Software Requirements Specification

*For <Travel Management System >*

*Version 1.0*

Prepared by <K21-3881,Muhammad Taha Jawaid>

*<15 December, 2022 >*

**Table of Contents**

[**1. Introduction** 1](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639232)

[1.1 Purpose 1](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639233)

[1.2 Document Conventions 1](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639234)

[1.3 Intended Audience and Reading Suggestions 1](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639235)

[1.4 Product Scope 1](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639236)

[1.5 References 1](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639237)

[**2. Overall Description** 1](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639238)

[2.1 Product Perspective 1](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639239)

[2.2 Product Functions 2](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639240)

[2.3 User Classes and Characteristics 2](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639241)

[2.4 Operating Environment 2](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639242)

[2.5 Design and Implementation Constraints 2](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639243)

[2.6 User Documentation 2](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639244)

[2.7 Assumptions and Dependencies 2](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639245)

[**3. External Interface Requirements** 3](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639246)

[3.1 User Interfaces 3](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639247)

[3.2 Hardware Interfaces 3](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639248)

[3.3 Software Interfaces 3](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639249)

[3.4 Communications Interfaces 3](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639250)

[**4. System Features** 3](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639251)

[4.1 System Feature 1 4](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639252)

[4.2 System Feature 2 (and so on) 4](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639253)

[**5. Other Nonfunctional Requirements** 4](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639254)

[5.1 Performance Requirements 4](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639255)

[5.2 Safety Requirements 5](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639256)

[5.3 Security Requirements 5](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639257)

[5.4 Software Quality Attributes 5](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639258)

[5.5 Business Rules 5](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639259)

[**6. Other Requirements** 5](file:///C:\Users\Khizar07\Documents\MY%203rd%20SEMESTER%20FILES\SRE\Project\IEEE_SRS_Template.docx#_Toc120639260)

# *Introduction:*

## 1.1.1 Introduction of the system:

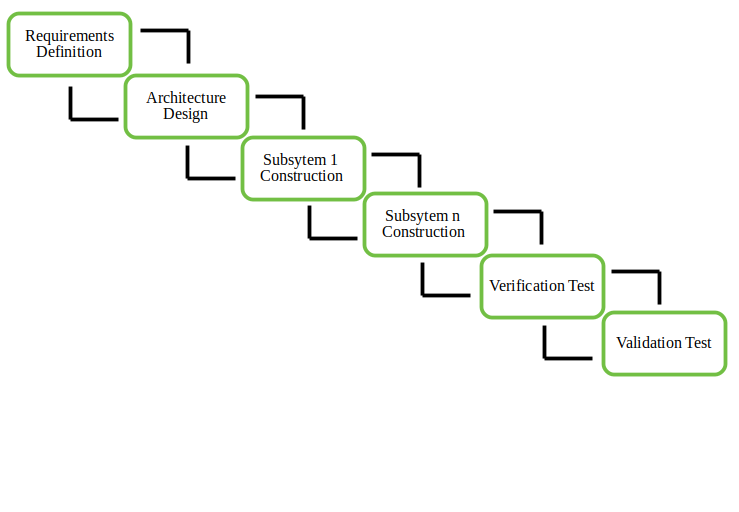
**“Travel and Tourism Management”** system is a system which will be used to book the touring destinations and hotels all around the globe with the help of simple and efficient desktop-based application. The purpose this system i.e. “Travelling Management System” is to register the users who want to travel to different places from all over the globe with a specific and efficient platform from which they can easily book, add, update their rides and hotels according to their choices.

There will be an admin class corresponding to the **User class** in the system, that will add various packages of hotels and destinations to the system/application. The users can register in and book their desired package according to their need. Since the system contains an **Administrative** side as well so all the users can confirm their details through the system admin. In addition, the system will provide the user a view facility through which they can see their confirmation.

**1.1.2 Choosing the Model(SDLC):**

The model which that was chosen for developing this TMS is “Incremental model.” The purpose of choosing that model for developing the system was to deliver the system functionalities into chunks through which user can easily get the details and can correct or tell the developers or team to update some of the functionalities. Therefore, by using incremental approach the system making shall not be dependent on developers only as this will include the user interventions at each stage (like system is as system is divided into different modules so after the customer talks the system can be updated), after completion a final product will be sent to customer after the efficient approach in building the system.

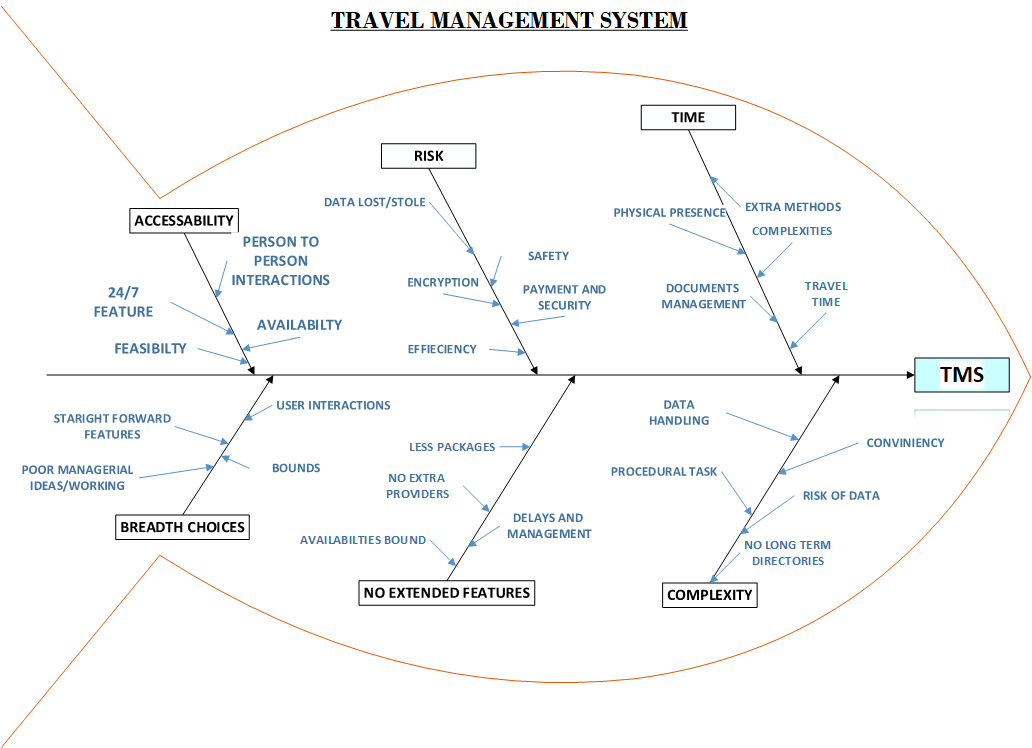
Figure 1:



## 1.1 Purpose:

The purpose of this document is to elaborate the details of **TMS**. Since “Travelling Management System” is an easiest and single platform through which a user can efficiently use it to perform all functionalities of booking from bottom to top (i.e. from registration to till payment). Some basic features that **TMS** shall provide to the user is that all the travelers can easily book the package/trip and can get all the required details of it by some clicks. The tourism management system will allow the user to access all the details i.e. location, events, statistics and reports etc. The document will be explaining all the details of the system, its features, functionalities and constraints etc.

However, the details of the system are briefly provided in the below document which can easily be followed by developers and any stakeholder. The purpose of the system can be easily understanding by the help of fishbone diagram.



## Document Conventions:

This document is written in IEEE format in order to make the Software Requirement Specification for “Travel Management System”. In this document,

1. The subheading for a particular main heading is provided by numeric list
2. TMS: Travel Management System.
3. User: The person who will use the system
4. Classes: Basically represent the two main side of the system that is the admin side and the user side.
5. BO: Business objectives
6. SC: Success criteria
7. DB: data base
8. BR: Business requirement
9. TMA: Travel management company.
10. DFD: Data Flow diagram.
11. ERD: Entity relationship diagram.

## Intended Audience and Reading Suggestions:

The intended audience concerns with the peoples to whom the system is being developed. Our project is basically a prototype for TMS, which is not restricted till the institutes as this can be use by different organizations, companies etc. some of the main intended audience and readers are provided below:

1. Developers
2. People who are willing to explore things and check packages.
3. Media.
4. Educational Institutes.
5. Business Travelers.
6. Multiple Travel Agencies.
7. Booking Engines.
8. Travel Guide & Agencies.

## Product Scope & Product Vision:

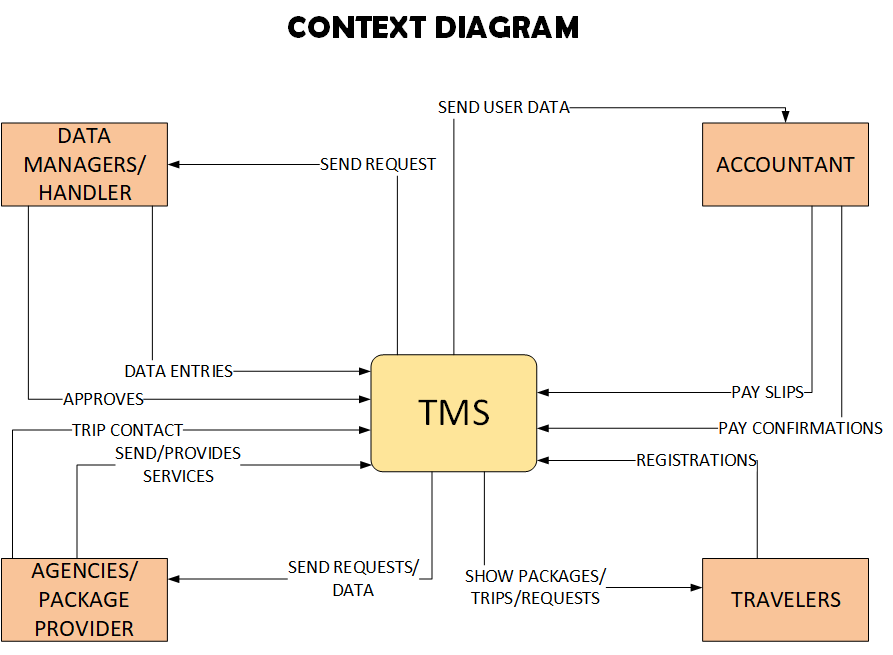
**1.4.0 Project Scope:**

Firstly, Travel management system is an Online travelling platform, which will help the user(Travelers) to register themselves from any place without going to booking offices manually. The basic need for creating the system was to minimize time and load factor that user waste in the manual systems.

Secondly, it is a system that manages all components of a user having business travel and professional level travel.  The tourism management system will allow the user to access all the details i.e. location, events, statistics and reports etc. after registering. The system consists of two main classes user and admin. At first, the system will help the user to first register themselves then the they will do the bookings for their desired places and hotels with some clicks. The system provides various functionalities like bookings, confirmations, travel service providers, trip details in one place for their feasibility which will be handled by the admin.

However, the system is only a desktop based application which will not work the mobile screens in the first publish. The system will be implemented in java language supporting the SQL database. In addition, the system would have a unique id password in order to register the system and without authorization no user will be accessed/authorized to the system as it follows the strictly security and safety requirements.

