**KAZINAZRUL UNIVERSITY**

A Project submitted in partial fulfillment of

The requirement for the Honors Degree of

Bachelor of Computer Applications

**ONLINE BUS TICKET BOOKING**Paper: -BCAC602

Year of Examination: - 2023-2024

**Submitted by**

**Name**: Punit Patel **Name**: Randhir Kumar Ray

**Reg no**:211211250010 **Reg no**: 211211250028

**Name**: Nitesh Singh **Name**: Bikash Thakur

**Reg no**: 211211250033 **Reg no**: 211211250007

**OF**

**Raniganj Institute of Information Technology**

**ACKNOWLEDGEMENT**

Thanks all to the almighty for blessing me and thanks for all that supported me throughout my final project, **Online Bus Ticket Booking**. I am also thankful for aspiring guidance, invaluably constructive criticism and friendly advice during this project. First and foremost, I express my appreciation to **Mr. Supriyo Roy** as my supervisor for monitoring, guidance and constant give advice throughout the course of this final year project. I want to take this opportunity to thank to my parent and also to all lecturers in Bachelor of Computer Application for the attention, guidance and opinion during project completion. My sincere thanks also go to all my friends, who always give me support and working together during developing this project. I would also to thanks to my friends for their valuable comments and suggestion on this project which gave me aspiration to improve this project. With all your help, I able to solve many difficulties in order to complete this project

**Date:**

**Signature:**

**Date:**

**Signature:**

**Date:**

**Signature:**

**Date:**

**Signature:**

**CERTIFICATE FROM PROJECT GUIDE**

This is to certify that this project entitled “**Online Bus Ticket Booking**” submitted in partial fulfillment of the degree of BCA, the Kazi Nazrul University, done by **Mr. Punit Patel** Reg No.**211211250010** is an authentic work carried out under the guidance of **Mr. Supriyo Roy** at **Raniganj Institute of Information Technology**. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief and project report is developed according to the “**BCA PROJECT & PROJECT REPORT STANDARD 2024, Kazi Nazrul University**”.

**Signature of the student        Signature of the Guide**

**CERTIFICATE FROM PROJECT GUIDE**

This is to certify that this project entitled “**Online Bus Ticket Booking**” submitted in partial fulfillment of the degree of BCA, the Kazi Nazrul University, done by **Mr. Nitesh Kumar** Reg No.**211211250033**is an authentic work carried out under the guidance of **Mr. Supriyo Roy** at **Raniganj Institute of Information Technology**. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief and project report is developed according to the “**BCA PROJECT & PROJECT REPORT STANDARD 2024, Kazi Nazrul University**”.

**Signature of the student                     Signature of the Guide**

**CERTIFICATE FROM PROJECT GUIDE**

This is to certify that this project entitled “**Online Bus Ticket Booking**” submitted in partial fulfillment of the degree of BCA, the Kazi Nazrul University, done by **Mr. Bikash Thakur** Reg No. **211211250007**is an authentic work carried out under the guidance of **Mr. Supriyo Roy** at **Raniganj Institute of Information Technology**. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief and project report is developed according to the “**BCA PROJECT & PROJECT REPORT STANDARD2024, Kazi Nazrul University**”.

**Signature of the student                     Signature of the Guide**

**CERTIFICATE FROM PROJECT GUIDE**

This is to certify that this project entitled “**Online Bus Ticket Booking**” submitted in partial fulfillment of the degree of BCA, the Kazi Nazrul University, done by **Mr. Randhir KumarRay** Reg No. **211211250028**is an authentic work carried out under the guidance of **Mr. Supriyo Roy** at **Raniganj Institute of Information Technology**. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief and project report is developed according to the “**BCA PROJECT & PROJECT REPORT STANDARD 2024, Kazi Nazrul University**”.

**Signature of the student                     Signature of the Guide**

**CERTIFICATE**

The foregoing project report entitled “**Online Bus Ticket Booking**” is hereby approved as a creditable project and has been presented in a satisfactory manner to warrant its acceptance as prerequisite to the degree for which it was submitted.

It is understood that by this approval, the undersigned do not necessarily endorse any conclusion drawn or opinion expressed therein, but approve the project for the purpose for which it is submitted.

**Head of the Institute**

**CERTIFICATE FROM COMPANY**

****

**CERTIFICATE FROM COMPANY**

****

**CERTIFICATE FROM COMPANY**

****

**CERTIFICATE FROM COMPANY**

****

**SELF CERTIFICATE**

This is to certify that the dissertation/project report entitled“**Online Bus Ticket Booking** ” is done by me is an authentic work carried out for the partial fulfillment of the requirements for the award of the degree of BCA under the guidance of **Mr. Supriyo Roy**.

I also certify that I am aware of the “**BCA PROJECT& PROJECT REPORT STANDARD 2024, THE KAZI NAZRUL UNIVERSITY**” issued by The Kazi Nazrul University and this project report is based on that standard. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief.

**Signature of the student**

**Name of the Student**: Punit Patel

**Reg No:** 211211250010

**College Name:** Raniganj Institute of Information Technology

**SELF CERTIFICATE**

This is to certify that the dissertation/project report entitled“**Online Bus Ticket Booking**” is done by me is an authentic work carried out for the partial fulfillment of the requirements for the award of the degree of BCA under the guidance of **Mr.Supriyo Roy**.

I also certify that that I am aware of the “**BCA PROJECT& PROJECT REPORT STANDARD 2024, THE KAZI NAZRUL UNIVERSITY**” issued by The Kazi Nazrul University and this project report is based on that standard. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief.

**Signature of the student**

**Name of the Student**: Randhir Kumar Ray

**Reg No:** 211211250028

**College Name:** Raniganj Institute of Information Technology

**SELF CERTIFICATE**

This is to certify that the dissertation/project report entitled“**Online Bus Ticket Booking**” is done by me is an authentic work carried out for the partial fulfillment of the requirements for the award of the degree of BCA under the guidance of **Mr.Supriyo Roy**.

I also certify that that I am aware of the “**BCA PROJECT& PROJECT REPORT STANDARD 2024, THE KAZI NAZRUL UNIVERSITY**” issued by The Kazi Nazrul University and this project report is based on that standard. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief.

**Signature of the student**

**Name of the Student**: Nitesh Singh

**Reg No:** 211211250033

**College Name:** Raniganj Institute of Information Technology

**SELF CERTIFICATE**

This is to certify that the dissertation/project report entitled“**Online Bus Booking System**” is done by me is an authentic work carried out for the partial fulfillment of the requirements for the award of the degree of BCA under the guidance of **Mr. Supriyo Roy** .

I also certify that that I am aware of the “**BCA PROJECT& PROJECT REPORT STANDARD 2024, THE KAZI NAZRUL UNIVERSITY**” issued by The Kazi Nazrul University and this project report is based on that standard. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief.

