# **ABSTRACT**

This project is based on TRAVEL AND TOURISM MANAGEMENT. The Tourism Management System is a comprehensive software solution designed to streamline and enhance various aspects of the tourism industry. The project is totally built at administrative end and thus only the administrator is guaranteed the access to the backend database.

The purpose of this project is to build an application program to reduce the manual work for managing Tourists, Booking, Places etc. Administrator can access and modify the information stored in the database of this system, this includes adding and updating of details, and it will give accurate information and simplifies manual work and also it minimizes the documentation related work. Provides up to date information. Finally booking confirmation notification will be send to the users. Tourists can register by providing personal details, make new reservation and book only one hotel and package.

The system offers a range of features to cater to the diverse needs of the tourism sector. For tourists, it includes an interactive portal where they can explore destinations, view available travel packages, make bookings, and access essential information such as local attractions, accommodations, transportation options, and safety guidelines. The platform provides a personalized experience, recommending travel itineraries based on user preferences and previous travel history.

On the service provider side, the Tourism Management System offers an integrated dashboard for managing bookings, tracking occupancy rates, and coordinating with other tourism-related businesses. Tourism operators and accommodation providers can easily update their offerings, manage availability, and communicate with customers through the platform.

**INDEX**

[**CHAPTER 1: INTRODUCTION** 3](#_Toc160581955)

[**1.1** **Background** 3](#_Toc160581956)

[**1.2** **Objective** 4](#_Toc160581957)

[**1.3 PURPOSE** 5](#_Toc160581958)

[**1.4 SCOPE** 5](#_Toc160581959)

[**1.5 APPLICABLITY** 6](#_Toc160581960)

[**1.6 Achievements** 6](#_Toc160581961)

[**1.7 Organization Report** 7](#_Toc160581962)

[**CHAPTER 2: SURVEY OF TECHNOLOGIES** 8](#_Toc160581963)

[**2.1 FRONT-END TECHNOLOGY** 8](#_Toc160581964)

[**2.2 IDE (Integrated Development Environment):** 8](#_Toc160581965)

[**2.3 BACK-END TECHNOLOGY** 9](#_Toc160581966)

[**CHAPTER 3: REQUIRMENT AND ANALYSIS** 10](#_Toc160581967)

[**3.1 Project Definition** 10](#_Toc160581968)

[**3.2** **Requirements Specification** 10](#_Toc160581969)

[**3.2.1 Functional Requirements:** 10](#_Toc160581970)

[**3.2.2 Non-Functional Requirements:** 11](#_Toc160581971)

[**3.3** **Planning And Scheduling** 11](#_Toc160581972)

[**3.4 Software and Hardware Requirements** 13](#_Toc160581973)

[**3.4.1 Hardware Specification:** 13](#_Toc160581974)

[**3.4.2 Software Specification:** 13](#_Toc160581975)

[**3.5 Preliminary Product Description** 14](#_Toc160581976)

[**3.6** **Modules Description** 15](#_Toc160581977)

[**3.7** **Feasibility Study** 15](#_Toc160581978)

[**CHAPTER 4: SYSTEM DESIGN** 17](#_Toc160581979)

[**4.1** **UML Diagrams** 17](#_Toc160581980)

[**4.1.1 USE CASE Diagram** 17](#_Toc160581981)

[**4.1.2 ER Diagram** 19](#_Toc160581982)

[**4.1.3 DFD Diagram** 20](#_Toc160581983)

[**4.1.4 Activity Diagram** 21](#_Toc160581984)

[**4.1.5 Sequence diagram** 22](#_Toc160581985)

[**4.2** **DATA DESIGN** 23](#_Toc160581986)

[**4.2.1 SCHEMA DESIGN** 23](#_Toc160581987)

[**4.2.2 DATA INTEGRITY AND CONSTRAINTS** 24](#_Toc160581988)

[**4.2.3 ALGORITHM DESIGN** 26](#_Toc160581989)

[**4.3 USER INTERFACE DESIGN:** 27](#_Toc160581990)

[**TEST CASE DESIGN:** 27](#_Toc160581991)

[**CHAPTER 5: IMPLEMENTATION AND TESTING** 30](#_Toc160581992)

[**5.1 IMPLEMENTATION AND APPROCHES** 30](#_Toc160581993)

[**5.2** **CODING DETAILS AND EFFICIENCY** 30](#_Toc160581994)

[**5.2.1 CODE EFFICIENCY:** 30](#_Toc160581995)

[**5.3** **TESTING APPROACH** 32](#_Toc160581996)

[**5.3.1 UNIT TESTING** 32](#_Toc160581997)

[**5.3.2 SYSTEM TESTING** 36](#_Toc160581998)

[**5.3.3** **BETA TESTING** 39](#_Toc160581999)

[**CHAPTER 6: RESULT AND DISCUSSION** 41](#_Toc160582000)

[**6.1** **TEST REPORTS** 41](#_Toc160582001)

[**6.2** **USER DOCUMENTATION:** 42](#_Toc160582002)

[ **Login** 42](#_Toc160582003)

[ **Signup** 43](#_Toc160582004)

[ **Add Personal Details** 44](#_Toc160582005)

[ **Update Customer Details** 45](#_Toc160582006)

[ **Check Packages** 46](#_Toc160582007)

[ **Book Packages** 46](#_Toc160582008)

[ **Book Hotel** 47](#_Toc160582009)

[ **Payments** 48](#_Toc160582010)

[**CHAPTER 7: CONCLUSION** 49](#_Toc160582011)

[**7.1** **Conclusion** 49](#_Toc160582012)

[**7.2** **FUTURE SCOPE OF THE PROJECT** 49](#_Toc160582013)

[**Reference** 50](#_Toc160582014)

# **CHAPTER 1: INTRODUCTION**

# **Background**

The Tourism Management System is an advanced dekstop-based solution aimed at revolutionizing the tourism industry. With the increasing demands of modern travelers and the growth of the sector, there is a need for a comprehensive platform to efficiently manage tourism operations. This project seeks to develop a user-friendly application that connects tourists and service providers, streamlining interactions for a more enjoyable travel experience.

The Tourism Management System offers a one-stop platform for tourists to explore destinations, access attraction information, and book travel packages, accommodations, and transportation seamlessly. It also provides personalized travel itineraries based on individual preferences and feedback. For service providers, the system offers a centralized dashboard for managing bookings, optimizing services, and coordinating with other businesses.

The Tourism Management System aims to create a future where travel becomes more accessible, enjoyable, and sustainable for all involved. Through detailed design, development, and continuous feedback, the system seeks to revolutionize the tourism industry and benefit both tourists and service providers alike.

The global tourism industry has long been a cornerstone of economic growth and cultural exchange, captivating the imaginations of millions of travelers worldwide. In recent years, this industry has witnessed exponential growth, thanks in part to increased connectivity, rising disposable incomes, and a growing appetite for diverse travel experiences.

By embracing cutting-edge technology and user-centric design principles, the Tourism Management System aspires to usher in a new era in the tourism industry, one that is marked by accessibility, sustainability, and enhanced user experiences.

# **Objective**

**Objective of project is**

The objective of the Tourism Management System project is to develop and implement an innovative dekstop-based software solution that addresses the challenges faced by the tourism industry and enhances the overall travel experience for tourists and service providers. By achieving these objectives, the Tourism Management System aims to contribute to the sustainable growth of the tourism industry, foster positive experiences for travelers, and promote the economic and cultural benefits that tourism brings to destinations around the world.

* One of the primary objectives of the Tourism Management System is to streamline the operations within the tourism industry. This entails simplifying the processes involved in booking accommodations, tours, transportation, and other travel-related services.
* Sustainable tourism is a crucial aspect of the modern travel industry. The system intends to contribute to responsible tourism by providing information about eco-friendly attractions, responsible travel options, and environmental guidelines.
* The project recognizes the importance of tourism in local communities and aims to ensure that it has a positive impact.
* Data security and privacy are paramount in any system handling personal and financial information. The objective is to implement robust security measures to protect user data, ensuring confidentiality and privacy throughout the system.
* The project doesn't end with the initial development. It envisions continuous improvement through user feedback, data analytics, and regular updates.
* Convenience and Accessibility

# **1.3 PURPOSE**

The purpose of the Tourism Management System project is to modernize and streamline the operations of the tourism industry by leveraging technology and innovative solutions. This comprehensive Dekstop-based platform is designed to benefit both tourists and tourism service providers, addressing the challenges and complexities faced by the industry.

Responsible and sustainable tourism is a critical objective of the project. It provides information about eco-friendly attractions, responsible travel options, and environmental guidelines to educate tourists and encourage them to make sustainable choices. It focuses on providing a user-friendly interface that caters to the diverse needs of tourists and service providers,

# **1.4 SCOPE**

* The project will include the development of a user-focused, dekstop-based platform accessible to tourists and tourism service providers.
* The system will provide tourists with features such as destination exploration, attraction information, travel package booking, accommodation reservations, and real-time notifications.
* Service providers will have access to a dashboard for managing bookings, updating offerings, and viewing analytics.
* The system will provide information about sustainable tourism practices, eco-friendly attractions, and responsible travel options.
* System administrators will have access to an administrative dashboard for managing user accounts and overseeing system performance.
* Collaboration with stakeholders, including tourism agencies and service providers, is essential for project success.