At last, the TMS will only be considered as a good product in market if it gets the rating of 4.5 pus in first 3 months of the initial release also if more than 50% of the user uses it in first release then it would make the product stable, consistent in its place and will make it efficient and productive for the user.



**1.4.1 Project Vision:**

The system “Travel management system” is for the customers who want to register themselves online and book their desire places and hotels through a specific application. The Travel management system is an online, simple and efficient system which will help the users(travelers) to register themselves in order to book the various places where they want to travel. Travel management system users will have the advantage to book their tickets on time, will have the advantage of paying the amount online, will have the advantage of receiving the digital slips of different purchases they will do. However, for this Travel management system its mandatory that user do have the internet connection in order to register themselves. In addition, from our product the travelers will be able to see their bookings and registrations furthermore they can also track their statistical data of the trip in order to get the details at any time anywhere. At last, in order to complete the registration successfully the user will just have to pay the total bill that he got after adding or updating the details.

## References

The Reference links (while making the SRS document) are given below:

1. <https://ijcsmc.com/docs/papers/October2019/V8I10201903.pdf>
2. <https://www.jenji.io/en-us/resources/what-are-the-advantages-of-a-travel-management-system>
3. <https://www.freeprojectz.com/uml/travel-management-system-class-diagram#:~:text=Classes%20of%20Travel%20Management%20System,all%20the%20operations%20of%20Bookings>
4. <https://www.travelperk.com/blog/travel-management-system/>
5. <https://www.booking.in>

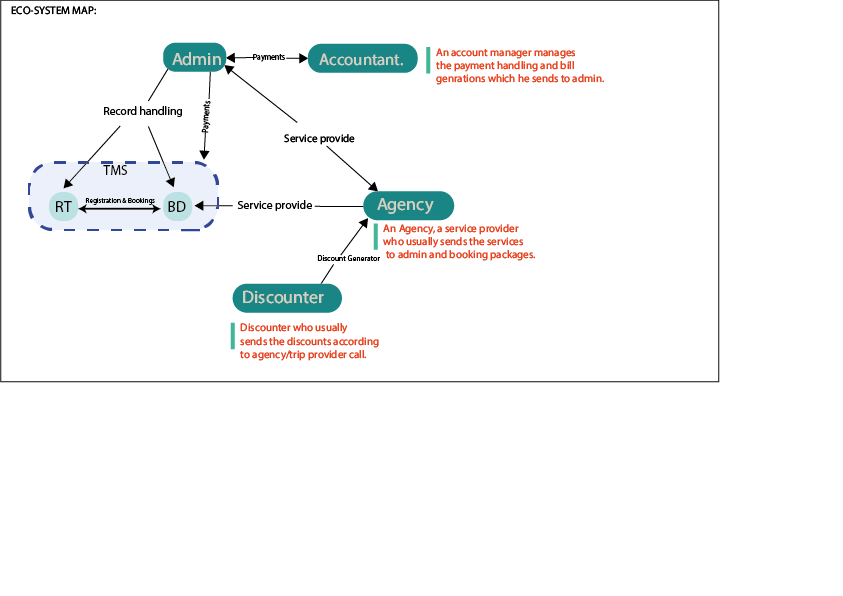
# *Overall Description:*

## Product Perspective

In different existing **TMS (Travel management system)** that is actually a platform through which users (travelers etc.) can register into the system they can just simply book the flights hotels and the destinations. Basically in current systems the user has to look for different packages from different agencies to get their desired packages that will require much more effort from user side to book his/her trip so that why in order to avoid such circumstances our system TMS will provide the auto-integrated systems like in TMS there will be the facility of packages to book which will be sponsored by different agencies from which user can efficiently book and can surely help them to save their time and save them from spending hefty costs by going manually. In addition, the system majorly depends upon two perspectives one from the user end and one from the administrator end. Like the administration will have the authority to accept or reject the request, to give the booking and updating the data or records etc. However, from user end, the user will be able to do multiple functionalities like add, delete, book, choosing destinations and etc. to book his/her details to register the package moreover the business requirements for our system are categorized below:

|  |  |  |  |
| --- | --- | --- | --- |
| **Business Objectives** | **Risk** | **B. Opportunities** | **Success Criteria** |
| **BO1-** In order to minimize the time consuming factor | **R1-** As the agencies will charge some amount in the total bill so that from this the user can be dissatisfied and the product rating can be decreased. | **BO1-** Saving the amount of certain $ on every hotel registration. | **SC1-** If the 70% of the peoples uses the application at its initial phase, then this would make the system rating higher. |
| **BO2-** The system shall maximize the user interaction and communication as compare to physical one. | **R2-** The poor quality code can affect the overall requirement of the system. | **BO2-** Making the market by achieving the rating of 4 above during its consecutive 3 releases. | **SC2-**The efficiency of the system enhanced after the first release by 50% |
| **BO3-** Saving the x$ on every hotel registrations. | **R3-** System will be capable of handling the maximum workload but if many people come to system at a single time then the program may crashes. | **BO3-** Increase in gross margin and saving up to 10% on every registration. | **SC3-** The rating of the system will increase by 50% if the 50% of the user register on the app by latest version. |
| **BO4 –** In order to achieve optimization and gross increase the amount by 100$ after 1 year if the registrations goes above 80%. | **R4-** using the system on non-spec device can cause frustrations and this could lead to decrease the rating of the system. | **BO4-** Increasing the efficiency and usability of the system during the frst four releases in order to make the progress stable and achievable.. | **SC4-** Registration goes above 80% the in 2 months giving the margin of X$ after the first 2 releases of the system. |
| **BO5 –** Minimizing the risk factors | **R5-** System should be able to perform at multiple directories if user tries to distribute his data. | **BO5-** Saving amount and profit on each registration. | **SC5-**Increase in traffic on the app under first 4 months |
| **BO6 –** With the help of system the data handling of big data will be efficient and optimized. |  |  | **SC6-**Application download and rating increase exponentially by months in a year |
| **BO7 –** The system will be able to reduce the amount of procedural task as compare to physical one. |  |  |  |
| **BO8 –** cost efficiency |  |  |  |
| **BO9 –** Availability function increased as compare to other so that well understanding occur. |  |  |  |
| **BO10 –** Multiple agencies providers will be provided to the users which will minimize the problem of single agency. |  |  |  |

**ECO-system:**



RT: registration tracking’s

BD: Hotel and trips bookings

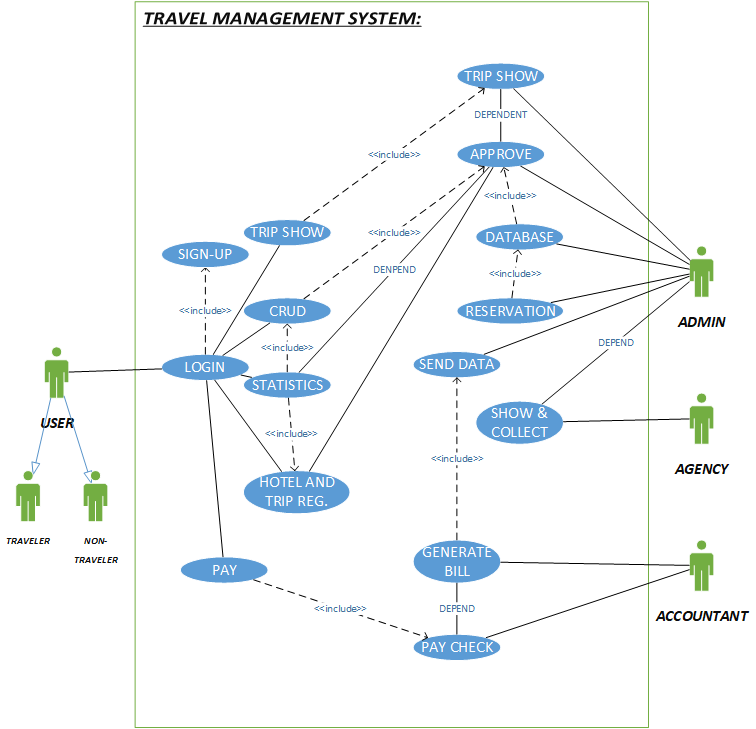
## Product Functions:

As the **TMS** is an end-to-end encrypted travelling system which will help the users to book their destinations and help the agencies to manage and connect the users(travelers) for booking, payments, providers detail etc. the system will give the user the better experience virtually as compared to doing the things manually. The system contains two main classes which is admin and user-ended class inter connected with the account and agency class as well. The functionalities of the system are basically categorized by the help of main classes as for interlinking classes the detailed overview will be provided in user classes and description (UCD). The main classes functionalities are given below:

1. **User Ended functions:**
   1. **The User Registrations:** The user can add the details for the registration. Basically it’s an end-to-end encryption between user details and database in order to keep user details secure.
   2. **Data update:** The user can update the data according to requirements and approvals from the admin. Moreover, the database should be checked and updated after the user perform registration or data entries.
   3. **Search the trip/package/location details:** The user can search for any specific key for trip/location/package. In response the system shall show him the appropriate details with correct providers and contact.
   4. **Feedback:** The system shall generate the feedback form according to the user according to the interaction with the system which can be filled up by the user.
   5. **Cancelation of a trip:** A user can cancel his trip. If a user wants to cancel his/her trip, then the system shall provide/send the information to admin so that his trips can be canceled.
   6. **Confirmations of booking:** A user can check the confirmations of his bookings for that thing the system shall send the confirmation to the user at their mail account/outlook account about the trip details.
   7. **paying the amount:** A user can pay the bill through secure method and after that the schedule of the trip of the selected package will be sent to user via mail.
   8. **Check statistics:** A user can the statistics after the approval from the admin.
2. **Admin ended functions:** 
   1. **Package track/statistics:** Admin shall be able to send the or update the user trip tracks/statistics after each action he/she perform on the application.
   2. **Package Creation:** Admin shall be responsible for creating the required package for the user and then generate a specific slip that will be sent to the user.
   3. **Ticketing Reservations:** The Admin shall check the details of user then he should send the slip of ticket reservation to the required user.
   4. **Update details:** After registering if user want to update his/her details then the Admin shall make the changes into the database and then send the update confirmation to user through system.
   5. **Slip generation:** Adminshall check the payments and send the details to account class so that he can generate the Slip of payment that shall be sent to the user which he/she can easily printout anytime.
   6. **Availability:** After successfully registering up the system the admin shall be able to send the details about all the tickets and flights etc. availability to the user with a unique ID.
   7. **Data base entries:** An admin will be able to add the user’s entries to data base.
   8. **Send data approvals:** An admin will be able to send the approvals and data to interlinking classes like agencies and accountant who will be responsible for the generation of user billings and the generation of desired packages.