**Signature of the student**

**Name of the Student:** Bikash Thakur

**Reg No:** 211211250007

**College Name:** Raniganj Institute of Information Technology

**SYNOPSIS OF THE PROJECT**

**Online Bus Ticket Booking Project in Django** created based on python, Django, and SQLITE3 Database. The proposed Online Bus Ticket Bookingproject system ensures that the users can book the Ticket as per their requirements by logging on to the website. It allows users to book their Online Bus Ticket, manage their bookings and search for the Travel packages at any point of time. The users will be able to select their desired Bus and get the price details.

**Admin features of Online Bus Ticket Booking**

**Dashboard** – A centralized interface where administrators can view key statistics, such as total bookings, upcoming trips, Upcoming Busses etc. This provides an overview of the system's performance.

1. **Booking Management**: Enables administrators to view and manage all bookings made through the system, including searching for bookings, modifying booking details, canceling bookings, and generating booking reports.
2. **Manage User**– The admin can manage the user’s account.Andupdate user’s account.
3. **Login and Logout** – By default one of the security features of this system is the secure login and logout system.

**Customer Features of Online Bus Ticket Booking**

1. **Login** **Page**– public enter their website credentials on this page by which you can see all the discussions and you have now access to login page.
2. **Register Page**– The page where new public created their login credentials for the website.
3. **Home Page**– When public visit the website, this is the system’s default page. This page shows the notices for website.
4. **Ticket Booking**–Book Your Bus ticket quickly and easily through our online booking system. Select your Source, Destination, and Date To book your tickets in just a few clicks.
5. **Show Ticket Details**– View Ticket details of own Booked Tickets, including the journey date, time, and seat Number.
6. **Ticket Cancelling**– Cancel own tickets hassle-free through our ticket cancelling feature.
7. **View Profile**– Access own Profile to view personal information and preferences.

**Index**

1. Objectivesand Scope……………………………………………………….2
2. Benefit………………………………………………………………………4
3. Category of Project………………………………………………………….5
4. Theoretical Background……………………………………………………..6
5. Definition of the Problem…………………………………………………...7
6. System Requirements Specifications…………………………………….….9
7. Methodology adopted for System Implementation…………………………14
8. Details of Hardware& Software…………………………………………….16
9. Justification for particular software……….……………………………...…17
10. Requirement of hardware and software to run the project at the user end….18
11. Scope System Maintenance & Evolution…………………………………...19
12. Cost and Benefit Analysis…………………………………………………...20
13. Security Issue……………………………………………………………….23
14. Detailed Life Cycle of the Project…………………………………………...24
    1. DFD…………………………………………………………………...24
    2. Input and Output Screen Design according to the DFD……...............26
    3. ERD…………………………………………………………………..28
    4. Tables Designed……………………………………….….………….29
    5. Software Engineering Process involved……………………………...31
    6. Code of the project…………………………………………………....33
    7. Screenshot of the forms…………………………………………….…48
    8. Methodology used for testing………………………………………...55
    9. Test Report………………………………………………………...….56
    10. Performance and Efficiency issues……………………………….58
        1. Conclusion……………………………………………….…58
        2. Future scope of the project…………………………………59
        3. Reference …………………………………………………...60
15. **OBJECTIVE AND SCOPE**

**The objectives of the developing Online Bus Ticket Booking:**

1.**Facilitate Convenient Booking**: The primary objective is to provide a user-friendly platform that allows passengers to book bus tickets conveniently from anywhere with an internet connection. This aims to eliminate the need for physical ticket counters and long queues, enhancing the overall customer experience.

2.**Efficient Management**: Streamlining the process of bus ticket booking for both passengers and bus operators is essential. This includes automating various tasks such as seat allocation, schedule management, and payment processing to improve efficiency and reduce operational overhead.

3.**Real-time Information**: Provide passengers with real-time information about bus schedules, seat availability, routes, fares, and any updates or changes to their bookings. This enhances transparency and allows passengers to make informed decisions.

**Scope of the Project**

**Booking**: User Interface: Develop an intuitive and user-friendly interface for both web and mobile platforms, allowing passengers to search for buses, view schedules, select seats, make payments, and receive booking confirmations seamlessly**.**

**Booking Management:** Implement features for passengers to search for buses based on criteria such as origin, destination, date, and time, as well as book tickets for single or multiple passengers, choose preferred seats, and select additional services if available (e.g., onboard amenities).

**Login Process:** In this the customer has to give out the login details i.e. user id and password and then only he can be logged on. The user id and password given by the customer are checked from the data stored in the database.

**Registration Process**: User must be registered before booking a cab. Proper

Validations will be provided to keep only authenticated users i.e., those users who will provide correct information. All the data supplied by the user will be stored in database and it will be used for further validation and authentication. During registration, user has to give login and password of their choice. Login names and passwords will be stored in the database so that the user can directly login without registering again and again.

1. **BENEFITS**

* **Convenience**: Users can book Bus tickets from anywhere with an internet connection, eliminating the need to visit physical ticket counters or wait in long queues. This convenience saves time and effort for users**.**
* **24/7 Availability**: Online booking systems are available 24 hours a day, 7 days a week, allowing users to make reservations at any time, even outside of regular business hours.
* **Easy Access to Information**: Users can easily access information about Bus schedules, available seats, fares, and other details through the online platform, enabling them to make informed decisions.
* **Exact location**: of the passenger eliminates unnecessary time wasting
* **Flexibility**: Online booking systems often offer flexibility in terms of selecting seats, choosing departure times, and modifying or canceling bookings if needed, providing users with greater control over their plans.
* **Special Offer and Discounts:** Users may have access to special offers, discounts, or loyalty programs available exclusively through the online platform, providing added value for online bookings.
* **OnlineBus TicketBooing :**System will help your business grow by drawing in more passengers

1. **CATEGORY OF PROJECT**

The project is a Web Based Application Project built on Django Framework.

**Web Based Application Projects:** A web application is the process of developing a website, activities in a network which are aimed at a pre-defined goal. We offer web application projects to develop and test an advanced website that can be used for various commercial purposes and analyze the activity, impact, popularity and Google ranking of the website.

Web applications are more popular because of the following reasons:

* Compared to desktop applications, web applications are easier to maintain by as they use the same code in the entire application. There are no compatibility issues.
* Web applications can be used on any platform: Windows, Linux, Mac… as they all support modern browsers.
* Mobile App Shop approval not required in web applications.
* Released any time and in any form. No need to remind users to update their applications.
* You can access these web applications 24 hours of the day and 365 days a year from any PC.
* You can either make use of the computer device.
* Web applications are a cost-effective option for any organization. Seat Licenses for Desktop software are expensive.
* Web-Based Apps are Internet-enabled apps that are accessed through the mobile’s web browser. Therefore, you don’t require downloading or installing them.

1. **THEORETICAL BACKGROUND**
   1. **System Architecture:** Understanding the architectural design of the online bus ticket booking system is crucial. It involves components like client-side interfaces (web), server-side processing, and database management.
   2. **Database Management:** A robust database schema is essential for storing and managing information such as user profiles, bus routes, schedules, seat availability, and bookings. Concepts like normalization, indexing, and querying play a significant role here.
   3. **User Interface Design:** Principles of user experience (UX) and user interface (UI) design are vital for creating intuitive and user.