# **1.5 APPLICABLITY**

* Tourists of all types, including leisure travelers, business travelers, and adventure seekers, can use the system to plan and manage their trips. It caters to a wide range of travel preferences and budgets.
* Marketing agencies specializing in tourism can leverage the system to create targeted marketing campaigns and promote tourism destinations.
* Institutions that provide training and education in tourism and hospitality can use the system as a practical tool for teaching students about the industry and its evolving technologies.
* Organizations that promote eco-friendly and responsible tourism can use the system to showcase sustainable travel options and educate tourists about minimizing their environmental impact.

## **1.6 Achievements**

* Travel and Tourism Management System project can be assessed based on various factors, including its impact on users, technological advancements, and contributions to the tourism industry.
* Optimizing operational processes for travel agencies, hotels, and tour operators, resulting in cost savings and improved productivity.
* Providing transparent pricing, accurate information about destinations and accommodations, and real-time updates on bookings and availability.
* Making travel services more accessible to a broader range of users, including those with disabilities or language barriers.

# **1.7 Organization Report**

The main structure and the report of these project follows the introduction of the system System design, requirement analysis, technology of this survey. However directly introducing my topic I have decide the overview section so that you can get some more picture of the discipline.

* Introduces technology in the project we are using. Survey and comparative studies of

technology used for the project development. Here I have been chosen Java as front-end, Netbeans as framework and using the My SQL as a database.

* Discussing about software and hardware requirements of the system and also include scheduling and planning of the project and also introduces conceptual model.

* Continues with modulization of the system, data designing with schema. Creating the data structures, creating relational schemas, designing user interface, applying security

mechanism and finally examine the result for accuracy.

* This chapter is the part that puts a planned system into action and examine in details the analysis and design of the system. The present chapter discusses the implementation of

the system, highlighting the testing exercise and describing some of the modification and

improvements after testing system.

* We’ll create the different test cases for our system to check that the system is working

properly or having any bugs. This test cases helps as to know that the system is have any

issues or bugs on it. Continues with test reports and user documentation manual for system users.

# **CHAPTER 2: SURVEY OF TECHNOLOGIES**

# **2.1 FRONT-END TECHNOLOGY**

AWT and Swing are used to develop Dekstop-based applications in Java. Awt is an abstract window toolkit that provides various component classes like Label, Button, Text Field, etc. to

show window components on the screen. All these classes are part of the Java.awt package. On

the other hand, Swing is the part of JFC (Java Foundation Classes) built on the top of AWT and

written entirely in Java. The javax.swing API provides all the component classes like JButton,

JTextField, JCheckbox, JMenu, etc. The components of Swing are platform-independent, i.e.,

swing does not depend on the operating system to show the components. Also, the Swing's

components are lightweight.

## **2.2 IDE (Integrated Development Environment):**

**NETBEANS:** NetBeans is a Java-based integrated development environment (IDE). The term also

refers to the IDE’s underlying application platform framework. The IDE is designed to limit coding

errors and facilitate error correction with tools such as the NetBeans Find Bugs to locate and fix

common Java coding problems and Debugger to manage complex code with field watches,

breakpoints, and execution monitoring. The NetBeans IDE can run on any operating system that

supports a compatible JVM including Linux, Windows, and OS X. The underlying NetBeans

platform supports creation of new applications and further development of existing applications

using modular software components. As an application running on the NetBeans Platform, the

NetBeans IDE itself is extensible and can be extended to support new languages. The IDE and

Platform were converted to open source by Sun Microsystems in 2000. Oracle continues to sponsor

the NetBeans project since acquiring Sun in 2010.

**NetBeans Advantages:**

* NetBeans is a cross-platform IDE, which means it can run on various operating systems such as Windows, macOS, and Linux. This cross-platform compatibility allows developers to work seamlessly across different environments.
* NetBeans provides a user-friendly interface with a straightforward layout, making it easy for both beginners and experienced developers to navigate and use the IDE efficiently.
* NetBeans has particularly strong support for Java development. It provides features like intelligent code completion, powerful debugging tools, and a visual designer for building graphical user interfaces (GUIs) with JavaFX.
* NetBeans is released under the Apache License, making it free to use and open source. This not only reduces costs for development teams but also allows developers to inspect and modify the source code as needed.

# **2.3 BACK-END TECHNOLOGY**

MYSQL is one of the most advanced general-purpose object-relational database management system and is open-source. Being an open-source software, its source code is available under MYSQL license, a liberal open-source license. Anyone with the right skills is free to use, modify, and distribute MYSQL in any form. As it is highly stable, very low effort is required to maintain this DBMS. For example, you can define your own data types, build out custom functions, even write code from different programming languages without recompiling your database. MYSQL tries to conform with the SQL standard where such conformance does not contradict traditional features or could lead to poor architectural decisions. Many of the features required by the SQL standard are supported, though sometimes with slightly differing syntax or function.

# **CHAPTER 3: REQUIRMENT AND ANALYSIS**

# **3.1 Project Definition**

The main objective of the Tourism Management System is to manage the details of Customer, Hotel Booking, Cancellation and Tourism places. It manages all the information about Users, Hotel, Packages etc.

This application will help in accessing the information related to the travel to the particular destination with great ease. The users can track the information related to their tours with great ease through this application. The travel agency information can also be obtained through this application.

Through this system, the propose system is highly automated and makes the travelling activities much easier and flexible. The user can get the very right information at the very right time. This system will include all the necessary fields which are required during online reservation time.

Tourists can register by providing personal details, make new reservation and book only one hotel and package and can make cancellation.

# **Requirements Specification**

The following are the various types of requirements:

# **3.2.1 Functional Requirements:**

* The system shall provide the user the ability to Sign up and create the username

and password.

* Users, including tourists, service providers, and administrators, should have secure and role-based access to the system.
* Enable users to book travel packages, accommodations, transportation, and activities seamlessly.
* Provide a centralized dashboard for service providers to manage bookings, update offerings, and analyze performance.

# **3.2.2 Non-Functional Requirements:**

* The system should respond to user requests within a specified timeframe.
* The system should be scalable to handle an increasing number of users and transactions.
* Provide user manuals and guides for system users.
* The system shall be cost efficiently. It shall not cost a lot if there is not many users.
* The system shall be scalable and robust.

# **Planning And Scheduling**

Planning and scheduling can be termed as most complicated part of software development. Success and failure of a project largely depends on how it has been planned, while its timely completion depends on its scheduling. Planning, for our purposes, can be thought of determining all the small tasks that must be carried out in order to accomplish the goal. Planning also takes into account, rules, and known as constraints, which, control when certain task can or cannot happen? Scheduling can be thought of as determining whether adequate resources are available to carry out the plan. Improper planning can lead to unusual late release of software which is not desirable.

**GANTT CHART:**

* GANTT charts display the tasks in a project as a box or line showing the calendar

duration of the task on the horizontal axis. Tasks are normally arranged in date order on the vertical axis. The time relation of all tasks to each other is therefore clearly apparent in a GANTT chart. The project status can be easily determined at intermediate dates in the project, and progress of individual tasks can be shown by filling in the task boxes.

* A Gant chart is a horizontal bar chart developed as a production control tool in 1917 by Frequently used in project management, a Gantt chart provides a graphical

illustration of a schedule that helps to plan, coordinate and track specific tasks in a project.

* A gantt chart is simply a timeline view of your project. The left hand side is a list of

tasks for your project organized into groups. The image below is an example of a

gantt chart that could be used to plan a concert.

* Gantt charts are especially useful for any project where you want to visualize how

long your project will take. You can also link tasks together by creating dependencies

between tasks. This ensures that tasks are done in the correct order. He was able to

really get a grip on his project and keep it under control.

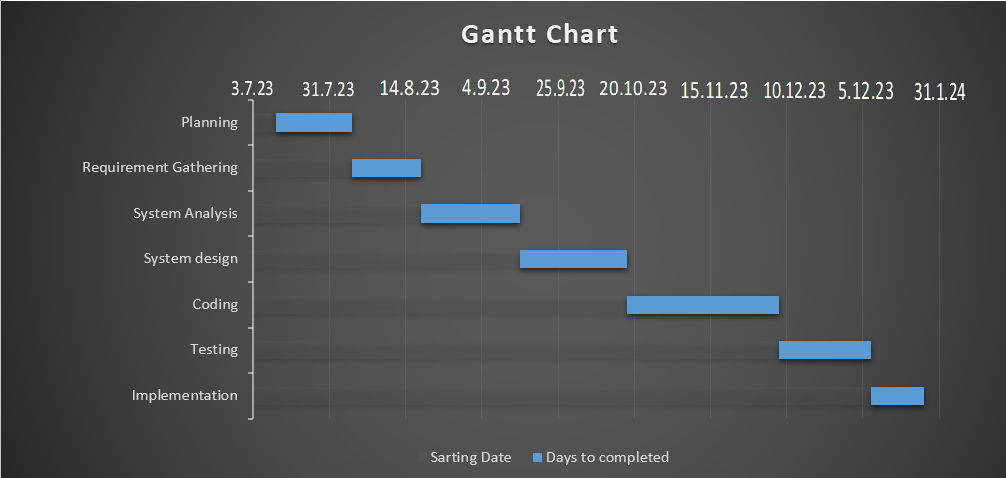
* Another great advantage that gantt charts provide is the ability to see who is doing

what and when they are doing it. This allows for better project planning by efficiently

assigning tasks to people on your team and scheduling them according to their current

workload. Gantt charts have become a common technique for representing the phases

and activities of a project because so they can be understood by a wide audience.