**Figure 2:**

**The use case diagram to define he system functionalities is given on the next page:**



## User Classes and Characteristics

Our system **TMS** basically have different type of functionality for different type of user which basically in our system are [User, Admin, Accountant, Travel agencies]. The system will have two types main user classes, one is the user end class and the other from the administrative end. However, the other two inter linked classes which also plays a role in the project are agencies class and the accountant class. The detail for these classes are given below:

* + 1. **User class:** This class includes user details and a basic sign-up /log-in for the end user who want to add details and booking for the packages through specific/available filters. This class will be able to manage the data of the customers. The functions that this class will provide will be:

**2.3.1.0)** A user can register his/her id.

**2.3.1.1)** A user add packages.

**2.3.1.2)** A user can add the packages.

**2.3.1.3)** A user can cart the hotel features.

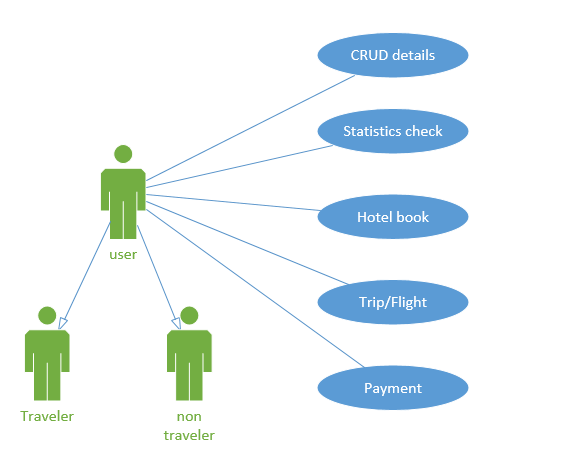
**2.3.1.4)** A user can book his trip/flight.

**2.3.1.5)** A user can check the details and statistics.

**2.3.1.6)** A user can get the ticket.

**2.3.1.6)** A user pay the bill.

**Figure:**

****

* + 1. **Travel Agency class:** The Agency class side will be link with the administrative class which (TA class) will send the packages details, updates, queries or tickets availability to the admins.

**2.3.2.0)** A agency member can upload the various packages according to filters.

**2.3.2.1)** A agency member can send the package details

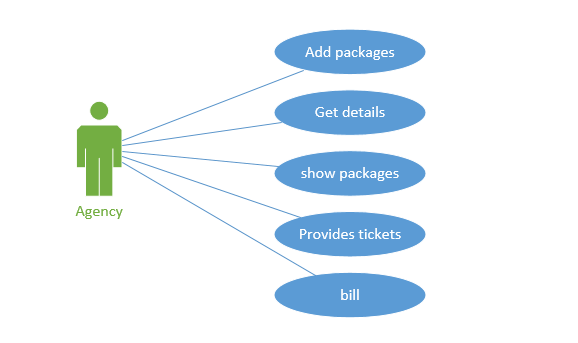
**2.3.2.2)** A agency member is responsible to provide the tickets amounts

**2.3.2.3)** A agency member can update the trip routes to admin

**2.3.2.4)** A agency member can set the prices on their packages

**2.3.2.5)** A agency member can provide multiple facilities with packages

**Figure:**



* + 1. **Admin class:** The administration class will help the user to make him updated and will manage the user data and their queries. Some of admin side function are given as below:

**2.3.2.0)** A admin can accept the approvals and send the confirmations.

**2.3.2.1)** A admin can change or update the details of a user for a package.

**2.3.2.2)** A admin can perform CRUD operations according to user provided details.

**2.3.2.3)** A admin can track the details of a user.

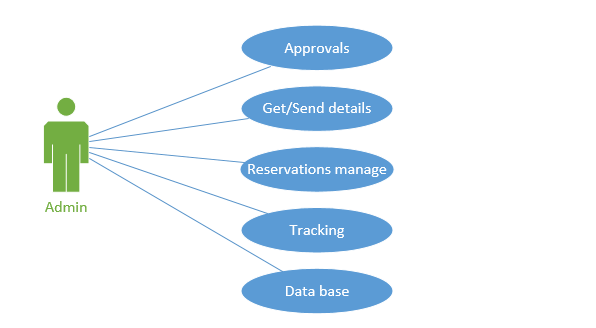
**2.3.2.4)** A admin can view the statistics and deadline of user trip and can notify.

**2.3.2.5)** A admin can perform security check validations for confirmations.

**2.3.2.6)** An admin can add the entries into the data base.

**2.3.2.7)** An admin can send the details to the agency and account class for generating the bills and packages for a specific user.

**Figure:**

****

* + 1. **Accountant Class:** An accountant class is a class which usually tries to generate the bills and payment setup for the user. The accountant class and generates the pay slips for a user. And all the things are generated by the approval of admin. The class functions for accountant are provided below:

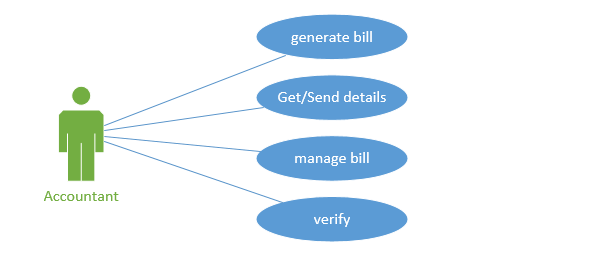
**2.3.2.0)** An accountant can check for payments.

**2.3.2.1)** An accountant can generate pay slips.

**2.3.2.2)** An accountant can add charges according to regions at the payment time and can manage all the payment records of the users.

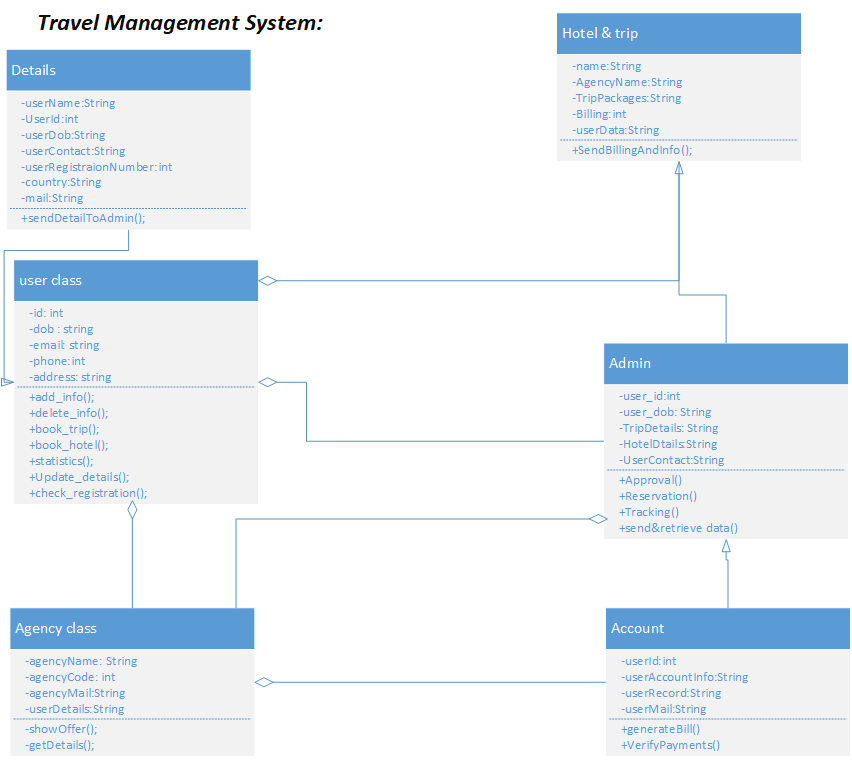
**2.3.2.3)** An accountant can check for dues and pay order link with admin.

**Figure:**

****

**FIGURE 3 (Class diagram for system):**

The elaboration of the classes for the system can be shown by class diagram which is given on the next page:



## Operating Environment

Basically the first three releases will mostly work on the desktop application so we will be launching the desktop version for the application that will probably work on the following operating systems with the hardware requirements of 50GB hard disk, 4Gb ram:

* + 1. Windows xp.
    2. Windows 07
    3. Windows 08
    4. Linux
    5. Windows 10/11
    6. Mac/MacOs

Further in future there will be an update of website version as well that will work on android, iOS and website version as well.

## Design and Implementation Constraints

The Design and implementation constraint for the TMS are given below:

* + 1. The System should be implemented through a specific and efficient language like java or kotlin
    2. The UI of the system should be consistent and should be responsive on all devices.
    3. Without internet connection the user cannot access the system. Hence its mandatory to have a stable internet connection.
    4. There should be at least 1 GB of ram on the system.
    5. Every user should be given a specific/unique user ID in order to prevent from unauthorized logins.
    6. As monthly system updates will happen after every 1 month so data should be saved before updating the system else the update will not occur.
    7. Except “google travel” no other APIs should be used while making the system.
    8. Web-based, Desktop-based or Smart-phone based system should use the camera access in order to capture the user details.
    9. There should be an inter linking between printer functions and the payment slip functions. So that user will be able to get the slips immediately.
    10. The system should have an OTP confirmation while login to the application.
    11. Application should be light and memory efficient that it will not cause the user to frustrate.
    12. There would be no more than 1-minute wait for any function that user will perform
    13. Payment feature should contain multiple payment options (visa, MasterCard, PayPal etc.)
    14. As per Government information the application should launch in specific regions first.
    15. The system should support bilingual language option with a simple Chabot.
    16. There should be a unique Id and password for each of the user that will be change on monthly basis.
    17. System should be connected through database in order to save information.
    18. System shall not charge any additional charges for national users.

## User Documentation:

Our system shall also provide the user documentations alongside with the software product for their ease to understand the system and its functionalities. The documentation will be available for both non-technical user and the technical user. Like for technical user (administration) the manual will be consisted with the operations to perform for booking adding or deleting the data or etc. similarly like for non-technical user the system will provide the different personas, manual for system start or tutorials for installing etc. as the documentation contains all the basic things through which user can get the idea for the system so user can easily get these thing in our TMS as well that is help, guidelines and use etc. There will different links be attached in the system through which user can get all the details, tutorials and etc.

Some of the documentations that user can found will be:

* User will be able to download the help manual from the help tab given on the system.
* There will be guided tutorials on the system page if the user is new.
* The help button with Chabot’s will be shown on main screens which will help the user if he/she stucks anywhere.

## Assumptions and Dependencies:

***2.7.1) Assumptions***: for our travel management system are provided below:

2.7.1.1) The compilation of code should be without the flaws

2.7.1.2) The data should be recorded into data base so that each user can access the record easily

2.7.1.3) The system should be memory efficient

2.7.1.4) Each user should have the unique id or password.

2.7.1.5) User-friendly and efficient design should be implemented in the system

2.7.1.6) User should not use any special characters while communicating with the system

2.7.1.7) The system should be workload efficient with more than 20 users else it should say the error of server down

2.7.1.8) The system should provide the customizability feature according to user.

2.7.1.9) The user can’t access the system after 12am.

2.7.1.10) The user shouldn’t be below 18 for registering into the system.

***2.7.2) Dependencies***: for our travel management system are provided below:

2.7.2.1) The system should be run on windows application means it will not work on different web services.

2.7.2.2) The database should be connected for the registration of the users

2.7.2.3) In order to run the system the device should have minimum specs.

2.7.2.4) The system shall start the process after the confirmations and approval from user.

2.7.2.5) In order to fetch packages the system shall access them from the multiple agencies system.

2.7.2.6) The admin system should be connected with both database and account/bank system in order to manage the things or data.

2.7.2.7) The payment should be done through PayPal or any secure service.

2.7.2.8) The system data should be save on cloud as well as a backup.