**1) A Home page to search for travelling mode:** This is the page where the user will provide the details required and search for the Bus.

**2) User portal:** The user can register and login using username and password to book the desired bus ticket.

**4) Admin:** Admin can manage the whole portal. Admin can view the bookings, registered users, and update the profile.

1. **DEFINITION OF A PROBLEM**

**Non-responsive Website**: A non-responsive website is one that fails to dynamically adjust its layout, content, and functionality based on the screen size, resolution, and orientation of the device accessing it. As a result, users may encounter difficulties in viewing and interacting with the website on smaller screens such as smartphones or tablets, leading to a poor user experience and potential loss of engagement and conversions.

**Absence of Online Payment System**: This refers to a situation where a website does not incorporate a secure and convenient method for users to complete financial transactions or payments online. Without an online payment system, users may be unable to purchase goods or services directly through the website, leading to limitations in monetization, customer satisfaction, and overall business growth.

**Domain Restrictions**: The website may be hosted on a domain with geographic restrictions, such as a country-specific top-level domain (e.g., India, .de) that limits its visibility to users within that country.

**Content Localization**: The website's content may be tailored to a specific language or culture, making it less appealing or relevant to users from other regions who speak different languages or have different cultural preferences.

**Lack of Customization:**Without the option to choose specific seats, users may miss out on personalizing their travel experience according to their preferences, such as seating comfort or proximity to amenities.

**Delayed Query Resolution:** The absence of real-time chat support prolongs the response time for user queries, potentially leading to frustration and dissatisfaction, especially in urgent situations or during peak booking times.

**Cultural and Regional Exclusion:** Limited language support may not only impact inclusivity but also hinder the project's ability to reach diverse demographics, particularly those from non-English speaking regions or with specific cultural preferences, limiting its market penetration and growth potential.

1. **SOFTWARE REQUIREMENTS SPECIFICATION**

The Software Requirements Specification is produced at the culmination of the analysis task.The function and performance allocated to software as part of system engineering are refinedbyestablishing a completeinformation description, adetailed functional andbehavioraldescription, an indication of performance requirements and design constraints, application or privatevalidation criteria, and other data pertinent to requirements.

**Proposed System**

**1. Details:** The new proposed system stores and maintains all the online users etc.

**2. Calculations:** The new proposed system updates tables and other information automatically and it is very fast and accurate.

**3. Registers:** There is no need of keeping and maintaining accounts and information manually. It remembers each and every record and we can get any report at any time.

**4. Speed:** The new proposed system is very fast with 100% accuracy and saves time.

**5. Manpower:** The new proposed system needs less manpower. Less people can do the large work.

**6. Efficiency:** The new proposed systems complete the work of many people in less time.

**7. Past details:** The new proposed system contains the details of each enquiry done by visitor.

**8. Reduces redundancy:** The most important benefit of this system is that it reduces the redundancy of data within the data.

**9. Work load:** Reduces the work load of the data store by helping in easy updates of the products and providing them with the necessary details together with financial transactions management.

The proposed system has the following requirements:

**Functional Requirements:**

* The System must provide following functionalities—
* Keeping records of registration of user.
* Keeping the records of ticket.
* Keeping the daily booking and scheduling.

**Non-Functional Requirements:**

* Security – Only authorized corporate workers may get access to the firm's secured page on the systems, and only users with proper passwords and usernames can log in to see the user page.
* Performance and Response Time – The system should have a high-performance rate while executing user input and should be able to offer feedback or a response in a short amount of time, often 50 seconds for extremely difficult activities and 20 to 25 seconds for less sophisticated jobs.
* Error Handling – Errors should be avoided as much as possible, and a suitable error message should be supplied to help the user through the recovery process. The importance of validating user input cannot be overstated. In addition, the time it takes to

recover from a mistake should be between 15 and 20 seconds.

* Availability –This system must be accessible at all times, 24 hours a day, seven days a week. In the event of a catastrophic system failure, the system should be back up and running within 1 to 2 business days, ensuring that the business process is not disrupted.
* Authentication and Authorization: Only authenticated users should have access to booking functionalities, and proper authorization mechanisms should be in place to restrict access to sensitive operations.
* User Interface Consistency: The user interface should be consistent across all pages and devices to provide a seamless user experience.
* Accessibility: The system should comply with accessibility standards (e.g., WCAG) to ensure that it is usable by individuals with disabilities.

**Identification of Need**

The old manual system was suffering from a series of drawbacks. Since whole of the systemwas to be maintained with hands the process of keeping, maintaining and retrieving theinformation was very tedious and lengthy. The records were never used to be in a systematical order, there used to be lots of difficulties in associating any particular transaction with aparticular context. If any information was to be found it was required to go through thedifferent registers, documents there would never exist anything like report generation.Therewould always be unnecessary consumption of time while entering records and retrievingrecords. One more problem was that it was very difficult to find errors while entering therecords. Once the records were entered it was very difficult to update these records. The reason behind it is that there is lot of information to be maintained and have to be kept in mind while running the business. For this reason, we have provided features Present system is partially automated (computerized), actually existing system is quite laborious as one has to enter same information at three different places.Following points should be well considered. Documents and reports that must be provided by the new system: there can also be few reports, which can help management in decision-making and cost controlling, but since these reports do not get required attention, such kind of reports and information were also identified and given required attention.

* Details of the information needed for each document and report.
* The required frequency and distribution for each document.
* Probable sources of information for each document and report.

With the implementation of computerized system, the task of keeping records in an organizemanner will be solved. The greatest of all is the retrieval of information, which

will be at the click of the mouse. So, the proposed system helps in saving the time in different operations and making information flow easy giving valuable reports

**Feasibility study**

It includes consideration of all the possible ways to provide a solution to the given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily done based on the future upcoming requirements.

**A. Economic Feasibility:** This is a very important aspect to be considered while developing a project. We decided that technology has to be based on minimum possible cost factor. All hardware and software cost has to be borne by the organization. Overall we have estimated that the benefits the organization is going to receive from the proposed system will surely overcome the initial costs and the later on running cost for system.

**B. Technical Feasibility:** This included the study of function constraints that may affect the ability to achieve an acceptable system. For this feasibility study, we studied complete functionality to be provided in the system, as described in the System Requirement Specification (SRS), and checked if everything was possible using different type of front end and backend platform.

**C. Operational Feasibility:** No doubt the proposed system is fully GUI based that is very user friendly and all inputs to be taken all self-explanatory even to a layman. Besides, a proper training has been conducted to let know the essence of the system to the users so that they feel comfortable with new system. As far our study is concerned the clients are comfortable and application as the system has cut down their loads and doing.

**System Analysis**

System analysis is a thorough examination of a system's different processes and their interrelationships both within and outside the system. The key question here is – why are there so many flaws in the current system? What measures should be taken to address the problem? When a user or management begins a study of the software utilizing the current system, analysis begins. Data was collected on numerous files, decision points, and transactions handled by the current system during the analysis. For example, Data Flow Diagrams, etc. are widely utilized in the system. For the collection of important information needed to create the system, training, experience, and common sense are necessary. The system's success is primarily determined by how well the problem is identified, fully studied, and appropriately implemented via the selection of a solution. A good analytical model should include not just methods for comprehending the problem, but also the framework for solving it. As a result, it should be extensively investigated by gathering data about the system. The suggested system should next be extensively examined in light of the requirements.