# **3.4 Software and Hardware Requirements**

# **3.4.1 Hardware Specification:**

|  |  |  |
| --- | --- | --- |
| **COMPONENT** | **MINIMUM** | **RECOMMENDED** |
| **PROCESSOR** | Intel i3 and above | Intel i3 and above |
| **RAM** | 2.00 GB | 4.00 GB |
| **MEMORY** | 256GB (HDD) | 512GB (HDD) |
| **INPUT PERIPHERALS** | Keyboard & mouse | Keyboard & mouse |

# **3.4.2 Software Specification:**

Specification Software means a collection of programs where the objective is to enhance the capabilities of the hardware machine. The following defines the software of the proposed system developments.

First of all, the system will work on any platform. Internet connection is a must to reach the system.

Moreover, most of the application will be coded by Java Software Specification

* **Operating system:** Windows 7 or higher
* **Development:** NetBeans IDE 7.3.1
* **Frontend:** JAVA Swing
* **Backend:** MYSQL 8.0CE

# **3.5 Preliminary Product Description**

The first step in the system development life cycle is the preliminary investigation to

determine the feasibility of the system. The purpose of the preliminary investigation is to

evaluate project requests. It is not design study nor does it include the collection of details to

describe the business system in all respect. Rather, it is the collecting of information that

helps committee members to evaluate the merits of the project request and an information

judgment about the feasibility of the proposed project.

**Analysts working on the preliminary isnvestigation should accomplish the following**

**objective:**

* Clarify and understand the project request.
* Determine the size of the project.
* Assess costs and benefits of alternative approaches.
* **Benefit to Organization**

The organization will obviously be able to gain benefits such as savings in operational cost, reduction in paperwork, better utilization of human resource and more presentable image increasing goodwill.

* **The Initial Cost**

The initial cost of setting up the system will include the cost of hardware software and labour. The same has to bear by the organization.

# **Modules Description**

1. **Login Module :**

A login module is a crucial component of any system that requires user authentication and access control. It's a fundamental part of security and user identity management. Below is an outline for a login module in the context of a Travel and Tourism Management System.

1. **Sign up Module:**

A signup module is a critical component for onboarding new users into a system. It involves user registration, information collection, and account creation. Here's a structured outline for a signup module within the context of a Travel and Tourism Management System.

1. **Add Details Module:**

Add Details Module in a Travel and Tourism Management System is responsible for gathering and managing detailed information about various elements such as tourist preferences, travel history, service offerings by providers, and other relevant data.

1. **Payment Module:**

Payment Module in a Travel and Tourism Management System is a crucial component that facilitates secure and seamless financial transactions.

# **Feasibility Study**

It is a measure of how beneficial or practical the development of information system would be to an organization can be termed as Project Feasibility. Feasibility analysis is the process by which feasibility is measured. Feasibility should be measured throughout the life cycle. The scope and complexity of an apparently feasible project can change after the initial problems and opportunities are fully analyzed after the system has been designed. Thus, a project that is feasible at one point may become infeasible later.

The checkpoints for feasibility study are:

* A survey phase checkpoint
* A study phase checkpoint.
* A definition phase checkpoint.
* A selection phase checkpoint.
* Procurement phase checkpoint.
* Design phase checkpoint.

Generally, there are following areas of risk for a new system that are Considered when confirming project feasibility:

* **Operational feasibility/Organizational and cultural feasibility**: It is measure of how well the solution will work in the organization. It is also a measure of how people feel about the system or project. This project is well developed based on the customs and culture followed by the organization and it is easy to use. So, it is operationally and culturally feasible.
* **Economic feasibility:** It is a measure of cost-effectiveness of a project or solution. This is often called a cost-benefit analysis. This project is economically feasible because it is with minimal of cost.
* **Technical feasibility:** It is a measure of how the practicality of a specific technical solution and the availability of technical resources and expertise. By using this software, it is easy to enter the details of Order as well as the details of master forms. In addition to this, calculation of salary and commission is also possible using this software.
* **Schedule feasibility:** It is a measure of how responsible the project timetable is. According to the organization, this software they have to use commercially in the next economical year. So, the schedule can be managed as they give enough time to build this software

# **CHAPTER 4: SYSTEM DESIGN**

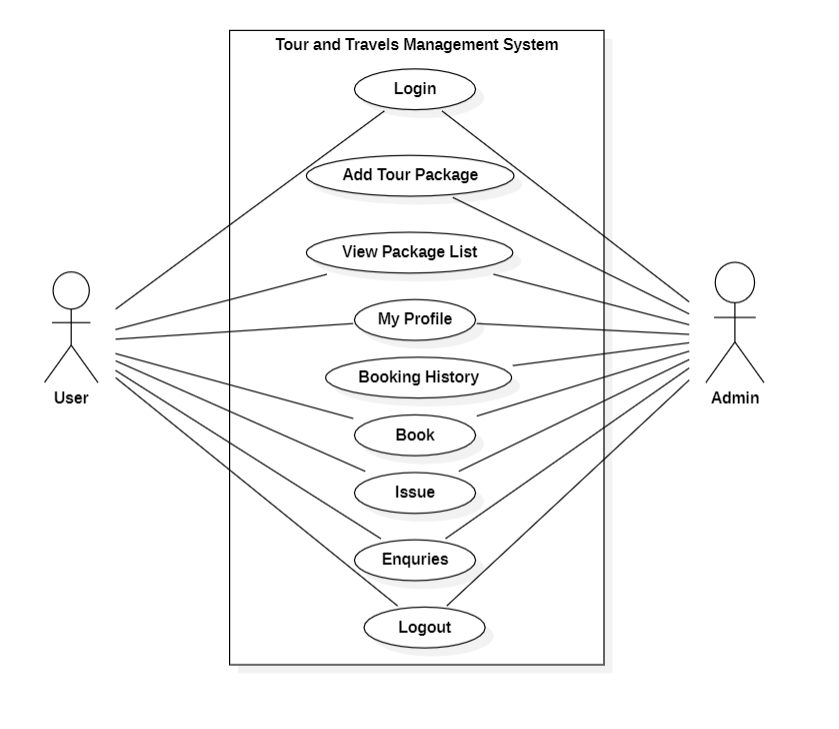
Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization. Once the software requirements have been analyzed and specified the software design involves three technical activities - design, coding, implementation, and testing that are required to build and verify the software. The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer’s requirements into finished software or a system. Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data.

# **UML Diagrams**

# **4.1.1 USE CASE Diagram**

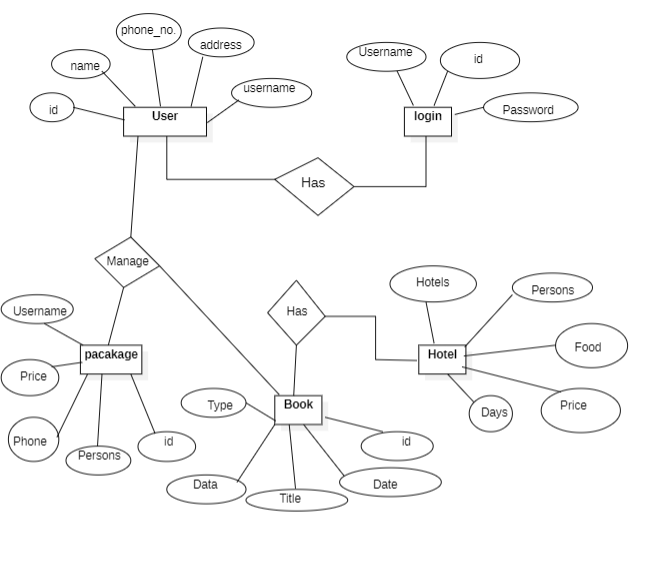
A Use case is a description of set of sequence of actions. Graphically it is rendered as an ellipse with solid line including only its name. Use case diagrams model behavior within a system and helps the developers understand of what the user require. The stick man represents what is called an actor. Use case diagram can be useful for getting an overall view of the system and clarifying who can do and more importantly what they cannot do. Use case diagram consists of use cases and actors and shows the interaction between the use case and actors.

* The purpose is to show the interactions between the use case and actor.
* To represent the system requirements from user’s perspective.
* An actor could be the end-user of the system or an external system.