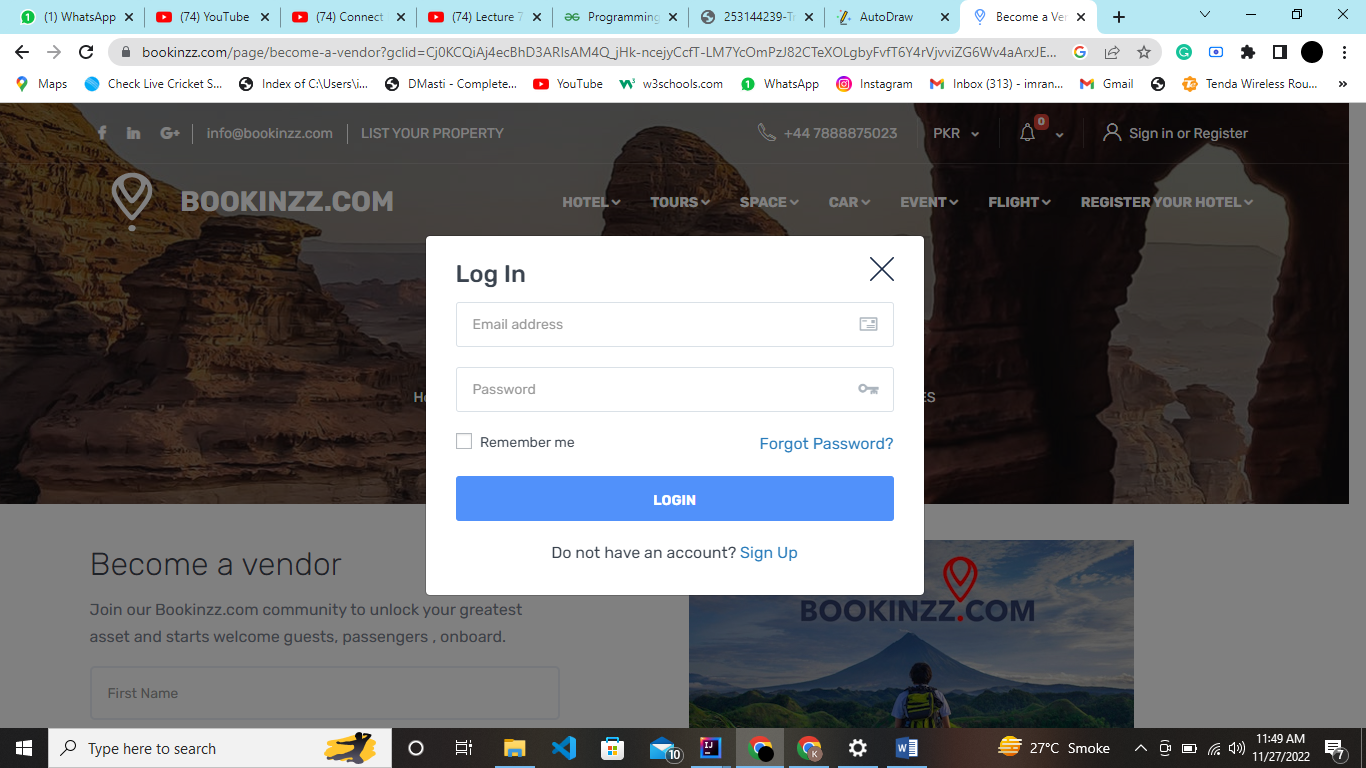
# *External Interface Requirements:*

## User Interfaces

The user interface in a system basically the show the overview of the system through the graphical interface pictures. Similarly, for our **TMS** the user interfaces can be described as below:

* + 1. **Login Window:**

The login screen for our system is attached below with the people name who interact with this screen.



**Users:**

*Customer:* From user perspective the user will simply just make the account and will proceeded to the home screen page after certain verifications.

*Agency:* The worker who is using the software.

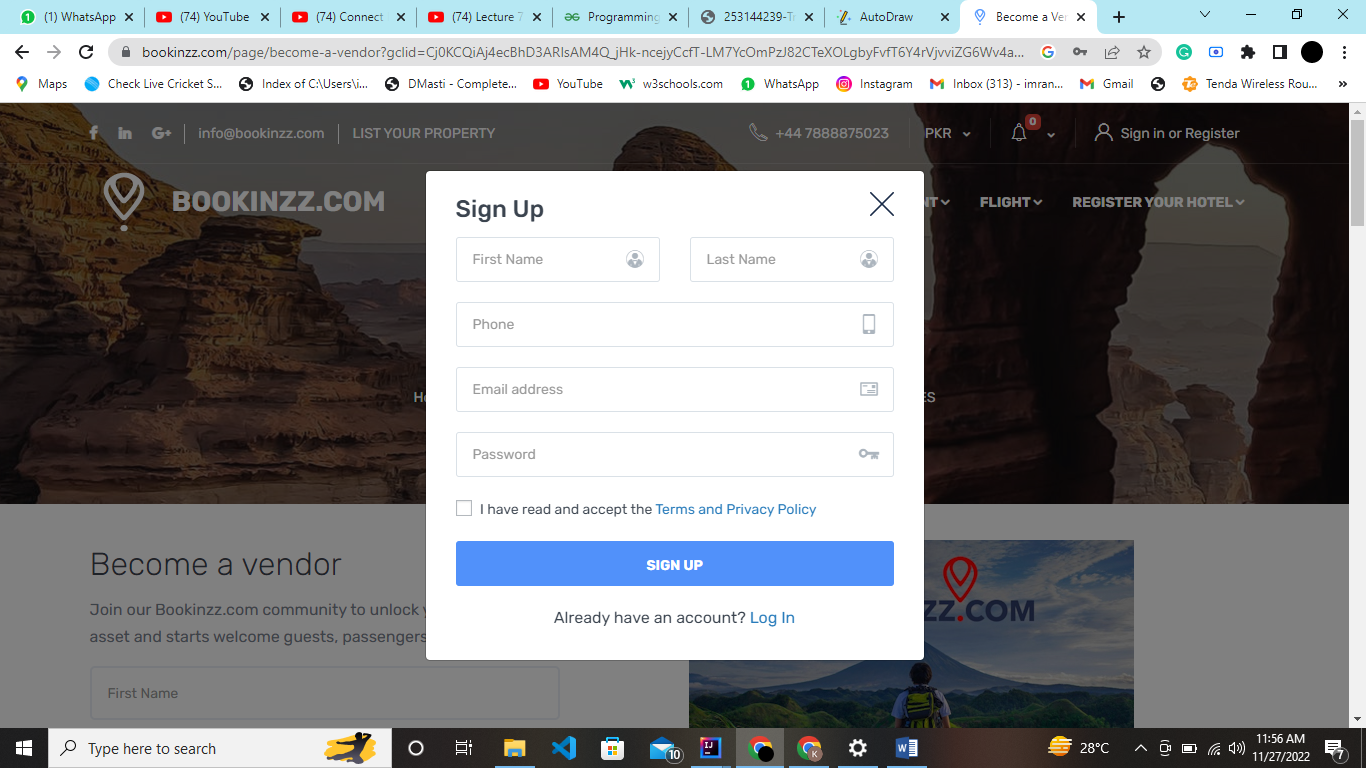
*Ethical Hackers:* The one with dealing the security concerns.

*Mangers:* Manager of company.

**Properties:**

* The login page shall be designed on the structure of safety and security requirements like no other user can access any others info.
* It includes two bars including username and password and two buttons of login and Signup.
* By pressing Signup button, customer can register by giving his data.
* After giving right email and password, login button will move to homepage.
* Having watch password to see the password.

1. **Signup:**



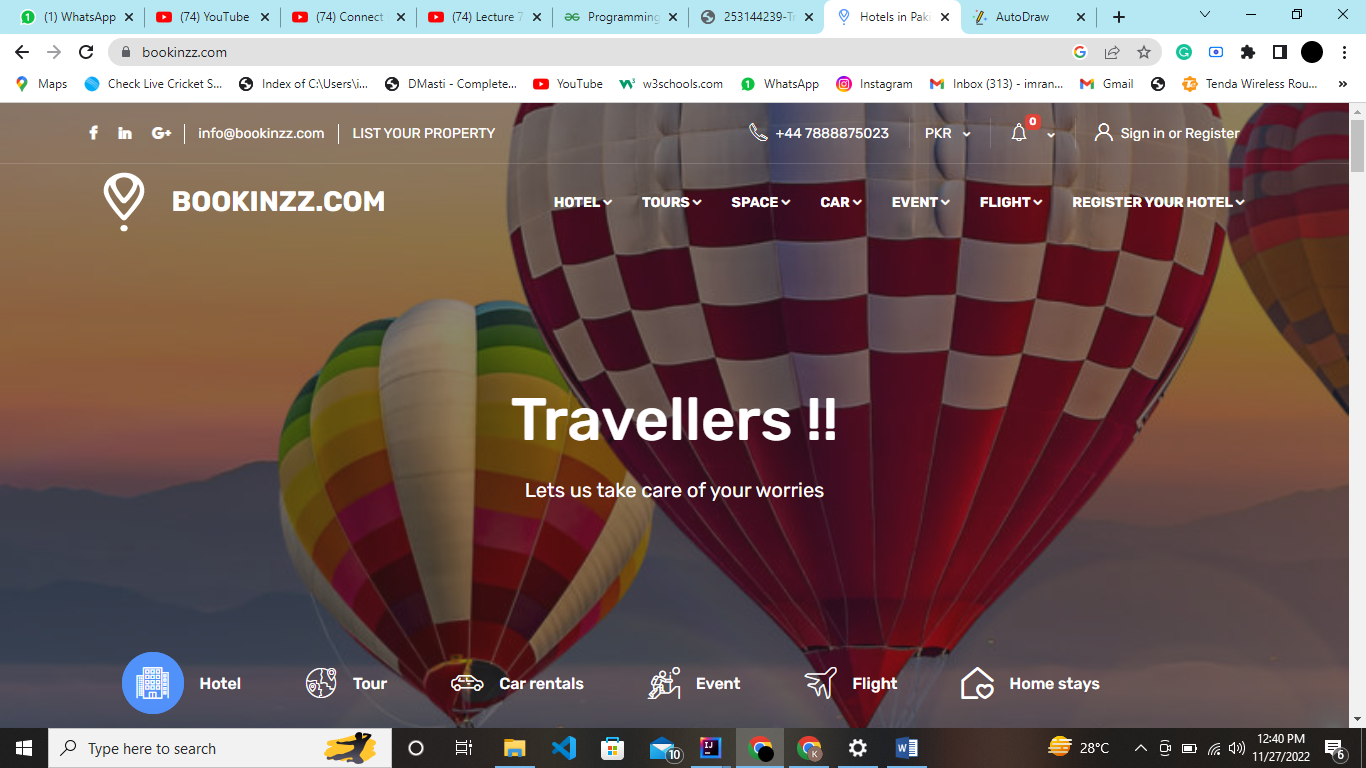
**User:**

The users for this one will include all the end-users, agency, admin and accountant etc.

**Properties:**

* This interface contains some text bars that will take information like first name, last name, phone number etc.
* Password bar will take password as dotted.