System analysis is divided into four sections.

1) Initial research and system architecture.

2)Using analytic tools to do structured analysis.

3)Feasibility study.

4)Analyze the cost and benefits.

1. **METHODOLOGY ADOPTED FOR SYSTEM IMPLEMENTATION**

This project is using Structured Analysis and Design Methodology (SADM) by application lies waterfall model. This methodology is suitable to time given in this project. It starts with planning, requirement analysis, design, implementation, testing, operation and maintenance. Every phase must fulfill the requirement specification. After the system complete, it will undergo maintenance as final test.

* **Planning Phase**: The goals in this project are to decide or choose the system and title for this project with supervisor. The system has been chosen to proceed wasOnline Bus TicketBooking. This system will develop for user’s facilities to schedule the Bus Ticket bookinginstantly according to desired destination. An abstract is done based on information gathered and the system that has been agreed to develop.
* **Requirement analysis phase**:The objectives are to collect all requirement for user and system. In this phase, information requirement gathered by review on online application current system and the technique they used. The requirement is discovered and upgrade from older system. To better understanding about the existing application scheduling system, an observation of real system and comparison from the existing system is done. In order to build the project that more efficient, the system must be user friendly and easy to use.
* **Design Phase**:In these phases contain several diagrams to show and guide the process and flow of the functioning system. This design phase will have divided into logical and physical model. The several diagrams are Data Flow Diagram (DFD) level 1 and 0 then Relationship Diagram (ERD). Those diagrams built to
* Know more details about system. Then physical model is designing the database and interfaces.
* **Implementation Phase**:Implementation phase is about transformation design into the real system by implementing coding. The system was implemented into coding based on sub module followed by system requirement. This project has been developed using Django Framework and Python used as local host.
* **Testing phase:** After writing the code, the system module is tested by using single unit testing to test the single module of the system. It to know whether the system fulfill the requirement or not and the completeness of the system. Then, the system conducted to test error of the whole system. Any error or bugs will be fixed, and the system will repeat testing phase until there’s no error founds.
* **Evaluation phase**:At this phase the system will evaluated before deploying to the end user. Some practice and techniques that were used in first iteration then can be used for the next requirement.
* **Deployment phase**:After the system free from bugs, it can have released and can used to the user. Once the system steady state, it reviewed that the system has meet the objectives and goals. The maintenance process also important part because if there was an error it can corrected so it will make sure all process of the system functioned well.

1. **DETAILS OF HARDWARE AND SOFTWARE**

**SOFTWARE REQUIREMENTS:**

* Windows or above
* Django Framework version 3 or above
* Python 3.7 or above
* Sqlite3 db.
* Draw.io
* MS Office 2019
* HTML
* CSS
* JS
* Bootstrap

**HARDWARE REQUIREMENTS:**

* Processor: Intel i3 or above
* Hard Disk: 1TB
* Keyboard
* Mouse
* LCD Monitor

### JUSTIFICATION FOR THE SOFTWARE

The new system was designed to solve problem affecting the manual system in use. It was designed use online thereby relieving both student and staff from much stress as experienced from the manual system.

This was the analyzing and storing of information either automatically or interactively, it made use of online access to internet. The proposed system also had some other features like.

* Accuracy in handling of data
* Fast rate of operation and excellent response time.
* Flexibility, that is to say, it can be accessed at any time
* Easy way of back up or duplicating data in diskette in case of data loss.
* Better storage and faster retrieval system.
* Accessibility from any part of the world.

1. **HARDWARE AND SOFTWARE TO RUN THE PROJECT AT THE USER END**

Minimum hardware and software requirement to execute Django project on your Windows operating system as follows:

**Hardware Requirements**

* Intel Core i3 processor or above
* Hard Drive of 500GB and above
* Minimum of 4 MB memory RAM
* Mouse, keyboard

**Software Requirements**

* Windows 8 OS or above
* Django 3.0 or above
* Python 3.6 or above
* Text Editors (Visual Studio Code )

1. **SCOPE SYSTEM MAINTENANCE & EVOLUTION**

Software development is never ending process and continues the life of the software as per the changing needs of the user from time to time. The project is no doubt has been developed keeping in mind easy modification and enhancement that may be required from time to time.

However, there are many scopes to modify this software. As because due to shortage of time, we here become unable to include many things. We are trying to cover all their existing system for sales return records of the items but due to shortage of time we become unable to include many things. Due to lake of time, I here include none of them and a future scope one can develop these returns which are so much essential. Only with a little more doing it is possible to design the formats for those returns. Moreover, an on-line system will be more helpful to the organization. With almost the same data with only a little modification an on-line system can be designed to fulfill their demands. All these can be considered to be future scope for this project.

1. **COST AND BENEFIT ANALYSIS**

In general, COCOMO estimates project cost, derived directly from person-months effort, by assuming the cost is basically dependent on total physical size of all project files, expressed in thousands single lines of code (KSLOC). The estimation formulas have the form:

<A name=formula></A>Effort (in person-months) = a x KSLOC<SUP>b</SUP>

The Effort Adjustment Factor is the product of the 15 adjustment parameters. Each adjustment parameter is categorized as very low, low, nominal, high, or very high. All the adjustment parameters are listed below:

RELY Required reliability 0.75 – 1.40

• DATA Database size 0.94 – 1.16

• CPLX Product complexity 0.70 – 1.65

• TIME Execution time constraint 1.00 – 1.66

• STOR Main storage constraint 1.00 – 1.56

• VIRT Virtual machine volatility 0.87 – 1.30

• TURN Computer turnaround time 0.87 – 1.15

• ACAP Analyst capability 1.46 – 0.71

• AEXP Application Experience 1.29 – 0.82

• PCAP Programmer capability 1.42 – 0.70

• VEXP Virtual machine experience 1.21 – 0.90

• LEXP Language experience 1.14 – 0.95

• MODP Use of modern practices 1.24 – 0.82

• TOOL Use of software tools 1.24 – 0.83

• SCED Required development schedule 1.23 – 1.10

Adjustment factors for the Online Cab Booking System are listed below:

• RELY 1.00 Nominal

• DATA 1.00 Nominal

• CPLX 0.85 Low

• TIME 1.00Nominal

• STOR 1.00 Nominal

• VIRT 0.87 Low

• TURN 0.87 Low

• ACAP 1.00 Nominal

• AEXP 1.13 Low

• PCAP 1.00 Nominal

• VEXP 1.00 Nominal

• LEXP 1.00 Nominal

• MODP 0.91 High

• TOOL 0.91 High

• SCED 1.00 Nominal

The EAF value evaluated to 0.60. I have estimated the size to be around 3.00.

From the calculation

I got EFFORT = 6.08

TIME = 4.96

* 1. **. SECURITY ISSUES**

In this project, user can view, search, book, and cancel seat very easily online. This is controlled by an admin who have full control over all the Booking and Listing of Buses. All the website data are stored into the Database.So there is a security issue of theft of Related to user login data, Booking etc.