****

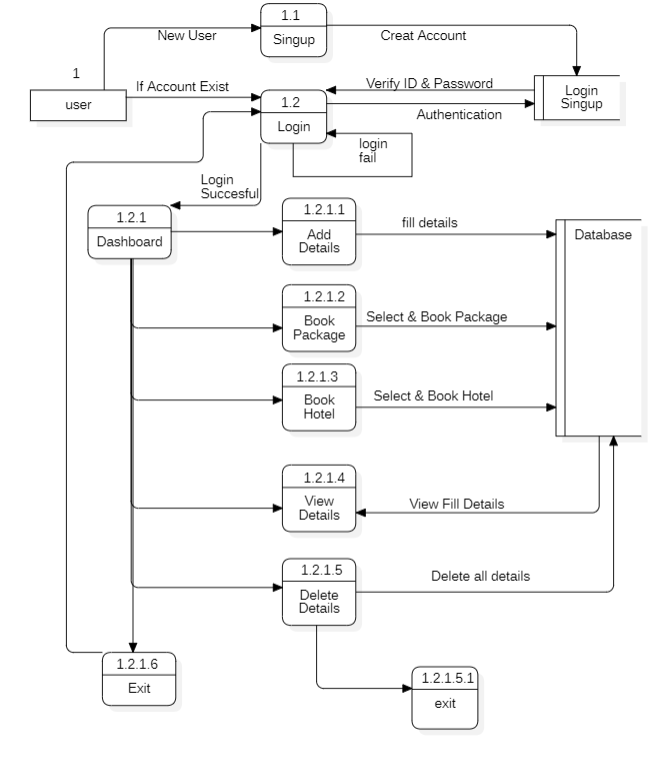
# **4.1.2 ER Diagram**

* ER model stands for an Entity-Relationship model. It is a high-level data model. This model is used to define the data elements and relationship for a specified system.
* It develops a conceptual design for the database. It also develops a very simple and easy to design view of data.
* In ER modeling, the database structure is portrayed as a diagram called an entity- relationship diagram.
* The below ER Diagram for TRAVEL AND TOURISM MANAGEMENT system shows different entities and their relationship with each other.

****

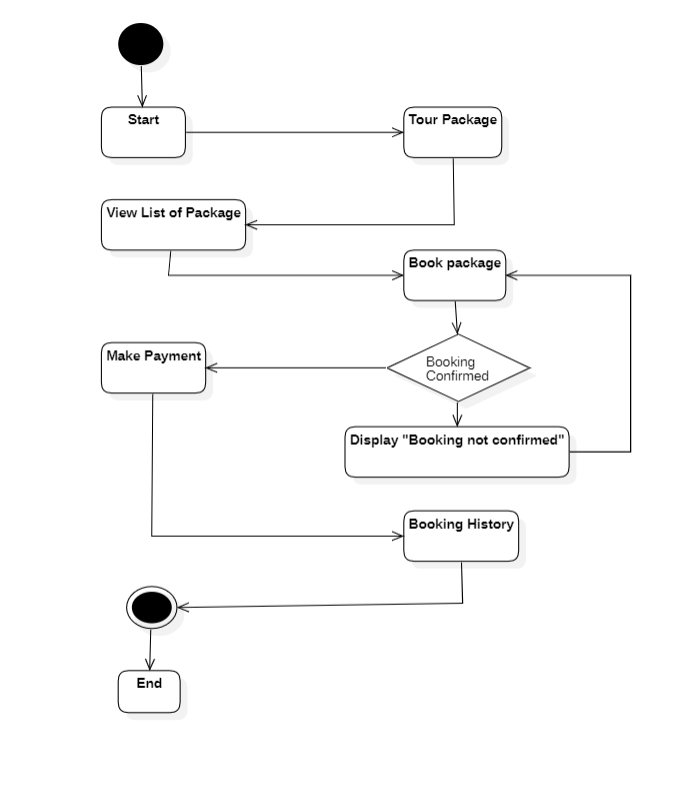
# **4.1.3 DFD Diagram**

* A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both.
* It shows how data enters and leaves the system, what changes the information, and where data is stored.
* The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.
* Below diagram shows how a data flows in a TRAVEL AND TOURISM MANAGEMENT system.



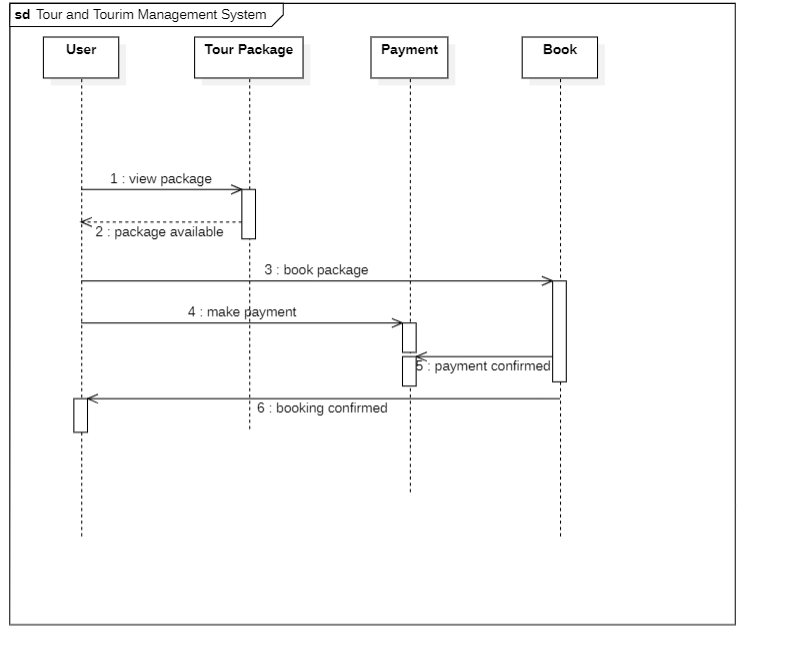
# **4.1.4 Activity Diagram**

* Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.
* The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.
* Activity diagram is sometimes considered as the flowchart. Although the diagrams look like a flowchart, they are not.



# **4.1.5 Sequence diagram**

A sequence diagram is a graphical representation of interactions between objects or components in a system, typically used to visualize the flow of messages or actions over time. In the context of a Tourism Management System, a sequence diagram can illustrate how different elements in the system interact with each other when a specific action or use case is executed.



# **DATA DESIGN**

Data design is process of designing a database. the main output of a data design is a

detailed logical data model of database. A logical data model is one of the main outputs of data design. The data model is usually represented as an entity relationship diagram or ER diagram. While a person can do both data design and database analysis, these are two different tasks. Database analysis takes that model and applies it to one or more database engines.

# **4.2.1 SCHEMA DESIGN**

A database schema is the skeleton structure that represents the logical view of the entire database. It defines how the data is organized and how the relations among them are associated. It formulates all the constraints that are to be applied on the data.

A database schema defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams. It’s the database designers who design the schema to help programmers understand the database and make it useful.

* **Account:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| username | name | password | security | answer |

* **Customer:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| username | id | id\_number | name | gender | country | address | phone | email |

* **Book package:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| username | package | persons | id | id\_number | phone | price |

* **Book Hotel:**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| username | hotel | persons | days | ac | food | id | id\_number | phone | price |

* **Hotel:**

|  |  |  |  |
| --- | --- | --- | --- |
| hotel | cost\_per\_day | food\_charges | ac\_charges |

# **4.2.2 DATA INTEGRITY AND CONSTRAINTS**

Data integrity is the maintenance and the assurance of the accuracy and consistency of data over its entire lifecycle, and is a critical aspect to the design, implementation and usage of any system which stores, processes, or retrieves data.

* **Account:**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Data Types** | **Constraints** |
| Username | String | Required |
| Name | String | Required |
| Password | String | Required |
| Security | String | Required |
| Answer | String | Required |

* **Customer:**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Data Types** | **Constraints** |
| Username | String | Required |
| ID | String | Required |
| ID\_Number | String | Required, Unique |
| Name | String | Required |
| Gender | String | Required |
| Country | String | Required |
| Address | String | Required |
| Phone | String | Required |
| Email | String | Required, Unique |

* **Book package:**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Data Types** | **Constraints** |
| Username | String | Required |
| Package | String | Required |
| Persons | Int | Required |
| ID | String | Required |
| ID\_Number | String | Required, Unique |
| Phone | String | Required |
| Price | Int | Required |

* **Book Hotel:**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Data Types** | **Constraints** |
| Username | String | Required |
| Hotel | String | Required |
| Persons | Int | Required |
| Days | Int | Required |
| AC | Boolean | - |
| Food | Boolean | - |
| ID | String | Required |
| ID\_Number | String | Required, Unique |
| Phone | String | Required |
| Price | Int | Required |

* **Hotel:**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Data Types** | **Constraints** |
| Hotel | String | Required |
| Cost Per Days | Int | Required |
| Food Charges | Int | Required |
| AC Charges | Int | Required |

# **4.2.3 ALGORITHM DESIGN**

Algorithm is a step-by-step procedure, which defines a set of instructions to be executed in a certain order to get the desired output. Algorithms are generally created independent of underlying languages, i.e., an algorithm can be implemented in more than one programming language.

**Pseudo-code:**

Pseudo-code is a plain language description of the steps in an algorithm or another system. Pseudo-code often uses structural conventions of a normal programming language, but is intended for human reading rather than machine reading.