1. **Home page:**

****

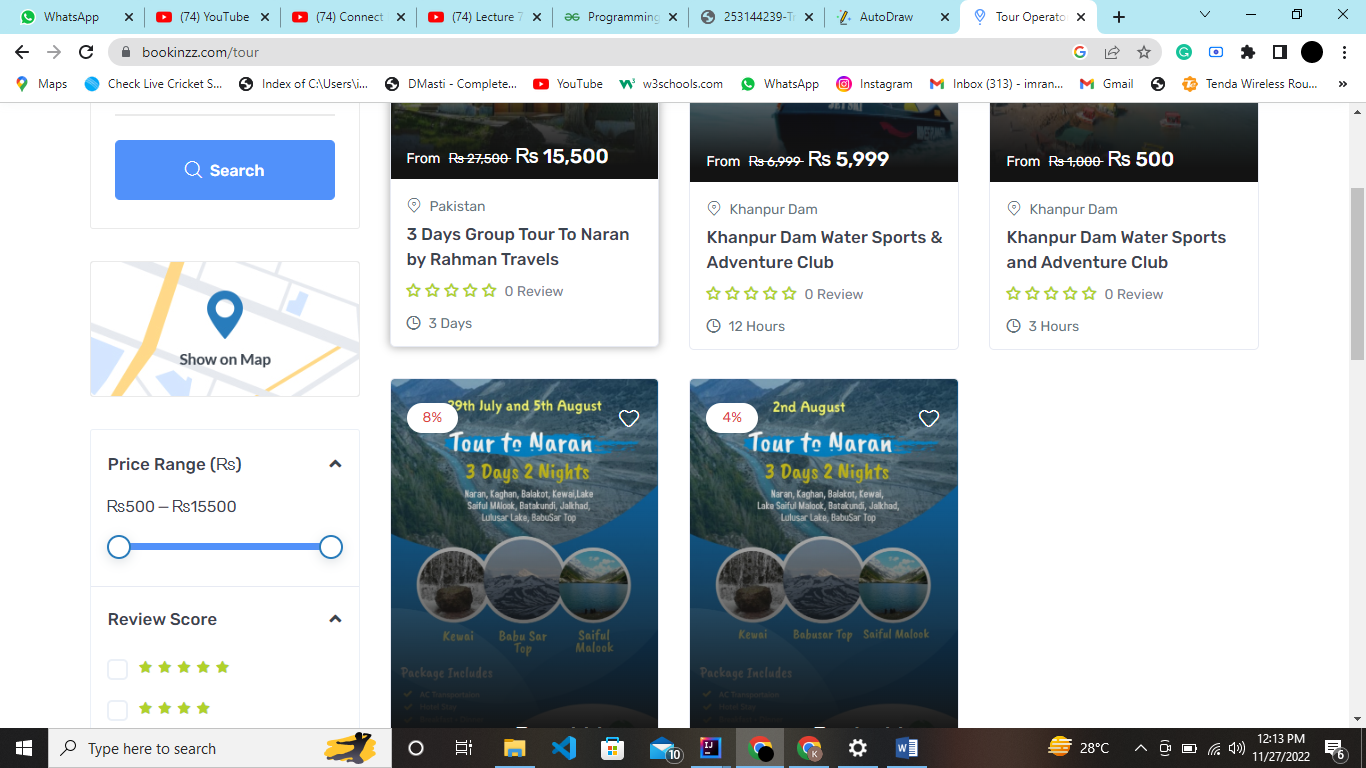
**Users:**

Customer, Hotel managers, Flight/Tours providers:

**Properties:**

* User can book hotel by pressing on hotel option.
* User can book a ticket for flight to particular area by pressing on tours option.

1. **Tours:**



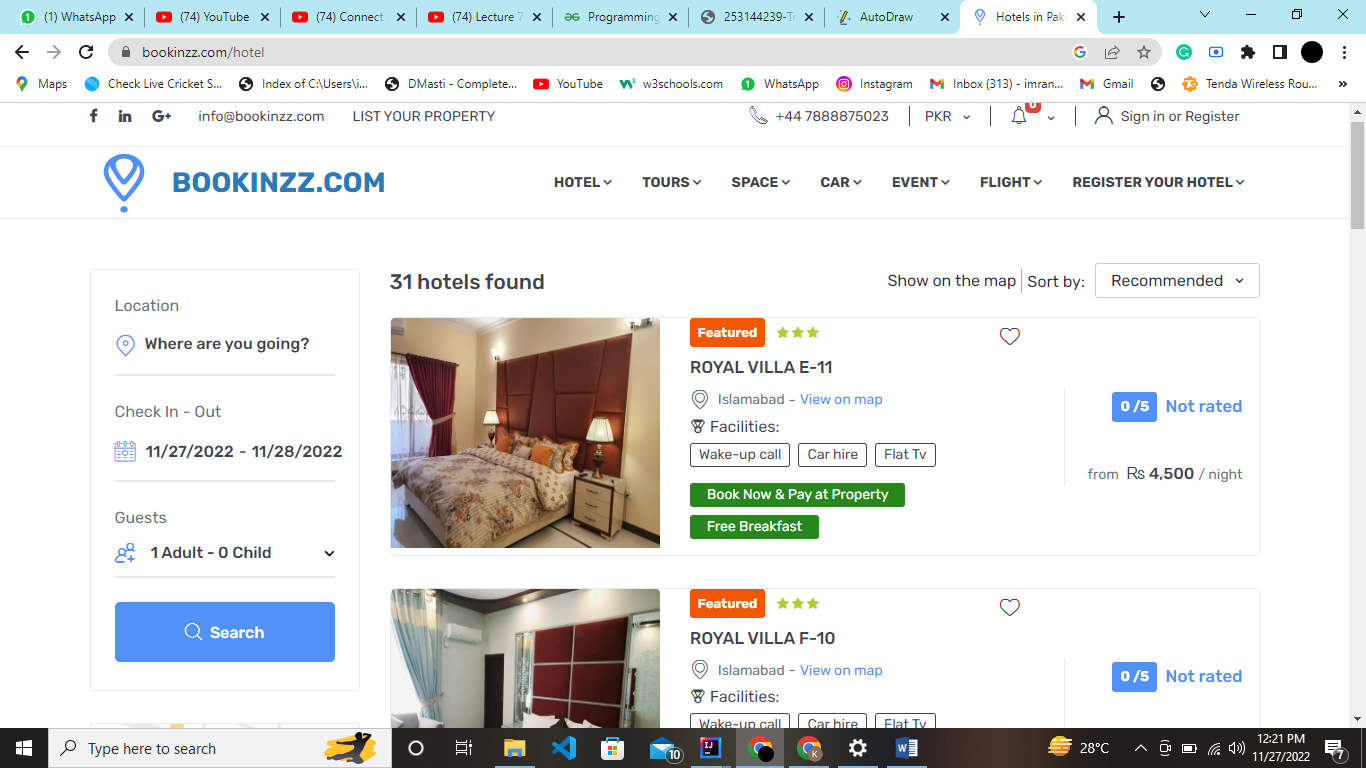
**User:**

Customer, Manager, De-bugger team: Always available for any bug appeared in System.

**Properties:**

* User can book Ticket for where he wants to travel.
* User can see price and with its discounts

1. **Hotel:**

****

**Users:**

Customer, agencies

**Properties:**

* Customer can select form many available hotel according to his budget.
* Customer can see rating about the hotel with facilities as well.

## Hardware Interfaces

The system will run on a computer with at least 256 MB RAM and with internet connection. The machine is connected to a printer for printing bills etc. The system will also available in android, in next update. Containing android version of 11 at least on it.

|  |  |  |  |
| --- | --- | --- | --- |
|  | | For PC | For MAC |
| Ram | At least 256 MB or more | | At least 256 MB or more |
| Internet connection | 500+ Kbps | | 500+ Kbps |
| Processor | 500 MHz or faster | | 1.8MHz intel core duo or faster |
| Bits | 16 bits or more | | 16 bits or more |
| Operating System | Windows XP of higher with latest updates installed | | Mac OS 10.3 or higher with latest updates installed |
| Video card | At least of 75 MB of video card | | At least 75 MB of video card |
| Versions | Windows 10 or more | | Version 10.7 or more |
| Memory | SSD | | SSD |

## Software Interfaces

The software uses windows and operating system. This software uses SQL database for data of flights, tours and hotel. GUI will be used in this software.

The software has different types of interface like,

* Graphical user interface

As system has most of interfaces has a visual way of interacting with user graphically.

* Menu-driven interface

As system has many menus in it like hotels, tours and etc.

* Form base user interface

Many forms are being used in system like login page, signup page and etc.

* Touch user interface

User interact with system using touching.

## Communications Interfaces

* + 1. The software must be web browser enabled and HTTP protocol is used to transfer data.
    2. Verifications should be done through mailing process through secure connections.
    3. Web interlink with giving the feedback forms.

# *4.* *System Features:*

It is very important to effectively understand the features of TMS as we are traveling internationally or domestically. Below are some safe and economically affordable features which need to be implemented by developers of the respective system known as “Travel Management System. The various feature for TMS are provided below:

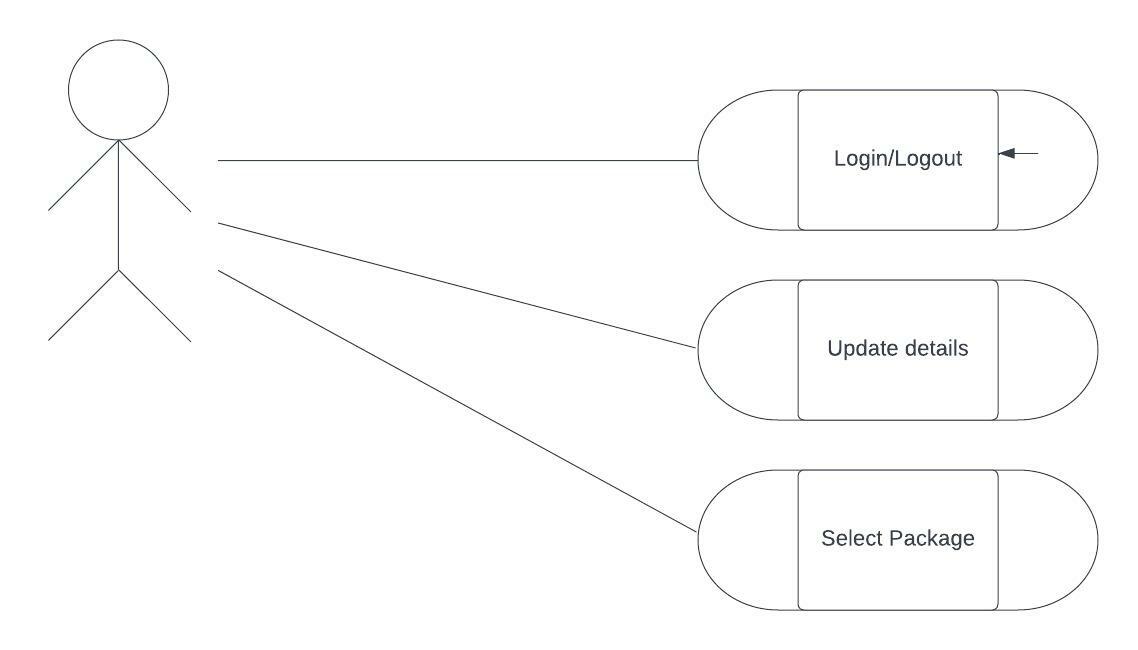
**4.1. Customizable and feasible setting/interface:**

**4.1.1. Description and priority**

The system needs to prioritize the smooth and friendly user interface to ensure perfect operations between the user and the system. For example, if a user wants to login or register themselves then they usually prioritize the smooth functionality of the system.