Online Bus Ticket booking provides a platform where all the Bus list are kept in an arranged way and User can choose own required bus by just selecting it from the system. After selecting bus and view all the description such as date, time, destination user can Book Seat of it in order to get the Ticket. If the database is compromised or login of admin section is compromised then there is a chance that the data can be misused.

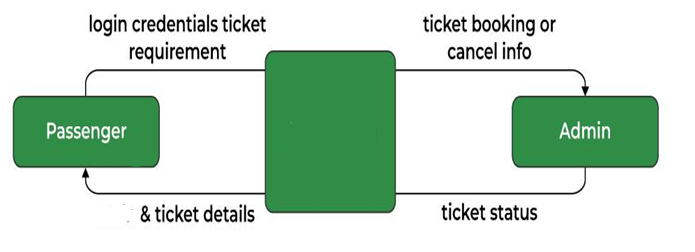
**14.DETAILED LIFE CYCLE OF THE PROJECT**

**14.1. Data flow diagram**

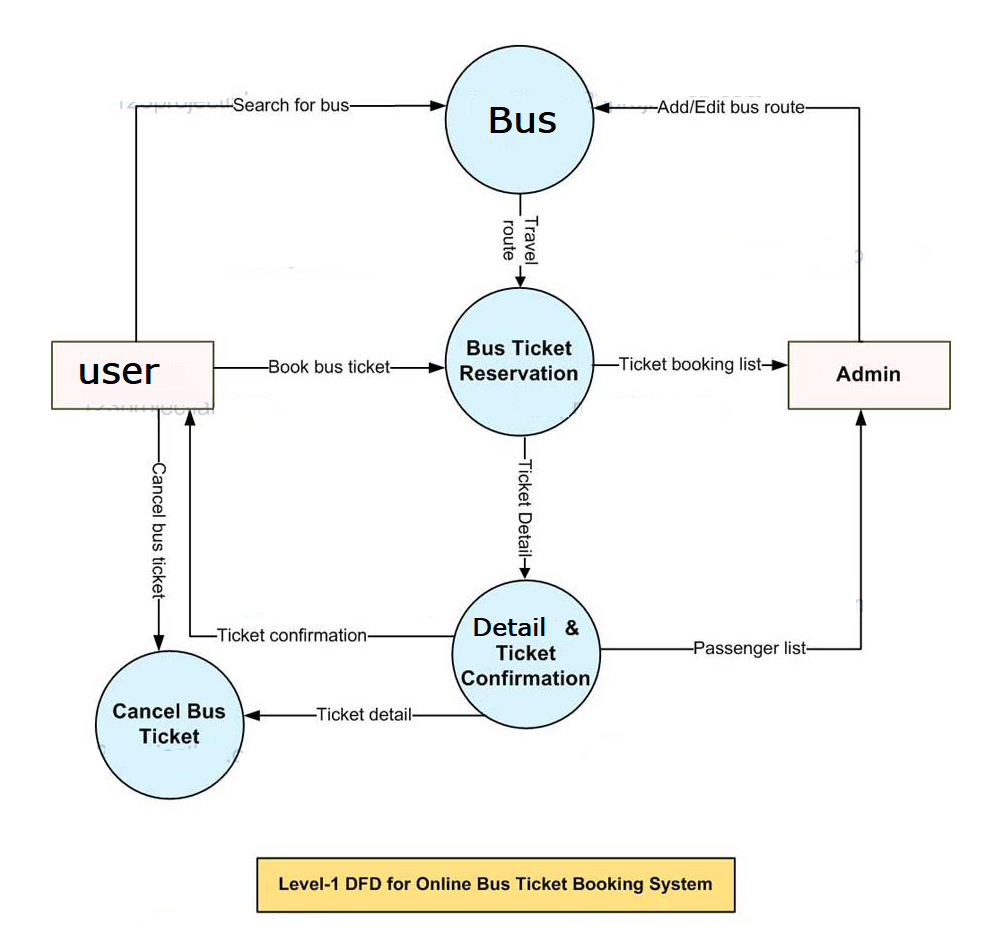
**0levelDFD.**

**Online Bus Ticket**

**Booking**

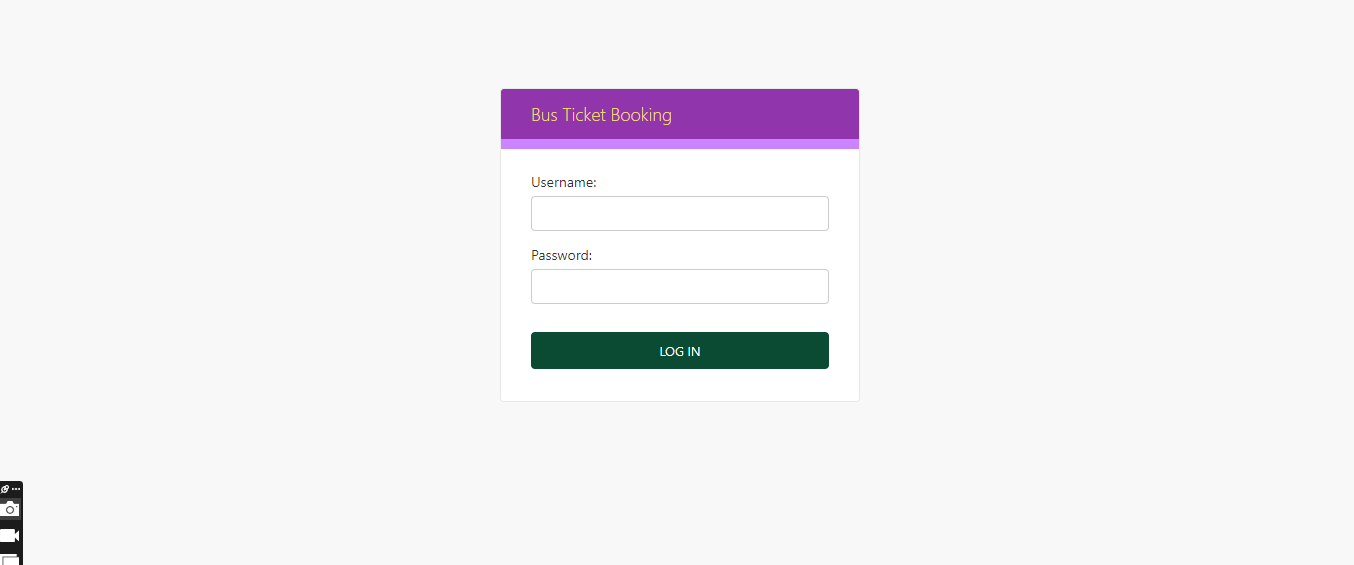