**Pseudo-code for Tourism Management System:**

**Users:**

1. Start.
2. New user Signup.
3. User Login.
4. Add Details.
5. Update Details.
6. Check Package.
7. Book Package.
8. Check Hotel.
9. Book Hotel.
10. Payment.
11. Exit.

# **4.3 USER INTERFACE DESIGN:**

The user interface (UI) is the point of human-computer interaction and communication in a device. This can include display screens, keyboards, a mouse and the appearance of a desktop. It is also the way through which a user interacts with an application. UI stands for User Interface. UI is the part of the Dekstop-Based application which a user interacts with. It simple terms, it is everything you see and touch, such as buttons, colours, fonts, navigation, etc.

# **TEST CASE DESIGN:**

The Test Case is the sets of conditions or variables under which a tester will determine whether the system under test satisfies requirements or work correctly. The process of developing test cases can also help to find problems in the requirements or design of application. Test case design refers to how you set-up your test cases. It is important that your tests are designed well, or you could fail to identify bugs and defects in your software during testing. There are many different test case design techniques used to test the functionality and various features of your software.

* **Login test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tests case**  **No.** | **Test case name** | **Test case description** | **Test case** | |
| **Steps** | **Expected** |
| **TC1** | Validate username and password | To verify  username | The username field is left empty  or the username is incorrect. | An error messages  “Enter Username” |
| To verify  password | The password field is left empty  or password is incorrect. | An error messages  “Enter Password” |

* **Registration test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tests case**  **No.** | **Test case name** | **Test case description** | **Test case** | |
| **Steps** | **Expected** |
| **TC2** | Validate  username | To verify  username | The username field is left empty. | An error messages  “Enter Username” |
| The username is Reapeated. | An error messages  “ Username Already  exist” |
| **TC3** | Validate  the name | To verify  name | The name field is left empty  or the name is only Alphabetical format | An error messages  “Enter Username” |
| **TC4** | Validate the Password | To verify password | The Password field is left empty  Or the password is use special char, number& alphabet. | An error messages  “Enter Password” |
| **TC5** | Validate the answer | To verify answer | The answer field is left empty. | An error messages  “Enter answer” |

* **Add details test cases:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tests case**  **No.** | **Test case name** | **Test case description** | **Test case** | |
| **Steps** | **Expected** |
| **TC6** | Validate  Id no. | To verify  Id no. | The Id no.field is left empty. | An error messages  “Enter Id no.” |
| The Id no.is valid. | An error messages  “Enter valid Id no” |
| **TC7** | Validate  the name | To verify  name | The name field is left empty  or the name is only Alphabetical format | An error messages  “Enter name” |
| **TC8** | Validate the country | To verify country | The country field is left empty | An error messages  “Enter country” |
| **TC9** | Validate the address | To verify address | The address field is left empty. | An error messages  “Enter address” |
| **TC10** | Validate the phone | To verify phone | The phone field is left empty. | An error messages  “Enter phone no.” |
| The phone no.is only int format and 10 digit | An error messages  ”Enter valid phone no.” |
| **TC11** | Validate the email | To verify email | The email field is left empty. | An error messages  “Enter email” |
| The email should follow  the format e.g., @,  domain name, etc. | An error message “Enter  valid email address” |

# 

# **CHAPTER 5: IMPLEMENTATION AND TESTING**

# **5.1 IMPLEMENTATION AND APPROCHES**

Project implementation (or project execution) is the phase where visions and plans become reality. This is the logical conclusion, after evaluating, deciding, visioning, planning, applying for funds and finding the financial resources of a project. Technical implementation is one part of executing a project.

An implementation plan for a project refers to a detailed description of actions that demonstrate how to implement an activity within the project in the context of achieving project objectives, addressing requirements, and meeting expectation.

# **CODING DETAILS AND EFFICIENCY**

# **5.2.1 CODE EFFICIENCY:**

Code efficiency is a broad term used to depict the reliability, speed and programming methodology used in developing codes for an application. Code efficiency is directly linked with algorithmic efficiency and the speed of runtime execution for software. It is the key element in ensuring high performance. The goal of code efficiency is to reduce resource consumption and completion time as much as possible with minimum risk to the business or operating environment. The software product quality can be accessed and evaluated with the help of the efficiency of the code used.

* To improve performance of Java code we use following technique to optimize code.
* To remove unnecessary code or code that goes to redundant processing.
* To make use of optimal memory and non-volatile storage.
* To ensure the best speed or run time for completing the algorithm.
* To make use of reusable components wherever possible.
* To create programming code that ensures data integrity and consistency.
* To make use of error and exception handling at all layers of software, such as the user

interface, logic and data flow.

* To develop programming code that's compliant with the design logic and flow.
* To make use of coding practices applicable to the related software.
* To optimize the use of data access and data management practices.
* To use the best keywords, data types and variables, and other available programming

concepts to implement the related algorithm.

# **TESTING APPROACH**

Implementation of test strategy for a particular project is known as "test approach". The test approach is usually defined in all test plans and test designs, i.e., how testing would be carried out. Test approach has two techniques:

**Proactive** - an approach in which the test design

process is initiated as early as possible in order to find and fix the defects before the build is created.

**Reactive** - an approach in which the testing is not started until after design and

coding are completed. Functional Testing is defined as a type of testing which verifies that each function of the software application operates in conformance with the requirement specification. This testing mainly involves black box testing and it is not concerned about the source code of the application.

The prime objective of Functional testing is checking the functionalities of the software system. Examples of Functional testing are:

* Unit testing
* Integration Testing
* Beta Testing

# **5.3.1 UNIT TESTING**

Unit testing is a type of software testing that focuses on individual units or components of a

software system. The purpose of unit testing is to validate that each unit of the software works as intended and meets the requirements. Unit testing is typically performed by developers, and it is performed early in the development process before the code is integrated and tested as a whole system.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test**  **Case ID** | **Test Field** | **Test Scenario** | **Test Data** | **Expected Result** | **Actual Result** | **Status (pass/fail)** |
| **TC1** | Login | User tries to Login | Username  &  Password | An alert message pops up | The user is denied to log in | **Fail** |
| Username  &  Password | The user id logged in successfully | The user is  logged in  successfully | **Pass** |
| **TC2** | Sign Up | New User tries to Signup | Username,Name,Password,Security Question & Answer | The user fills incomplete data  (Alert: Please enter the details) | The user fills incomplete data  (Alert: Please enter the details) | **Fail** |
| Username ,Name ,Password,Security Question & Answer | The user proceeds for signup and data entered successfully to database | The user proceeds for signup and data entered successfully to database | **Pass** |
| **TC3** | Add personal details | User fill details to add personal details | Choose Id Type,  Id no. , Phone no. , Address, Country, Email Id, Gender | User details added successfully | User details added successfully | **Pass** |
| **TC4** | Update Personal Details | User update details to update personal details | Choose Id Type,  Id no. , Phone no. , Address, Country, Email Id, Gender | User details update successfully | User details update successfully | **Pass** |
| **TC5** | View Details | User Details | Choose Id Type,  Id no. , Phone no. , Address, Country, Email Id, Gender | Should display the user details | User details displayed successfully | **Pass** |
| **TC6** | View  Packages | All Packages | Various Types of Packages | Should display the available packages | Packages displayed successfully | **Pass** |
| **TC7** | Book Package | User book package to  Book package | Package Details( Total Persons,  Price) | User book package successfully | User book package successfully | **Pass** |
| **TC8** | View Book Package | Book Packages | Booked Package Details | Should display the book packages | Book Packages displayed successfully | **Pass** |
| **TC9** | View Hotels | All Hotels | Various Types of Hotels | Should display the available hotels | Hotels displayed successfully | **Pass** |
| **TC10** | Book Hotels | User book hotel to  Book hotels | Hotels Details( Total Persons,  Price,foodincluded) | User book hotel successfully | User book hotel successfully | **Pass** |
| **TC11** | View Book Hotels | Book Hotels | Booked Hotels Details | Should display the book hotel | Book hotel displayed successfully | **Pass** |

# **5.3.2 SYSTEM TESTING**

The next level of testing is system testing and acceptance testing. This testing is done to check if the system has met its requirements and to find the external behavior of the system. System testing involves two kinds of activities.