Hence, the priority is high for this feature.The rating of this feature is 8 on the scale of 1-9.

Suppose the following use case for **the above feature in the below diagram.**



**Figure 1: Use-case diagram for user-friendly interface**

**4.1.2. Stimulus/Response sequences**

As this feature mostly deals with the interface attribute so the response and stimulation will not be much longer. The stimulation of a system needs a quick response and stimulation of a system should be maintained thoroughly in the interface according to the user commands.

**4.1.3. Functional Requirements**

* The theme of the system can be modified.
* The system shall show the pop ups to the user whenever he updates the system setting or do anything with the system.
* The system shall support generalizability and task migrate ability like whenever user tries to communicate the system should interact with him friendly through minimal dialogue boxes or chat bot queries.

**4.2. Online payment options (As system supports on branch payments):**

**4.2.1. Description and priority**

We can expect easy and upgraded payment options other than cash from TMS while traveling like QR codes, UPI IDs, online banking, debit, or credit cards as users now usually pay via cards or online banking. Hence, the priority is high for this feature in the system. The rating of this feature is 7.5 on the scale of 1-9.

We can also find out the priority of the system through the following user story.

*“On payment information, I need to know how to pay for my trip If there are any fees and fares required (and how I will be refunded for failures in services). If there are incentives or restrictions and there are other ways to pay for fares or fees if I don’t have a credit card, a smartphone or internet access.”*

**4.2.2. Stimulus/Response sequences**

Online payment can be accessed after filling in the personal details in the system. Then, the system will respond quickly and take the user to online payment options after clicking the payment tab.

**4.2.3. Functional requirements**

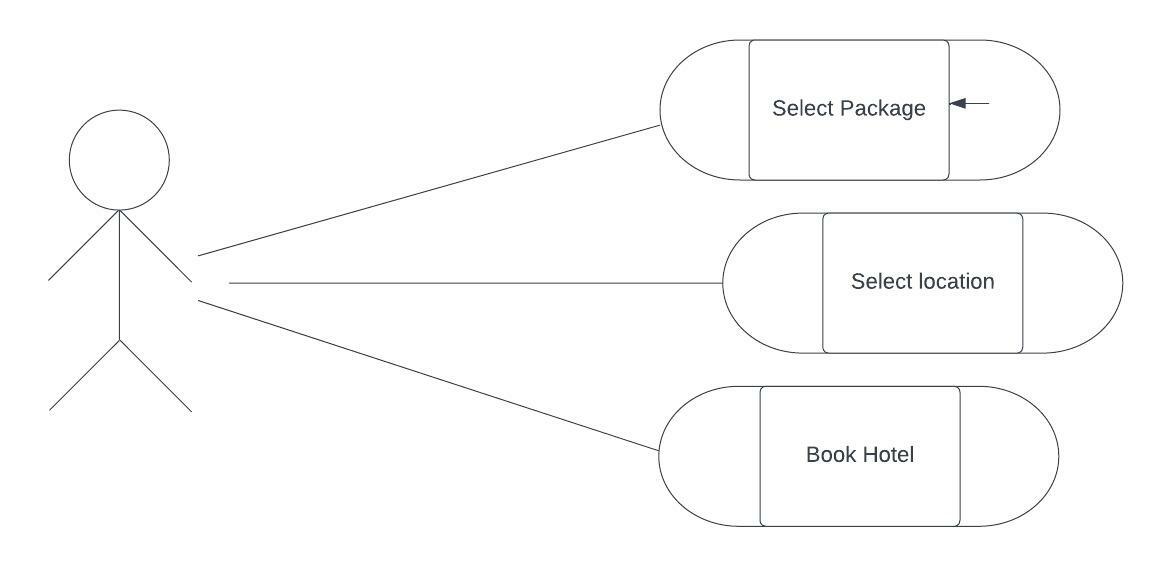
* The payment confirmation message would be sent to the user’s email address and contact number that was filled in by the user before in personal details.
* Dialogue pop-up with error message will appear in case of any failure of payment.
* The system will send the email to the user before acceptance of payment to let the user know about the transaction happening through his/her account.

**4.3. Wide inventories/services availability:**

**4.3.1. Description and priority**

Maintaining a large inventory (large number of hotel/car/ bus vendors, agents) in travel management systems is most beneficial. If you don’t have enough availability of low-cost accommodation, it may tend towards a loss of great market exposure eventually resulting in loss of business.

Hence, the priority for this feature is also higher. The rating is 7 on the scale of 1-9.



**Figure 2: Use-case diagram for wide inventories**

**4.3.2. Stimulus/Response sequences**

The user can have multiple choices in the hotel booking tab in which all the stimulated options are present with wider choices. The system is responsive enough to support the wide inventories within the system.

**4.3.3. Functional Requirements**

* System will showcase every option i.e., packages, billing, and hotels available after selecting the destination.
* The price of every facility would be displayed after selecting the destination.
* A price calculator will also appear to calculate the estimated amount.

**4.4. Breadth choice packages for users (Robust Software)**

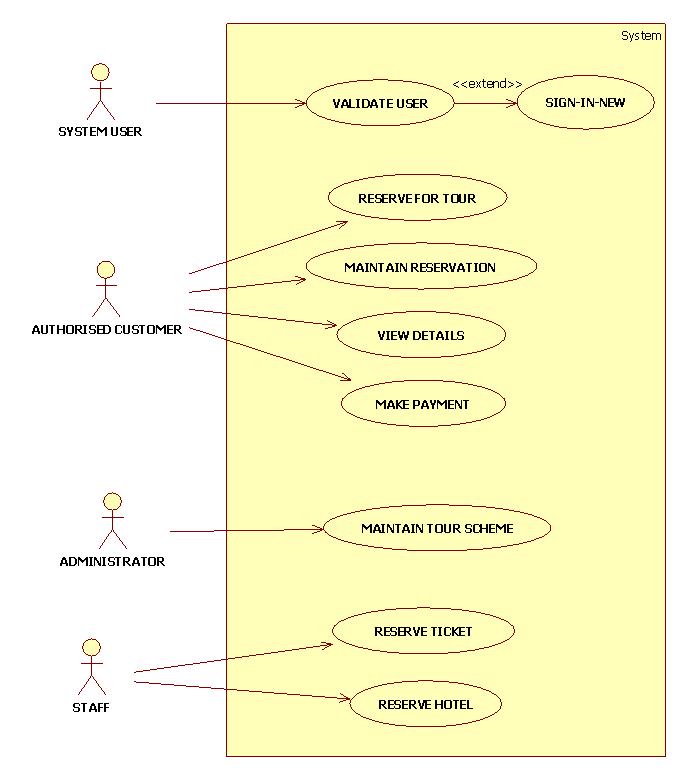
**4.4.1. Description and priority**

The business of the travel industry totally depends on festivals, holidays, and different packages. There are a few days where these travel solutions may face huge traffic, so every travel agency should have a robust system to handle the traffic. Major operations should be run smoothly, and the system should not crash.

The priority of this feature is 9 on the scale of 1-9. Hence, it should be prioritized in a much higher position than other features.

*Have a look at the figure 3 on the next page*

**Figure 3: This use case shows multiple users accessing the system at the same time.**



**4.4.2. Stimulus/Response sequences**

In case of multiple user interventions at the same time, the system should be stimulated with robustness and the response sequences should be same as before.

**4.4.3. Functional Requirements**

* System should respond to every user at the same time with correct information.
* If the system crashes due to workload, then the system should auto save the details filled in by the user.

**4.5. Budget Friendly**

4**.5.1. Description/Priority**

Watch out for hidden extras that have the potential to eat into the cost-effectiveness of the system, such as charges on accommodation, amendments to bookings, or support requests. Where possible, choose a business travel booking tool with a transparent pricing structure.

The charges should be affordable, and the priority of this feature is medium. The rating of this feature is 6 on a scale of 1 to 9.

*“As a tourist, I want to invite my friends to travel to the different parts of the world but the travel agency are bit too expensive nowadays as they look for personal profits as well, the travelling packages should be budget friendly which can attract many tourists and can increase the business growth as well.”*

**4.5.2. Stimulus/Response sequences**

The system should display the pricing according to the destination selected by the user and special discounts should be stimulated on different occasions like ‘Eid, Christmas, and Diwali etc.

**4.5.3. Functional Requirements**

* The total amount of the bill will be shared to the registered email and contact number of the user.
* Special discounts would be advertised to let the user know about the latest deals and offers happening on Travel Management System.
* The promotion code and coupons will also be offered on special occasions.

**4.6. Customer care support**

**4.6.1. Description and priority**

If a customer has any issues related to any bookings or reservations or during travel or after travel, customer care support comes in the picture here. The travel agency should provide strong, polite, and supportive services to their customers so that the customers are satisfied with the services provided and the company is happy with the customers’ feedback in order to grow the business.

Since customer care support is really essential in order to gain the trust of customers the priority of this feature is higher and has a rating of 8 on the scale of 1 to 9.

*“As a frequent flyer, I expect to the system’s customer care staff to provide sufficient services in case of any difficulty happening to me in the usage of Travel Management System.”*

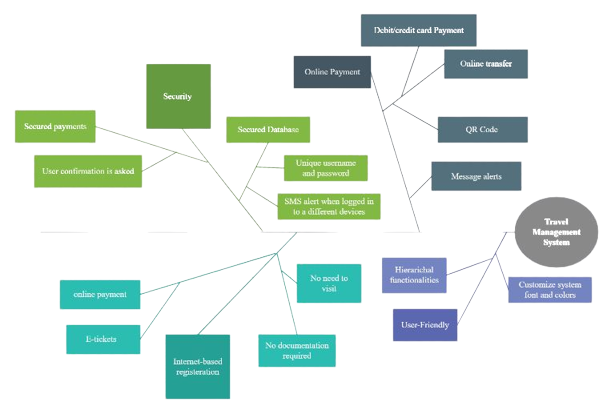
**4.6.2. Stimulus/Response Sequences**

The response rate of customer care support should be quicker enough for any type of query and the staff should have enough stimulation towards the customers.

**4.6.3. Functional requirements**

* The system will show a dialogue box to ask for the customer’s feedback at the end of the process.
* A message would be sent to the customer’s registered email and contact number when his/her problem is resolved.
* A Chabot will be integrated to offer quick help to the recent queries.

The understandability for the feature of the system can be also shown by the help of feature tree diagram which is provided on the next page:



**Figure 4: Feature Tree diagram for the remaining system features**

# *Other Nonfunctional Requirements:*

There are many non-functional requirements that have been discussed below:

* 1. **PERFORMANCE REQUIREMENT:**

**5.1.1)** The user should not wait for a long time after performing some task. He will get the response as soon as possible.

**5.1.2)** The system should contain an interactive and user friendly interface and the functionalities that have been used in the system, should be easily understandable by the user.

**5.1.3)** If the user faces some type of difficulties/queries while using the system, so there is a help desk where he can resolve his queries with in minimum time.

**5.1.4)** The system should be effective, so the user should not feel any difficulty while using the system.

**5.1.5)** A transition from one window to the next window should be smooth**.** The user can customize the look of transition effects.