**1levelDFD.**

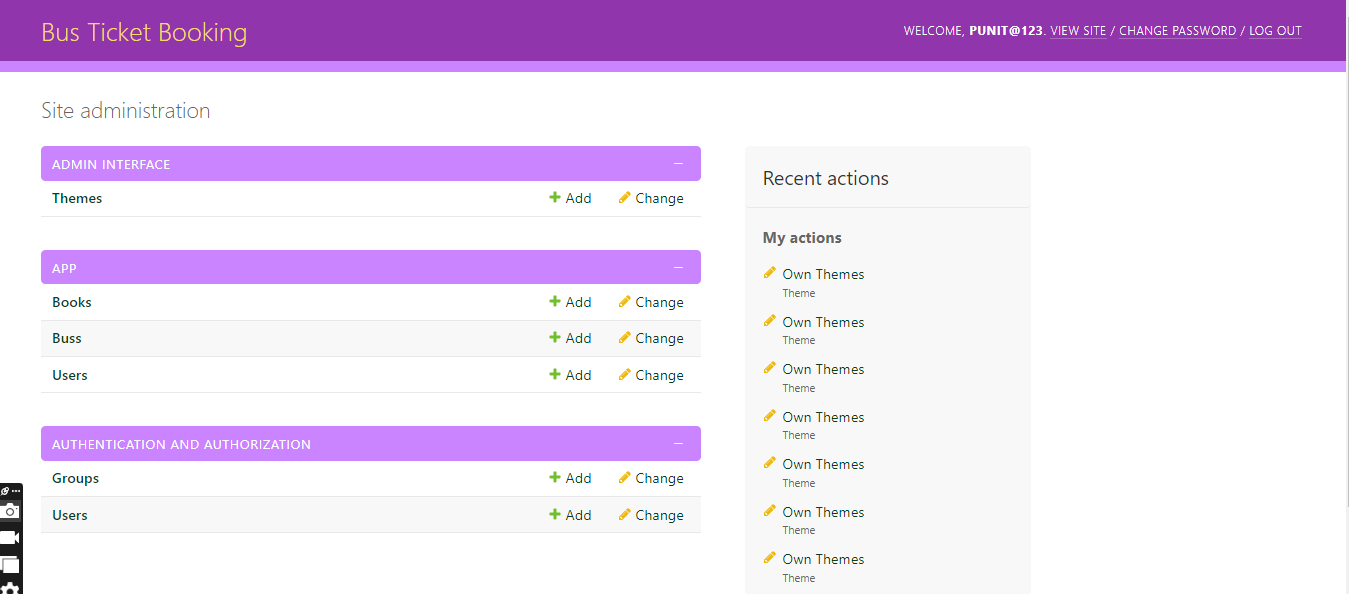


**14.2. Input and Output Screen Design according to the DFD**

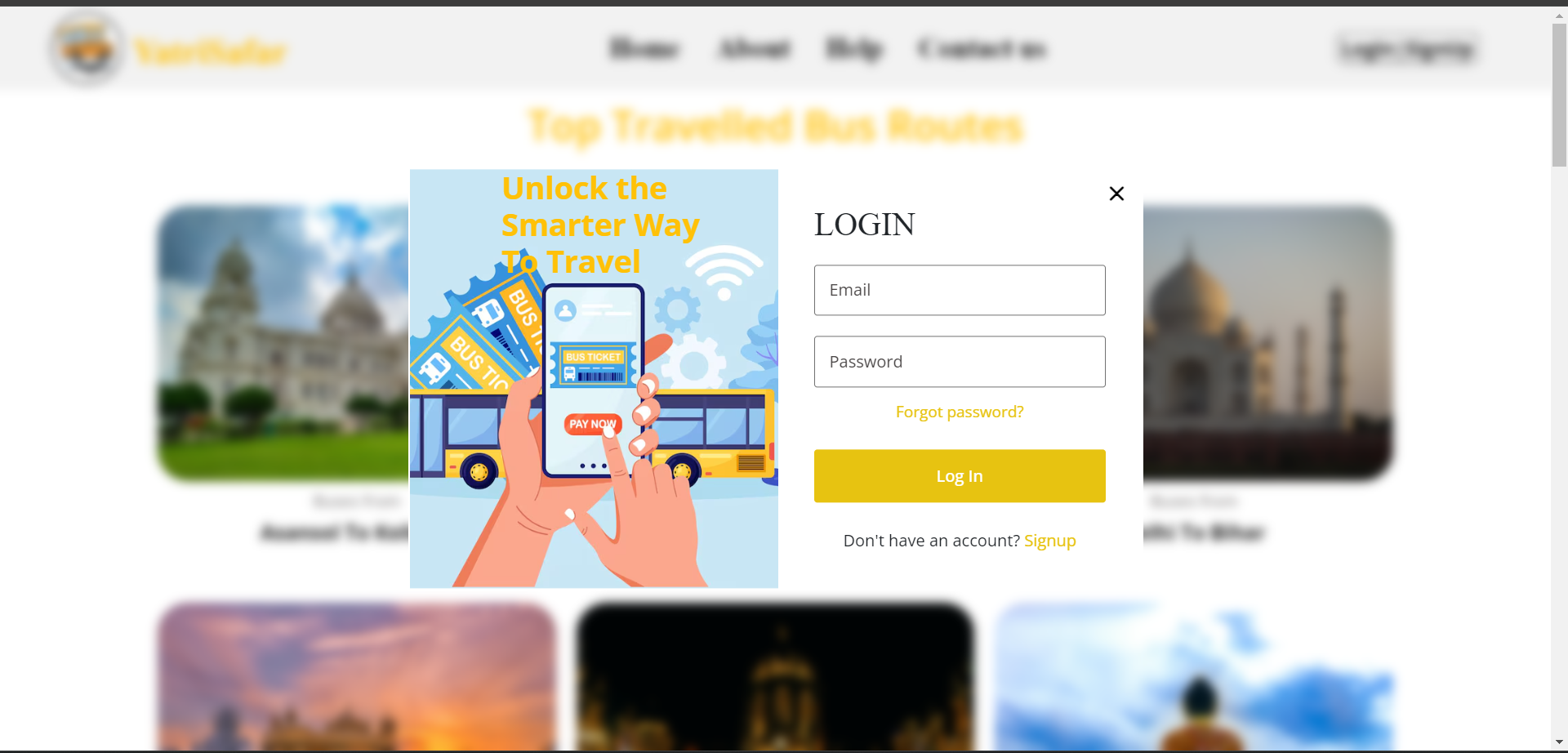
**Admin Login**

****

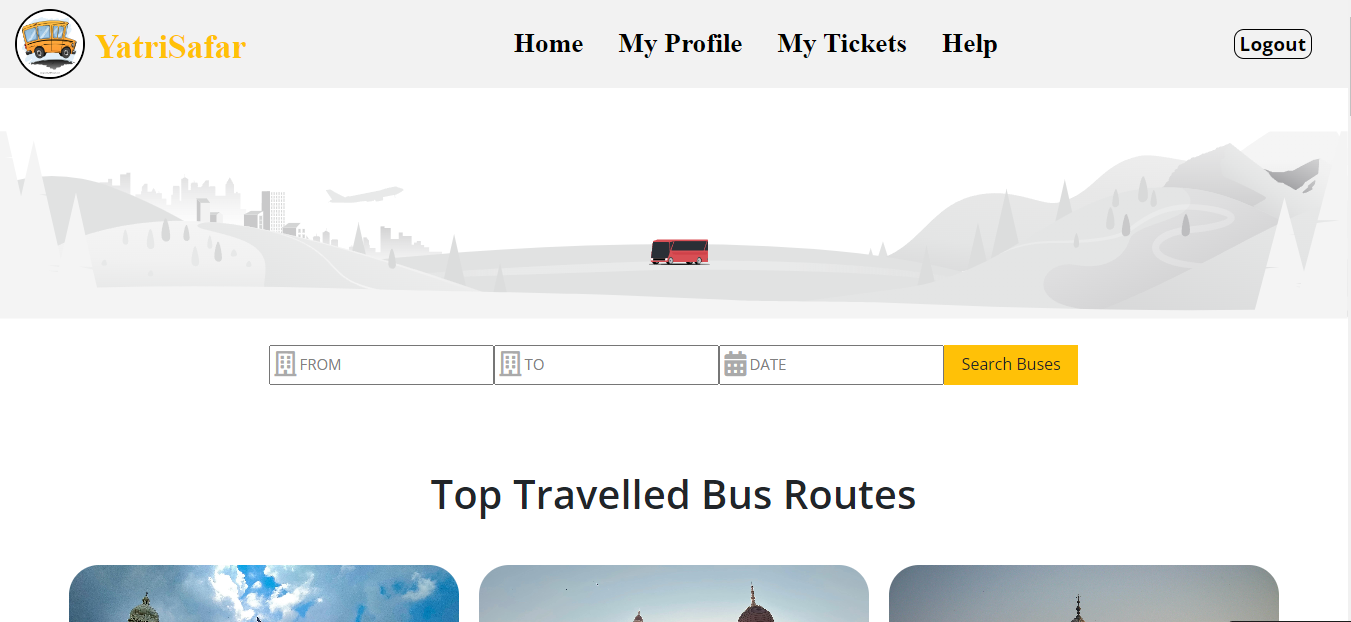
**Admin after Login**

****

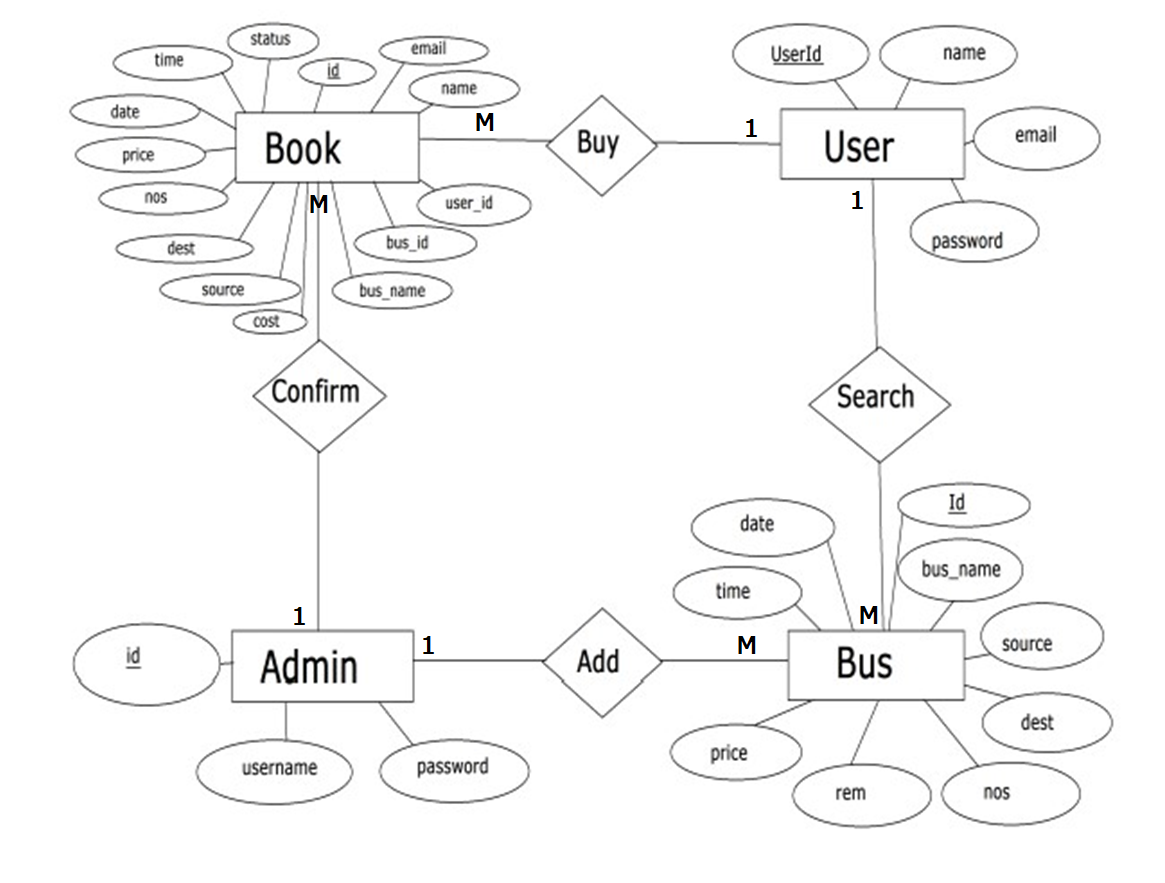
**User Login**

****

**Users after Login**

****

**14.3.Entity Relationship Diagram**

****

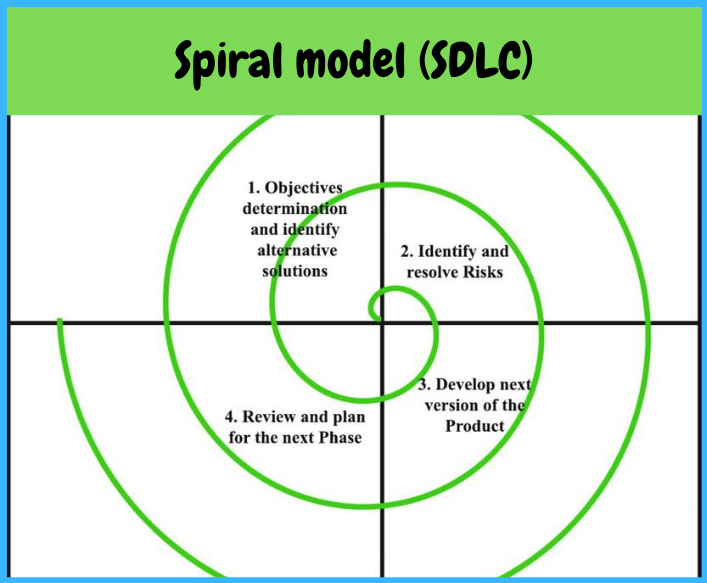
**14.4.Tables Designed**

|  |  |
| --- | --- |
| **Table** | **Fields** |
| User | Username |
|  | Password |
|  | First Name |
|  | Last Name |
|  | Email Id |

|  |  |
| --- | --- |
| **Table** | **Fields** |
| Busses | Bus Name |
|  | Source |
|  | Destination |
|  | Total Seat |
|  | Available Seat |
|  | Ticket Price |
|  | Date |
|  | Time |

|  |  |
| --- | --- |
| **Table** | **Fields** |
| Booked | Passenger Name |
|  | Bus Name |
|  | Source |
|  | Destination |
|  | Price |
|  | Date |
|  | Time |
|  | Status |

**14.5. Software Engineering Process involved**



We choose the Spiral MODEL due to the following reasons:

Iterative Development: Allows for iterative development cycles, enabling continuous refinement and improvement of the system based on feedback and changing requirements.