* Integration testing
* Acceptance testing

The next level of testing is called the Integration testing. In this many tested modules are combined into subsystems, which were then tested. Test case data is prepared to check the control flow of all the modules and to exhaust all possible inputs to the program. Situations like treating the modules when there is no data entered in the test box is also tested.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Field** | **Test Scenario** | **Test Data** | **Expected Result** | **Actual Result** | **Status**  **(Pass/Fail)** |
| **TC1** | Sign Up | New User tries to Signup | Username, Name, Password , Security Question & Answer | The user proceeds for signup and data entered successfully to database | The user proceeds for signup and data entered successfully to database | **Pass** |
| Sign in | User tries to Login | Username  &  Password | The user id logged in successfully | The user is  logged in  successfully | **Pass** |
| Add personal details | User fill details to add personal details | Choose Id Type,  Id no. , Phone no. , Address, Country, Email Id, Gender | User details added successfully | User details added successfully | **Pass** |
| Update Personal Details | User update details to update personal details | Choose Id Type,  Id no. , Phone no. , Address, Country, Email Id,  Gender | User details update successfully | User details update successfully | **Pass** |
| View Details | User Details | Choose Id Type,  Id no. , Phone no.,  Address, Country, Email Id, Gender | Should display the user details | User details displayed successfully | **Pass** |
| View  Packages | All Packages | Various Types of Packages | Should display the available packages | Packages displayed successfully | **Pass** |
| Book Package | User book package to  Book package | Package Details( Total Persons,Price) | User book package successfully | User book package successfully | **Pass** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Field** | **Test Scenario** | **Test Data** | **Expected Result** | **Actual Result** | **Status**  **(Pass/Fail)** |
| **TC2** | Sign Up | New User tries to Signup | Username, Name, Password , Security Question & Answer | The user proceeds for signup and data entered successfully to database | The user proceeds for signup and data entered successfully to database | **Pass** |
| Sign in | User tries to Login | Username  &  Password | The user id logged in successfully | The user is  logged in  successfully | **Pass** |
| Add personal details | User fill details to add personal details | Choose Id Type,  Id no. , Phone no. , Address, Country, Email Id, Gender | User details added successfully | User details added successfully | **Pass** |
| Update Personal Details | User update details to update personal details | Choose Id Type,  Id no. , Phone no. , Address, Country, Email Id,  Gender | User details update successfully | User details update successfully | **Pass** |
| View Details | User Details | Choose Id Type,  Id no. , Phone no.,  Address, Country, Email Id, Gender | Should display the user details | User details displayed successfully | **Pass** |
| View  Packages | All Packages | Various Types of Packages | Should display the available packages | Packages displayed successfully | **Pass** |
| Book Package | User book package to  Book package | Package Details( Total Persons,Price) | User book package successfully | User book package successfully | **Pass** |

# **BETA TESTING**

Beta Testing is performed by real users of the software application in a real environment. Beta testing is one of the types of User Acceptance Testing. A Beta version of the software, whose feedback is needed, is released to a limited number of end-users of the product to obtain feedback on the product quality. Beta testing helps in minimization of product failure risks and it provides increased quality of the product through customer validation. It is the last test before shipping a product to the customers. One of the major advantages of beta testing is direct feedback from customers.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test**  **Case ID** | **Test Scenario** | **Test Data** | **Expected Result** | **Actual Result** | **Status (pass/fail)** |
| **TC1** | Check the internet should be connected on devices | Internet  Services | Login/Signup successfully | Login/Signup successfully | **Pass** |
| **TC2** | Update of user data | User Data | Proper entry of fill details and book packages/hotels | Proper entry of fill details and book packages/hotels | **Pass** |

The beta version of the software is delivered to a restricted number of users to accept their feedback and suggestions on quality improvement. Hence, there are two types of beta version:

* **Closed beta version**

Closed beta version, also known as a private beta, it is released to a group of selected and invited people. Those people will test the software and evaluate their features and specifications. This beta version represents the software which is capable of delivering value, but it is not ready to be used by everyone. Because it shows the issues like lack of documentation or missing vital features.

* **Open beta version**

Open beta is also known as a public beta. The open beta opened to the public. Any user as a tester can assess the beta version to provide the relevant feedback and reviews. Open beta version improves the quality of the final release. This version helps to find the various undetected errors and issues.

# **CHAPTER 6: RESULT AND DISCUSSION**

# **TEST REPORTS**

Test Report is an important deliverable which is prepared at the end of a Testing

project, or rather after Testing is completed. The prime objective of this document is to

explain various details and activities about the Testing performed for the Project, to the

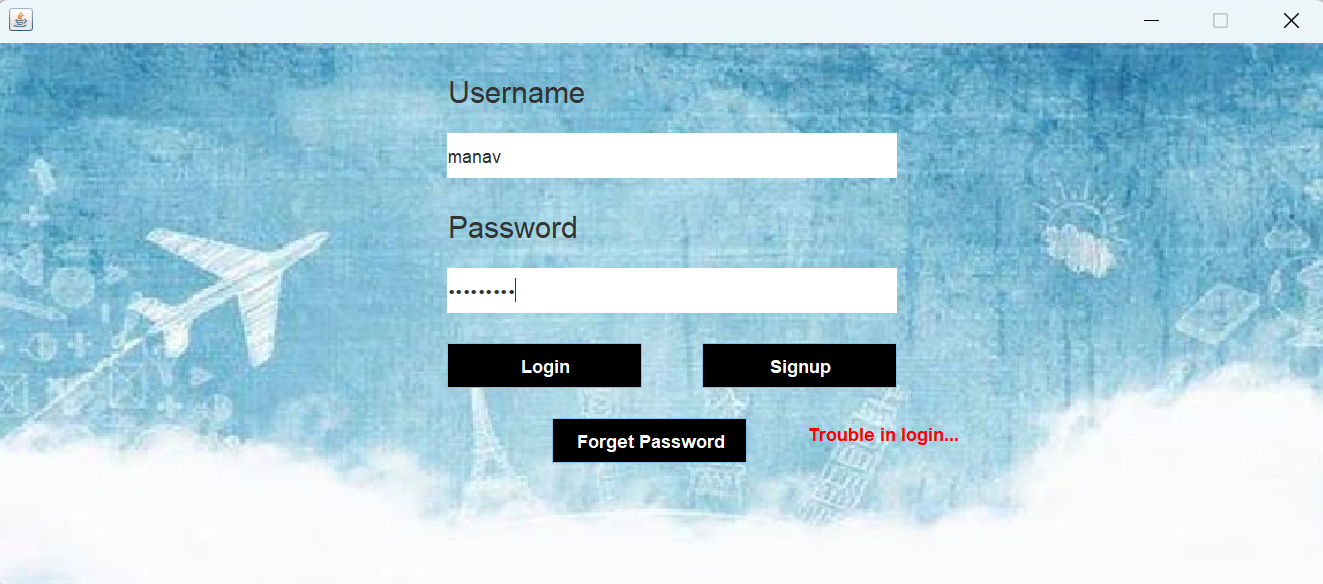
respective stakeholders like Senior Management, Client etc.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Report** | | | | | |
|  | | | | | |
| **Functions** | **Description** | **% TCs**  **Executed** | **% TCs**  **Passed** | **TCs Pending** | **Priority** |
| User Registered /  Login | Check user registration and  login | 100% | 100% | 0 | High |
| Adding users details | Check users are add details | 100% | 100% | 0 | Medium |
| Update user details | Check users are update details | 100% | 100% | 0 | Medium |
| View details | Check users are view details | 100% | 100% | 0 | Medium |
| Book hotel | Check users are book hotel | 100% | 100% | 0 | High |
| View hotels | Check users are view hotels | 100% | 100% | 0 | Medium |
| Book package | Check users are book package | 100% | 100% | 0 | High |
| View book package | Check users are book package | 100% | 100% | 0 | High |

# **USER DOCUMENTATION:**

User documentation is important in any development project. These documents help to explain our product to users by providing them with necessary information. The user manual is essential for every software product because it serves as the ultimate guide regarding our product.

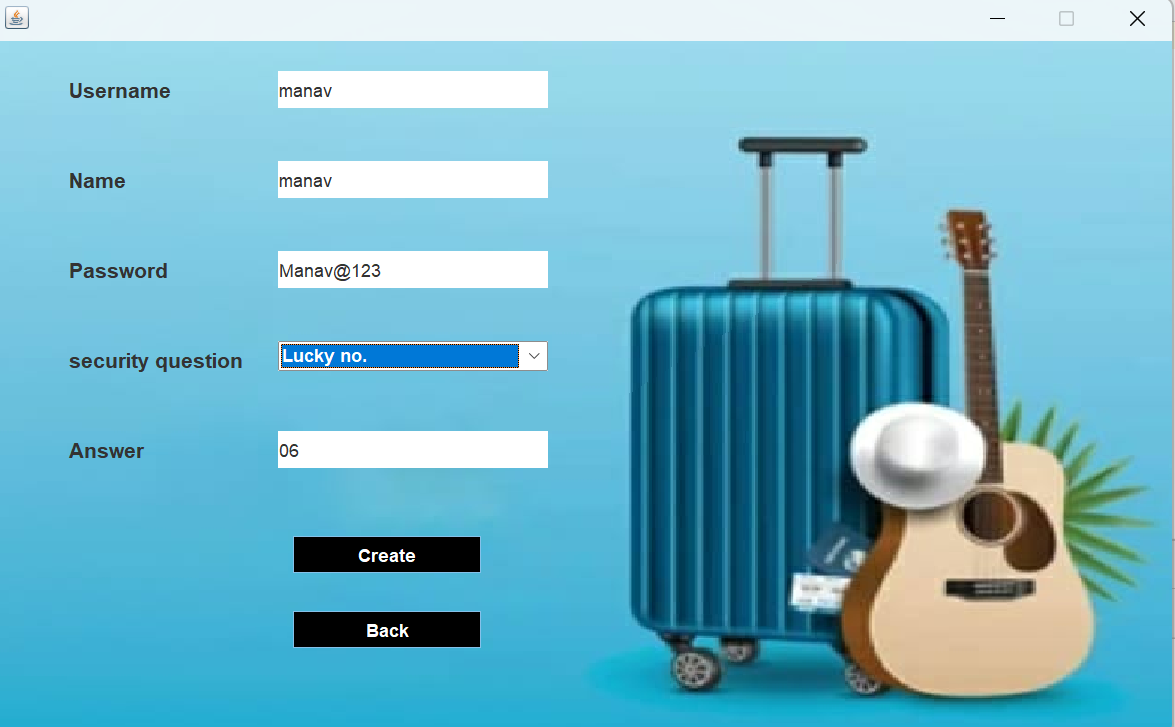
# **Login**

****

The main User Interface (UI) of the software. This page is available to the user on starting the software. The user can Login though this page to access his / her account, If the user does not have an account, he/she can create an account by clicking on the Signup button present on the screen that will navigate the user to the Signup form page which collects some basic details from the user and creates his / her account.