**5.1.6)** The system should be flexible and maintainable, so that in future certain changes will be made and can be maintained.

**5.1.7)** The system should be efficient in the way that it uses less memory and stores a large amount of data and can be able to manage large amount of data.

**5.1.8)** The system should be effectively handle the work load and can be able to handle the situation if multiple users are visiting our application.

**5.1.9)** If a specific glitch or the error is occurred in the system, the system should be able to handle it for a specific period of time without any disturbance and interruption.

**5.1.10)** The system should be able to manage all the data and data files that are stored at a particular location.

**5.1.11)** The system should be able that it should not disclose the user' personal information except his name to any other user.

* 1. **SAFETY REQUIREMENT:**

**5.2.1)** The password should contain 8 characters i.e. 2 upper case letters, 3 lower case letters, 2 numeric letters and 1 especial letter.

**5.2.2)** If the user wants link with another system the system should ask for the user confirmation whether he wants to share his data or not.

**5.2.3)** In the system there should be exception handling for any of the error occurred when the system is running.

**5.2.4)** In the system there should be a backup or recovery system if the user performs some wrong action by mistake, so he should have an option to go back or recover.

**5.2.5)** The system should not use the same ID of two persons, it should use a unique ID for each person’s authorizations.

**5.2.6)** The system should not allow the third party actions to make the system safe.

**5.2.7)** The information that is given by the user should be protected enough for user’s verification.

**5.2.8)** If the user wants do the payment through credit card the system should check that whether the user have enough money or not.

* 1. **SECURITY REQUIREMENT:**

**5.3.1)** The system should have the property of confidentiality; the system should secure the access to user’s data and personal details.

**5.3.2)** There should be a two-way process between the user and the system while login (the user while entering his username and password it should tell that whether it is right or wrong).

**5.3.3)** The should check that the user is registered/authorized, if he is not registered user then the system should the user to register yourself.

**5.3.4)** The system should ensure that the user’s personal data should not be leaked and also ensure that the data should not be misused.

**5.3.5)** The data that is entered by the user while creating an account, it should be secured by the system.

**5.3.6)** The system ensures that if the user is unregistered so he can only read the data and cannot write the details

**5.3.7)** There should be a base line and monitoring for changes, if the user is doing some prohibited action then the alert message should be generated to the control room.

**5.3.8)** The system should ensure that the database record should not be displayed to other

users except the administrator and management team.

**5.3.9)** The system should secure the payment transection and maintains the record in the database.

* 1. **SOFTWARE QUALITY ATTRIBUTE:**

**5.4.1)** **Usability:** The system should be user friendly like when the user uses the system for the first time, he will not face any type of difficulty to use the system and the system should be maintainable i.e. if the customer wants to make certain changes in the system so the system should accept that changes.

**5.4.2)** **Efficiency:** The system should be capable enough to manage work load like when the user performs some task so the system should complete that task in few seconds without any delay.

**5.4.3)** **Reliability:** The system should ensure that when the user performs some action so some output should be come rather than system fails that action.

**5.4.4)** **Integrity and testability:** The system should be capable to handle/stop unauthorized logins and some illegal and malware actions perform by the user and send the message of alert to the control room of the system.

**5.4.5)** **Portability:** The system should be portable and can work on multiple-platforms without any disturbance and lagging. It also includes the behavior and cost of implementation of the system at any particular area.

**5.4.6)** **Learnability:** The system should ensure that the function and design of the system should be efficient so that user can easily judge and use the system without any delays.

**5.4.7)** **Maintainability:** The should be capable to accept upcoming versions and allows the user to make changes within the software.

* 1. **BUSSINESS RULE:**

*A business rule is guidance that there is an obligation concerning conduct, action, practice, or procedure within a particular activity or sphere.* In our travel management system, the business rule that can be implemented can be described as follows:

There are five types of business rules:

1. Facts
2. Constraints
3. Action Enablers
4. Inferences
5. Computations
   * 1. **Facts:**
6. Every member should have a unique id
7. There will be a Unique Serial Code given to every user from which they can get discounted coupon.
8. Tax will be charged if user is outside the country
9. Generation of payment slip will be shown after completing all the prerequisite.
10. In order to access system internet connection is mandatory.
11. Additional charges will be for any facility user want in his package.
12. OTP confirmation while login and doing payments.
13. Email should be unique
14. Password should be unique
15. Tickets should be reserved after checking the id and email of each user.
    * 1. **Constraints:**
16. Without “CNIC” and “E-MAIL” user can’t add details
17. System shall not calculate the payment without unique id
18. At a time, user can only book for 1 person
19. No user can select more than 1 package
20. Without backup user data can be lost
21. External users can only read the system state
22. Without the id and number, the details can’t be added
23. Only admin can accept the approvals for package booking
24. Without OTP a user can’t login into
25. User should have credit card in order to make purchases
26. Except the week days the user can’t book the package.
27. User will not be able to add more than 2-3 tickets for a journey.
28. Only admin can add, approve, update, track the details of a user.
    * 1. **Action Enablers:**
    1. If the all the flights that user want is not there or full then send a feedback to user
    2. If the user serial code is expiring in few days, then he should request so that new serial code should be given
    3. Show different agency packages related to user range that he entered
    4. If payment has been submitted, then show the slip.
    5. If user request for registration has approved, then send him different details from he can book or add
    6. Email configuration should be sent if the user is signing up for the first time on the system
    7. Show the flight details to the customer after according to the packages and price
    8. If user enter the correct OTP, then the system should send him to registration page
    9. If the user has completed all the details, then system should send a feedback form for review.
    10. If the user has applied the discount code, then system should not generate it again for next 1 month
    11. User can delete/cancel the details or registrations if the data that he enters matches with the database.
    12. Notification should be sent after the payment
    13. Authorization configuration should be send to user after doing all the steps.
        1. **Inference:**

**1**. If the user doesn’t add her details for the next 2 weeks after signing up, system should consider his id delete.

**2**. The serial code will be considered as expiry if user don’t use it between the period of 1 month

3. If the user does not pay the bill, then system should auto ban him

**4**. If selected price from user is high, then system should be considering him to follow business class packages else should considering to follow middle class packages.

5. If admin is not responding to user request, then user should consider to try again later

6. The system will cancel the flights or packages if the given data is wrong on authorization.

7. System should consider a user a black-user for year if he/she not pay the bill.

8. If user can’t login to the system, then the system should consider him/her a new or fresh user/customer.

9. If the serial code for the coupon is applied, then the system should show it as expiry or limit reached or coupon already applied.

10. If the user id is already existed on the system for more than 6 years, then admin/system should consider him as premium user

* + 1. **Computations:**

The computations through which the user will be able to see the various amount on his package for the system which basically are derived from some formulas can be defined as follow:

* + - 1. The total bill of the user should be generated from the sum of the number of ticket with the total amount of packages he bought.
      2. The package activity percentage of a certain user shall be calculated by total number of activities divided by total number of days spent multiplied by hundred. As a result, this will generate the activity percentage for the user.
  1. **BUSSINESS RULE CATALOG:**

The overall catalog for the system can be defined as below:

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Rule of definition** | **Type of rule** | **Static OR Dynamic** |
| BR-001 | Every member should have a unique id | Fact | Static |
| BR-002 | There will be a Unique Serial Code given to every user from which they can get discounted coupon. | Fact | Static |
| BR-003 | Tax will be charged if user is outside the country | Fact | Static |
| BR-004 | Generation of payment slip will be shown after completing all the prerequisite. | Fact | Static |
| BR-005 | Internet connection is mandatory to access the system. | Fact | Static |
| BR-006 | Additional charges will be for any facility user want in his package. | Fact | Static |
| BR-007 | OTP confirmation while login and doing payments. | Fact | Static |
| BR-008 | Email should be unique. | Fact | Static |
| BR-009 | Password should be unique. | Fact | Static |
| BR-010 | Tickets should be reserved after checking the id and email of each user. | Fact | Static |
| BR-011 | Without “CNIC” and “E-MAIL” user can’t add details. | Constraint | Dynamic |
| BR-012 | System shall not calculate the payment without unique id. | Constraint | Dynamic |
| BR-013 | User can only book for 1 person at a time. | Constraint | Dynamic |
| BR-014 | User cannot select more than 1 package. | Constraint | Static |
| BR-015 | The details cannot be added without the id and number. | Constraint | Static |
| BR-016 | Only admin can accept the approvals for package booking. | Constraint | Dynamic |
| BR-017 | Without OTP a user cannot login into the system. | Constraint | Dynamic |
| BR-018 | User should have credit card in order to make purchases. | Constraint | Dynamic |
| BR-019 | User cannot book the package except the week days. | Constraint | Dynamic |
| BR-020 | User will not be able to add more than 2-3 tickets for a journey. | Constraint | Dynamic |
| BR-021 | Only admin can add, approve, update, track the details of a user. | Constraint | Static |
| BR-022 | External users can only read the system state. | Constraint | Static |
| BR-023 | If all the flights that user want, is not there or full then send a feedback to user. | Action Enabler | Static |
| BR-024 | If the user serial code is expiring in few days, then he should request, so that new serial code should be given. | Action Enabler | Static |
| BR-025 | Show the slip when the payment has been submitted. | Action Enabler | Static |
| BR-026 | If user registration is approved, then send him different details from where he can book or add. | Action Enabler | Static |
| BR-027 | Email should be send to the user if the user is signing up for the first time on the system. | Action Enabler | Static |
| BR-028 | The flight details of the customer should be shown according to the package he choose. | Action Enabler | Static |
| BR-029 | If user enter the correct OTP, then the system should send him to registration page. | Action Enabler | Dynamic |
| BR-030 | If the user has completed all the details, then system should send a feedback form for review. | Action Enabler | Dynamic |
| BR-031 | If the user has applied for the discount code, then system should not generate it again for next 1 month. | Action Enabler | Dynamic |
| BR-032 | User can delete/cancel the details or registrations if the data that he enters matches with the database. | Action Enabler | Static |
| BR-033 | Notification should be sent after the payment. | Action Enabler | Static |
| BR-034 | Authorization configuration should be send to user after doing all the steps. | Action Enabler | Static |
| BR-035 | If the user does not add his details for the next 2 weeks after signing up, system should consider his id to be deleted. | Inference | Dynamic |
| BR-036 | The serial code will be considered as expired if user do not use it between the period of 1 month. | Inference | Dynamic |
| BR-037 | If the user does not pay the bill, then system should auto ban him. | Inference | Static |
| BR-038 | If selected price from user is high, then the system should be considering him to follow business class packages else should considering to follow middle-class packages. | Inference | Dynamic |
| BR-039 | If the admin is not responding to the user’s request, then user should consider trying again later. | Inference | Dynamic |
| BR-040 | The system will cancel the flights or packages if the given data is wrong on the authorization. | Inference | Static |
| BR-041 | System should consider a user a black user for a year if he/she does not pay the bill. | Inference | Static |
| BR-042 | If the user cannot log in to the system, then the system should consider him/her a new or fresh user/customer. | Inference | Static |
| BR-043 | If the serial code for the coupon is applied, then the system should show it as expiry or limit reached or coupon already applied. | Inference | Static |
| BR-044 | If the user id already exists on the system for more than 6 years, then admin/system should consider him as a premium user. | Inference | Dynamic |
| BR-045 | Ticket price + number of days stay + sales tax | Computation | Static |
| BR-046 | A package the hat user chooses + sales tax | Computation | Static |
| BR-047 | Ticket price + number of days stay + sales tax + penalty(if the user paid after the due date) | Computation | Static |

