A successful piece of software tends to be subjected to a the Succession of changes resulting in an increase in its requirements. This is based on premise that as the software becomes useful, the user experiment with new cases beyond the scope for which it was initially developed. Vxpansi01 n requirements can take the form enhancement of existing system functionality and improvement in computational efficiency.

c. Adaptive Maintenance

Modification of a software product performed after delivery to keep a are product usable in a changed or changing environment. Adaptive maintenance includes any work initiated as a consequence of moving the software to a different hardware or software platform. It is a change driven by the need to accommodate modifications in the enviroment of software system. The environment in this context refers to the totality of all conditions and influences which act from outside upon the system. A change to the whole or part of this environment will Warrant a corresponding modification of the software.

d. Preventive Maintenance

Preventive maintenance is a schedule of planned maintenance actions aimed at the prevention of breakdowns and failures. The primary goal of preventive maintenance is to prevent the failure of equipment before it actually occurs. It is designed to preserve and enhance equipment reliability by replacing worn components before they actually fail. Preventive maintenance activities include equipment checks, partial or complete overhauls at specified periods

Long-term benefits of preventive maintenance include:

- ❖ Improved system reliability.
- ❖ Decreased cost of replacement.
- ❖ Decreased system downtime.

**SCOPE FOR FUTURE
ENHANCEMENT**

7. SCOPE FOR FUTURE ENHANCEMENT

The drawbacks of the existing system as listed before are fully evacuated. All the existing inconsistencies are fully solved as this system is implemented.

- The manual method of complaint registration' ,complaint processing was found to be tedious, especially when carried out for a large number of complaints, this makes the entire process time-consuming and error prone.
- In the existing system the can have a link for bill payments such as water bill and electricity bill.

CONCLUSION

CONCLUSION

8. CONCLUSION

The proposed Bus Ticket Booking System Project is a very effective plus efficient GUI-based component. This software is well tested; it works properly to meet the user requirements as described in the project. Currently the system is web-based giving all the required user result details; various future enhancements such as bill payment, applications for new connection, etc. can be incorporated into the system.

BIBLIOGRAPHY

BIBLIOGRAPHY

9. BIBLIOGRAPHY

1. Jalote, Pankaj. "**SOFTWARE ENGINEERING**", Third Edition, Spring Publishers, Narosa Publishing House.
2. Rob, Peter. and Coronel, Carlos. "**DATABASE SYSTEMS DESIGN AND MANAGEMENT**", Fifth Edition, Thomson Learning Publication.
3. Pressman, Roger S. "**SOFTWARE ENGINEERING**", Fifth Edition, McGraw-Hill Publication.
4. Bayross, Ivan. "**SQL/MySQL**" Second Revised Edition, BPB Publications

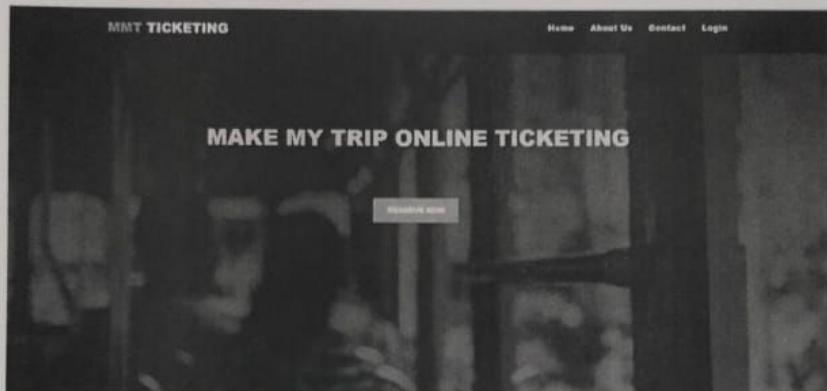
APPENDIX

APPENDIX

10. APPENDIX

10.1 SCREEN SHOTS

Home Page



Itinerary Details

A screenshot of the "Itinerary Details" page. The top navigation bar shows "MMT Online Ticketing" and "Reservable". The main area is titled "STEPS FOR BOOKING" and lists four steps: 1. ITINERARY (highlighted), 2. ACCOMMODATION, 3. PASSENGER INFO, and 4. PAYMENT/BOOKING. Step 1 has sub-options: "SEARCH FLIGHT/ TRAIN", "ACCOMMODATION TYPE", "PASSENGER DETAILS", and "TOTAL PAYMENT". A modal window titled "ITINERARY" is open, showing fields for "Origin" (Kozhikode), "Destination" (Bangalore), and "Departure Date" (08-Nov-1999). A "SEARCH" button is at the bottom of the modal.

Accommodation Details

The screenshot shows a search interface for bus tickets. At the top, there are fields for 'From' (Muvattupuzha), 'To' (Kochi), 'Date' (20/08/2017), and 'Time' (10:00 AM). Below this is a table titled 'ACCOMMODATION' listing various seating options with their prices:

Accommodation	Price
Normal	Rs. 120
Economy A	Rs. 140
Economy B	Rs. 160
Super	Rs. 180
Carpet	Rs. 200
Deluxe	Rs. 220

Below the table is a 'Total of Passenger' input field and a 'NEXT →' button.

Passenger Details

The screenshot shows a form for entering passenger information. It includes fields for 'Name' (Suresh), 'Contact' (9846543210), 'Address' (Kerala, India), and a 'Booked(E1)' section. The 'Booked(E1)' section contains fields for 'Full Name (1)', 'Age (1)', and 'Gender (1)'. The 'Full Name (1)' field has 'Suresh' typed into it. The 'Age (1)' field has '21' typed into it. The 'Gender (1)' dropdown is set to 'Male'. There is also a 'NEXT →' button at the bottom.

Payment Details

PAGE FIFTEEN

PAYMENT INFO

Departure

Kochi - Bangalore

Departure Date: 2021-12-21 08:00AM

Business Class, One Way, The First Day, ₹ 1,000/-

₹ 1,000/-

Contact Info

Book By: 2021-12-20 23:59PM

Contact No: +91 9876543210

Address: Unnithan Road

1 PASSENGERS - Booked ID: E1eC798fb4f4

Name	Age	Gender	Departure Price
John	21	Male	₹ 500
TOTAL			₹ 1,000

BOOK NOW

Registration Details**Admin Login**

NUIT Online Booking

Please Log in Here.

Username:

Password:

Remember Me

Login

Forgot Password

Reservation Management

MMT Ticketing Admin - Reservations						
Search	Book ID	Book By	Contact	Address	Return Date	Action
MMT-1234567890	MMT-1234	Parthasarathy	Sunaina-123-Kerala	2023-12-31		
Showing 1 to 10 of 20 entries						

Transaction Details

MMT Ticketing Admin - Reservations							
Name	Age	Gender	Accommodation	Paid	Balance	Action	Search
Ajith	20	Male	Economy A	300			
Ajith	20	Male	Gold	340			
Aswath	20	Male	Economy B	300			
Aswath	20	Male	Delux	300			
Aswath	20	Male	Single	300			
Aswath	20	Male	Delux	300			
Aswath	20	Male	Economy B	300			
Aswath	20	Male	Single	300			
Aswath	20	Male	Single	300			
Aswath	20	Male	Single	300			
Showing 1 to 10 of 20 entries							

Accomodation Management

ACCOMODATION

Accommodation	Seats	Fare
Single	10	500
Economy A	20	300
Economy B	20	200
Normal	20	400
Super	20	600
Delux	20	700

EDIT DETAILS

Select Accommodation:

* Seats: Fare:

Update:

[Home](#) [Logout](#)