The login page collects information about the user Username and Password and matches it with the database and decides the authenticity of the user and then allows access to the user. If the user makes a wrong entry while entering his Username and Password his / she can correct the same.

# **Signup**



The signup form: this page is displayed when the user needs to create a new account and can be accessed through the login panel by clicking in the signup button. This page collects name ,create a new password , security question and answer and create own username details of the user for creating an account and save the details in a particular form that can be used for further references.

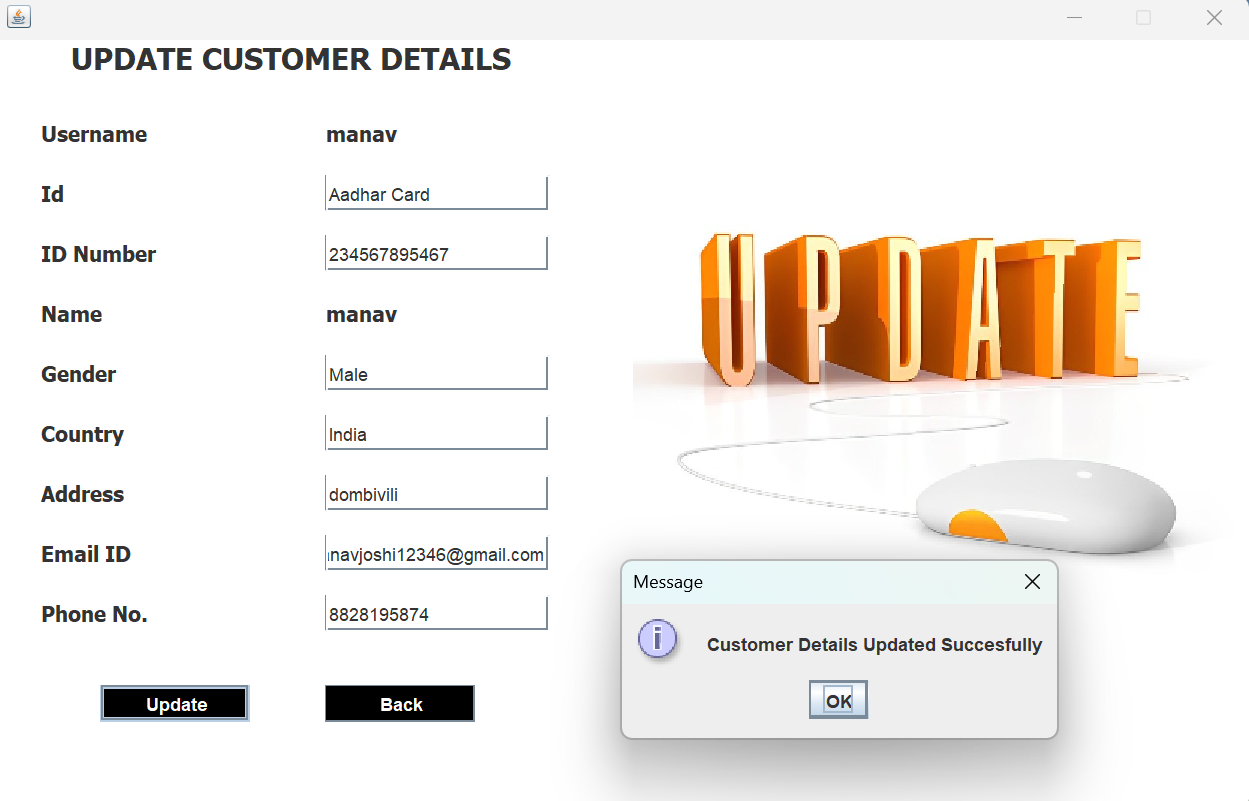
If the username is already exist in database then not create account so the replaced the unique username

# **Add Personal Details**

****

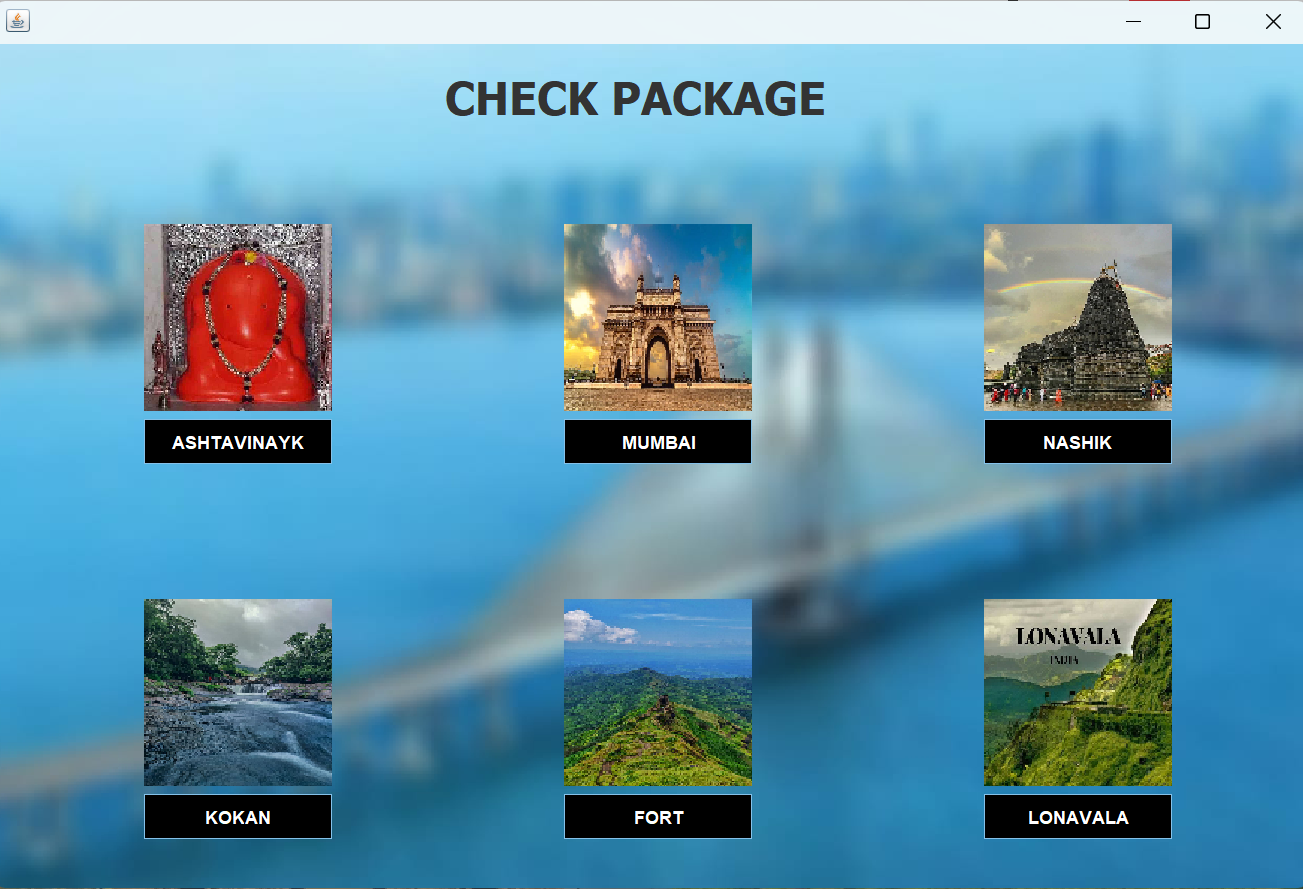
This module allows users to input additional information or details about various aspects of the tourism process, such as Aadhar card no., Email ID, Phone no. , Country, Gender or Address.

# **Update Customer Details**



This module allows users to update the additional information or details where will be the wrong or incorrect about various aspects of the tourism process, such as Aadhar card no., Email ID, Phone no. , Country, Gender or Address.