# *Other Requirements:*

* 1. ***Data Base requirement:***
     1. The data base implementation will be through MySQL packages
     2. The data base should not contain the username of same type like occur the system should give them error.
     3. The data base should not exceed the table quantity so that data organization should be minimum.
     4. The data should be only created by the admin server side
     5. Static or dynamic both data should be acceptable in table.
     6. The data entries should be assigned in sorted and organized manner so that it would be easy to take out the details.
  2. **Internal quality requirements**
     1. **Efficiency:**
        1. The system should use less memory usage.
        2. The system should be capable of managing the big data at a single time.
        3. The system should be capable enough to handle the huge traffic.
        4. The system should give the pop up to user if he enters the wrong pin or the time of OTP exceeds
     2. **Supportability:**
        1. The system should manage the workload of the print copies if user try to print the copy of slip from his pc
        2. As the system support printer facility as well so the system should generate run time messages if the user stuck at some point
        3. There should be performance and efficiency functionalities use whenever user tries to give commands from his portal system.
     3. **Portability:**
        1. The system should support the application working on mac as well.
        2. After first two release the system will be switched for web and mobile based interface without changing much things
        3. The system shall build on cross platform products so that it would be easy to deal that system on other systems with changing its internal state again and again.
        4. There shall be an inter link with web browser as well so that there will be no problems while going for feedbacks or emails.
     4. **Scalability:**
        1. The system should be capable enough to perform according to the ram size.
        2. The system should add 10 tables after each 100 registrations in order to minimize load factors.
        3. System should customize itself according to the given user commands.
        4. Agencies should be added to user account according to their needs so that there will be maximum efficiency factor.

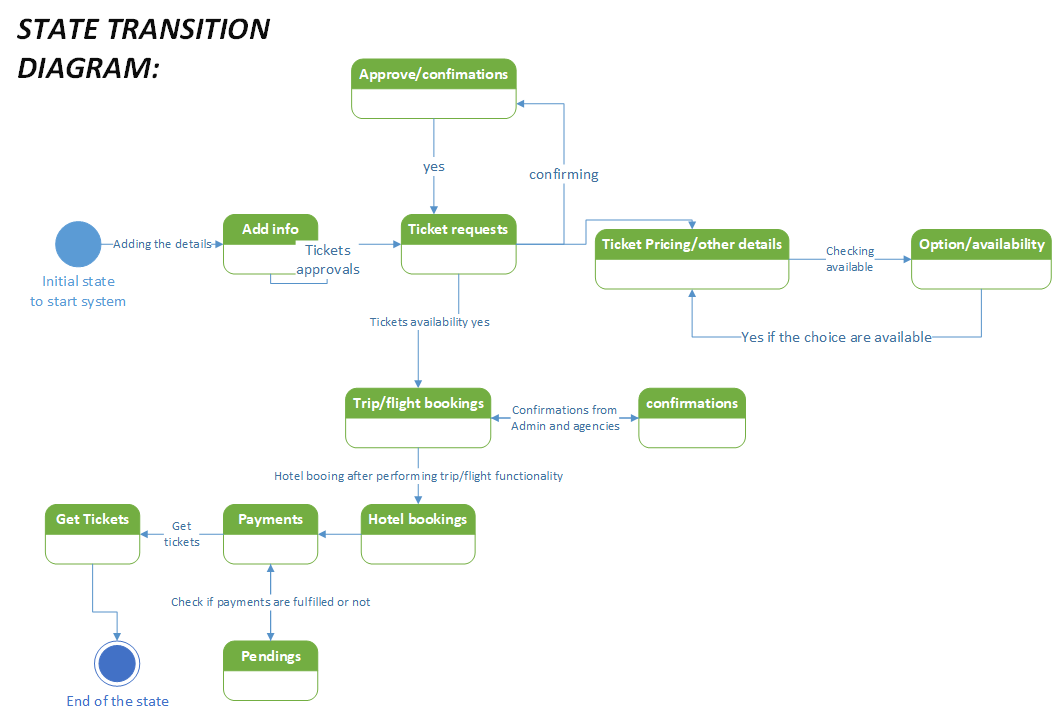
**Appendix A: Glossary**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Actors | Users connected with a particular product or system. |
| Stakeholders | The people who are linked with the building of the system |
| User persona | Description to define a particular user class. |
| OTA | Online travel agency |
| OBT | Online booking tool |
| TMS | Travel management system |
| TMC | Travel management company |
| VCN | Virtual card number |
| Documentation | A guide through which user can understand the system. |
| HB | Hotel brands |
| BR | Business requirements |
| SC | Success criteria |
| BO | Business objectives |
| Inference | Inferring to a thing, object or a fact |
| Business rule catalog | A table to define a statement about particular requirement. |
| Fish bone | A diagram to represent a purpose of a particular problem. Also known as cause and effect diagram. |
| user | A reviewer, author, developer or a reader. |
| dependencies | Factors on which a specific requirement of a function rely. |
| reviewer | Person who examine the article and has the authority to recommend or cancel the article. |
| Data base- DB | A collection of information monitored by the system. |
| ERD | Entity relation Diagram |
|  |  |

**Appendix B: Analysis Models**

The various diagram models for our system can are given as follows which basically means to understand the system better and clearly with the different diagram models.

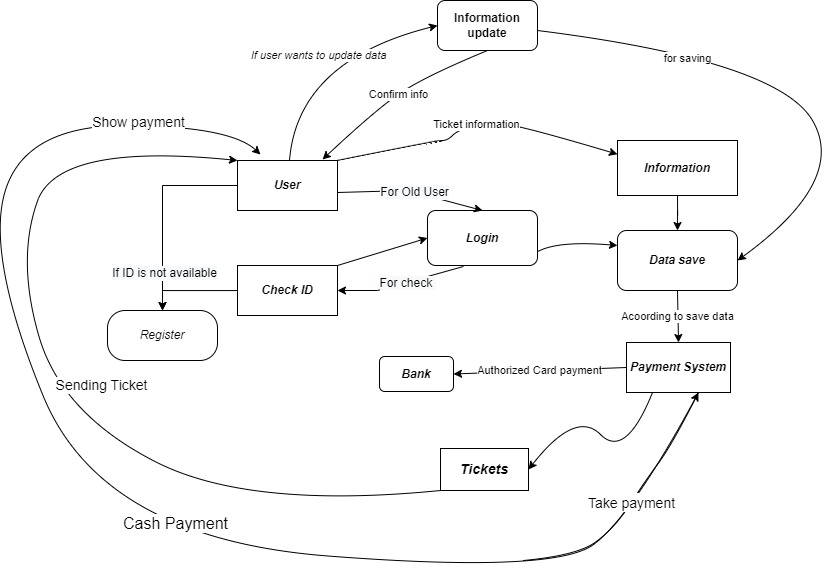
1. **STATE TRANSITION DIAGRAM:**

****

1. **Entity Relationship Diagram:**

****

1. **LEVEL ZERO DIAGRAM:**



**Appendix C: To Be Determined List**

1. **Data Structures for the system:**
   1. **Booking a hotel:**

**BOOKHOTEL** = HOTEL\_ID**+** HOTEL NAME**+**

HOTEL\_START\_DATE**+**

HOTEL\_END\_DATE**+**

**[**FIVE STAR HOTEL, FOUR STAR HOTEL**]** **+**

**[**HOTEL\_EMAIL, HOTEL PHONE**]** **+**

**(**HOTEL\_SOCIAL\_SITE\_MAIL**)** **+**

1**{**USER\_NAME**+**USER\_ID**+**

USER\_BILLING**+**

USER\_VERIFY**+**

USER\_REQUIREMENT**+**

USER\_NUMBER**}** N**+**

HOTEL\_DESTINATION**+** **(**HOTEL FEATURES**)** **+**CALCULATE\_BILL**+**

[PAYMENT CARD, BRANCH SUBMISSION] **+**

END

* 1. **Booking a TRIP:**

**BOOK\_TRIP** = TRIP\_NAME+ TRIP\_FLIGHT\_ID +

TRIP\_START\_ID+TRIP\_END\_ID+

[LOCATION A…. LOCATION Z]

+ 1{USER\_NAME+

USER\_ID+

USER\_BILLING+

USER\_CONTACT+

USER\_ADDRESS

} N+

(USER\_PREVIOUS\_RECORD) +

[FLIGHT A TO DESTINATION, FLIGHT B TO DESTINATION] + CALCULATE\_BILL**+**

[PAYMENT CARD, BRANCH SUBMISSION] **+**

END

* 1. **Payments:**

**Payments** = Username+ user\_FLIGHT\_type +

User\_hotel\_START+user\_hotel\_END+

User contact [email, phone] +

1{

PIN\_VERIFY

}3+

(User address) +

[Pin method, credit card method] +

1{

Ticket\_verification+

[notifier email, notifier sms]

}3**+**

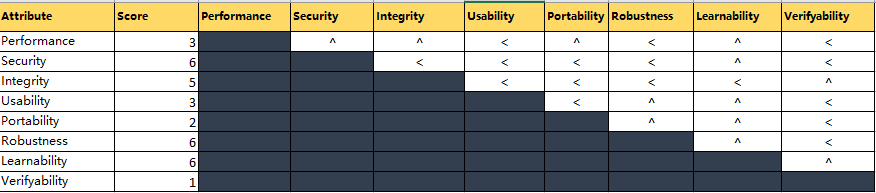
**Ticket\_detail\_slip+**

END

1. **Wieger’s prioritization chart:**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Relative**  **Weights :** | **2** | **1** |  |  | **1** |  | **0.5** |  |  |
| **Feature** | **Relative**  **benefit** | **Relative**  **penalty** | **Total**  **value** | **Value%** | **Relative**  **cost** | **Cost%** | **Relative**  **risk** | **Risk%** | **Priority** |
| Give Breadth  Choices to user to communicate better | **5** | **2** | **12** | **11.5%** | **3** | **12%** | **1** | **6.6%** | **0.269** |
| Wider  Tools to follow the multiple provider and selections | **3** | **5** | **11** | **10.5%** | **6** | **24%** | **2** | **13.3%** | **0.0615** |
| Online payments methods | **9** | **1** | **19** | **18.2%** | **5** | **20%** | **3** | **20%** | **0.113** |
| Generate the payment slips | **8** | **1** | **17** | **16.3%** | **3** | **12%** | **3** | **20%** | **0.169** |
| Budget  Friendly packages generation by agencies | **5** | **4** | **14** | **13.4%** | **2** | **8%** | **2** | **13.3%** | **0.314** |
| Provide Customer care support | **7** | **1** | **15** | **14.4%** | **5** | **20%** | **1** | **6.6%** | **0.135** |
| Give Discount on  packages | **7** | **2** | **16** | **15.3%** | **1** | **4%** | **3** | **20%** | **0.239** |
| **Totals** | **44** | **16** | **104** |  | **25** |  | **15** |  |  |

1. **Quality Attribute Prioritization:**

****