# **Check Packages**

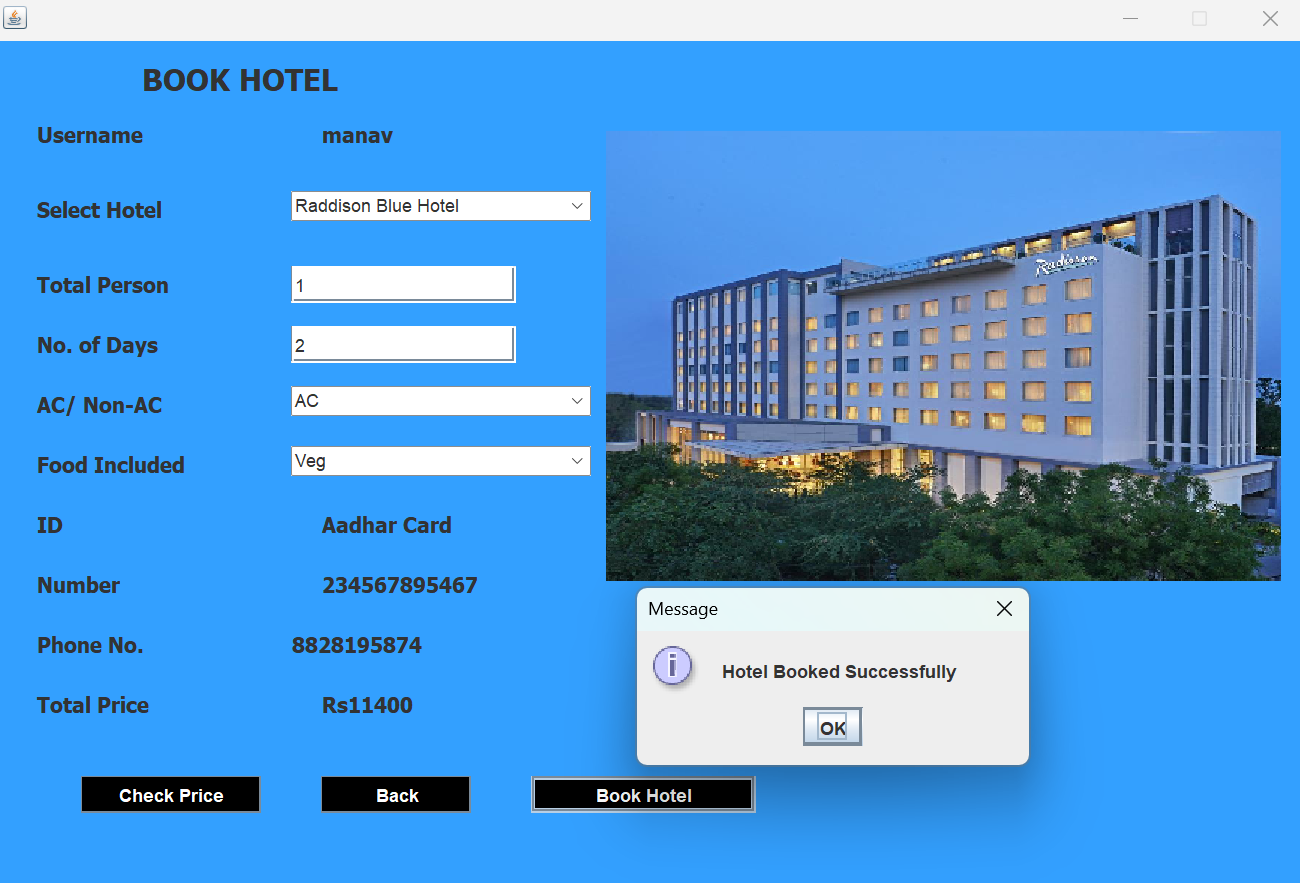


This module allows users can be check the available packages details.

# **Book Packages**

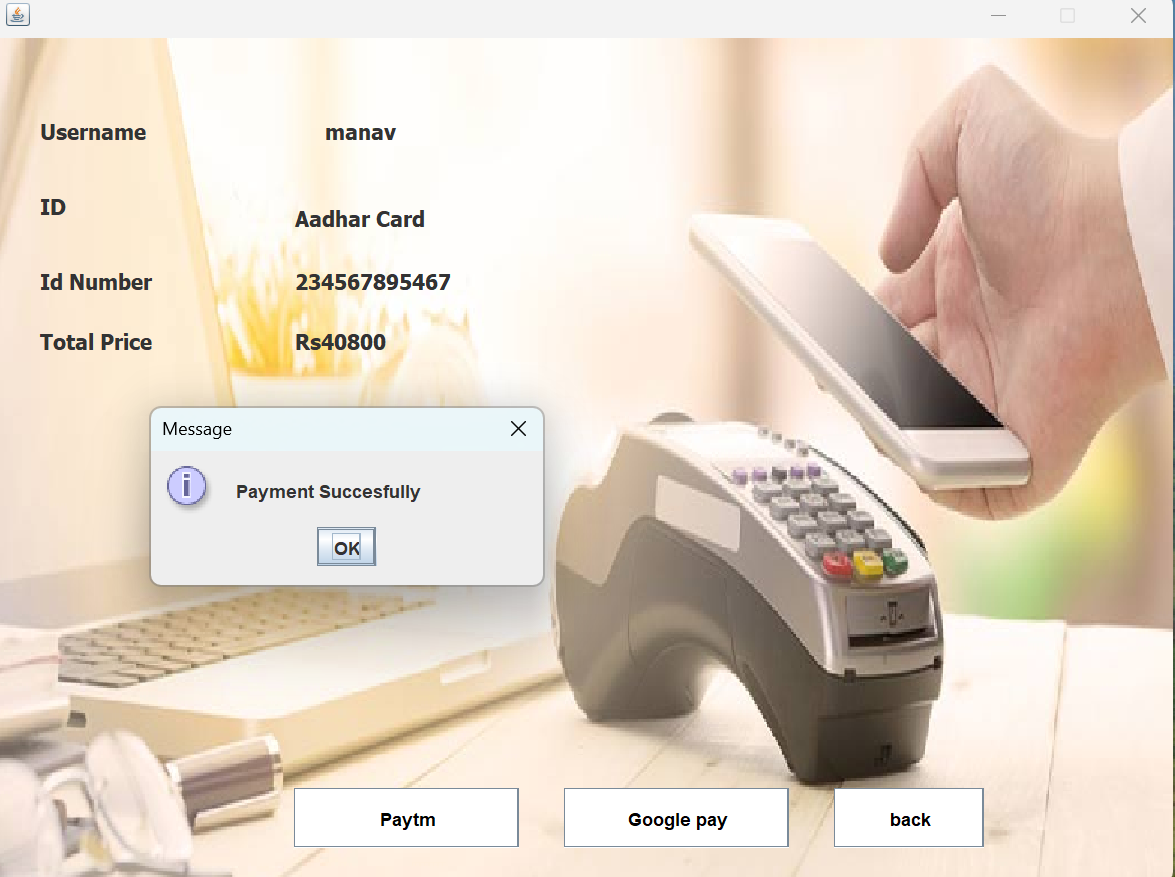
In this module user can be selected the different packages and entering the no. of persons and check the package price or the booked the satisfied package.

# **Book Hotel**



In this module the user can check the hotel in check hotel module and select the specific hotel or fill the following data like total persons , no. of days , ac/no-ac ,food included or not etc. and then chek the price the price will be satisfied then book this hotel.

# **Payments**



In this module will be the calculate the all prices for example booked packages and booked hotel price will be total find out and show on the payment panel and user can be pay this specific amount after pay amount sent the payment succesfully message on screen.

# **CHAPTER 7: CONCLUSION**

# **Conclusion**

The process of the system we can consider here, can maintain the databases of the system. We can insert to the databases and retrieve all the information. The main aim of this project is to help the tourists to manage their trip. It makes all operation of the tour company easy and accurate. The standalone platform makes tourism management easy by handling requests and providing servers for the customers located at different parts of the various cities. Different modules have been incorporated in this project to handle different parts and sector of the tour management field.

# **FUTURE SCOPE OF THE PROJECT**

With the knowledge I have gained by developing this application, I am confident that in the future I can make the application more effectively by adding these services.

* Implementation of blockchain for secure and transparent transactions, including booking accommodations, verifying identity, and managing loyalty programs.
* Implementation of smart city concepts to create intelligent and connected tourist destinations.
* Generate and send payment receipts to users
* We can create android/IOS application also.

# **Reference**

1. SOFTWARE ENGINEERING (Techmax publications)

Author name: M.A. Ansari, Anil Prajapati.

1. SOFTWARE ENGINEERING

Author name: Roger S. Pressman

1. UNIFIED MODELING LANGUAGE

Author name: Gradi Booch, Ivar Jacobson

1. Tutorials - Javatpoint.

www.javatpoint.com. <https://www.javatpoint.com>

1. MYSQL Database tutorials (javatpoint) Https://[www.javatpoint.com/mysql-tutorial](http://www.javatpoint.com/mysql-tutorial)
2. Core Java tutorial (javatpoint). Https://[www.javatpoint.com/java-tutorial](http://www.javatpoint.com/java-tutorial)
3. Java AWT tutorials (javatpoint). Https://[www.javatpoint.com/java-awt](http://www.javatpoint.com/java-awt)