Risk Management: Incorporates risk analysis and mitigation at each iteration, facilitating early identification and resolution of potential issues before they escalate, thereby reducing project risks.

Flexibility: Offers flexibility to accommodate changes and enhancements throughout the development process, ensuring alignment with evolving stakeholder needs and market demands.

Enhanced Stakeholder Involvement: Encourages active involvement of stakeholders in the development process through regular reviews and feedback sessions, fostering collaboration and ensuring the final product meets user expectations.

Quality Assurance: Facilitates rigorous testing and validation at each iteration, leading to higher quality deliverables and reducing the likelihood of defects or errors in the final product.

Progressive Elaboration: Allows for progressive elaboration of requirements and design elements over multiple iterations, enabling a more thorough understanding of project complexities and ensuring comprehensive solutions.

First of all, the feasibility study is done. Once that part is over the requirement analysis and project planning begins. After the requirements study is done, the design process begins, followed by the coding process. Once the programming is completed the testing is done.

In this model, the sequences of activities performed in a software development project are: -

1. Requirement Analysis

2. Project Planning

3. System design

4. Detail design

5. Coding

6. Unit testing

7. System integration & testing

**14.6. Code of the Project**

**Models.py**

# Create your models here.

from django.db import models

# Create your models here.

class Bus(models.Model):

bus\_name = models.CharField(max\_length=30)

source = models.CharField(max\_length=30)

dest = models.CharField(max\_length=30)

nos = models.DecimalField(decimal\_places=0, max\_digits=2)

rem = models.DecimalField(decimal\_places=0, max\_digits=2)

price = models.DecimalField(decimal\_places=2, max\_digits=6)

date = models.DateField()

time = models.TimeField()

def \_\_str\_\_(self):

return self.bus\_name

class User(models.Model):

user\_id = models.AutoField(primary\_key=True)

email = models.EmailField()

name = models.CharField(max\_length=30)

password = models.CharField(max\_length=30)

def \_\_str\_\_(self):

return self.email

class Book(models.Model):

BOOKED = 'B'

CANCELLED = 'C'

TICKET\_STATUSES = ((BOOKED, 'Booked'),

(CANCELLED, 'Cancelled'),)

email = models.EmailField()

name = models.CharField(max\_length=30)

userid =models.DecimalField(decimal\_places=0, max\_digits=2)

busid=models.DecimalField(decimal\_places=0, max\_digits=2)

bus\_name = models.CharField(max\_length=30)

source = models.CharField(max\_length=30)

dest = models.CharField(max\_length=30)

nos = models.DecimalField(decimal\_places=0, max\_digits=2)

price = models.DecimalField(decimal\_places=2, max\_digits=6)

cost = models.DecimalField(decimal\_places=2, max\_digits=6, default=0)

date = models.DateField()

time = models.TimeField()

status = models.CharField(choices=TICKET\_STATUSES, default=BOOKED, max\_length=2)

def \_\_str\_\_(self):

return self.name

**views.py**

from django.shortcuts import render,redirect

from django.http import HttpResponse, HttpResponseRedirect

from .models import \*

from decimal import Decimal

from django.contrib.auth import authenticate, login, logout

from django.contrib.auth.models import User

from .forms import UserLoginForm, UserRegisterForm

from django.contrib.auth.decorators import login\_required

from django.contrib import messages

# Create your views here.

#def home(request):

# return render(request, 'app/index.html')

def about(request):

return render(request, 'app/about.html')

def help(request):

if request.user.is\_authenticated:

return render(request, 'app/homehelp.html')

else:

return render(request, 'app/indexhelp.html')

def contact(request):

if request.user.is\_authenticated:

return render(request, 'app/homecontact.html')

else:

return render(request, 'app/indexcontact.html')

def home(request):

if request.user.is\_authenticated:

return render(request, 'app/home.html')

else:

return render(request, 'app/index.html')

def myprofile(request):

return render(request, 'app/myprofile.html')

def aboutus(request):

return render(request, 'app/aboutus.html')

@login\_required(login\_url='signin')

def findbus(request):

context = {}

if request.method == 'POST':

source\_r = request.POST.get('source')

dest\_r = request.POST.get('destination')

date\_r = request.POST.get('date')

bus\_list = Bus.objects.filter(source=source\_r, dest=dest\_r, date=date\_r)

if bus\_list:

return render(request, 'app/list.html', locals())

else:

messages.error(request,"Sorry no buses availiable")

return render(request, 'app/home.html', context)

else:

return render(request, 'app/home.html')

@login\_required(login\_url='signin')

def booking(request,id):

d=Bus.objects.get(id=id)

return render(request,'app/booking.html',{'u':d})

@login\_required(login\_url='signin')

def booked(request,id):

context = {}

if request.method == 'POST':

#id\_r = request.POST.get('bus\_id')

passenger = request.POST.get('passenger')

seats\_r = int(request.POST.get('no\_seats'))

bus = Bus.objects.get(id=id)

if bus:

if bus.rem> int(seats\_r):

name\_r = bus.bus\_name

cost\_r = int(seats\_r) \* bus.price

source\_r = bus.source

dest\_r = bus.dest

nos\_r = Decimal(bus.nos)

price\_r = bus.price

date\_r = bus.date

time\_r = bus.time

username\_r = request.user.username

email\_r = request.user.email

userid\_r = request.user.id

rem\_r = bus.rem - seats\_r

Bus.objects.filter(id=id).update(rem=rem\_r)

book = Book.objects.create(name=passenger, email=email\_r, userid=userid\_r,bus\_name=name\_r,

source=source\_r, busid=id,

dest=dest\_r, price=price\_r,cost=cost\_r, nos=seats\_r, date=date\_r, time=time\_r,

status='BOOKED')

print('------------book id-----------', book.id)

#book.save()

return render(request, 'app/booked.html', locals())

else:

messages.error(request,"Sorry Select Fewer Number of Seats")

d=Bus.objects.get(id=id)

return render(request,'app/booking.html',{'u':d})

#return render(request, 'app/booking.html', context)

else:

return render(request, 'app/home.html')

@login\_required(login\_url='signin')

def cancellings(request):

context = {}

if request.method == 'POST':

id\_r = request.POST.get('bus\_id')

#seats\_r = int(request.POST.get('no\_seats'))

try:

book = Book.objects.get(id=id\_r)

bus = Bus.objects.get(id=book.busid)

rem\_r = bus.rem + book.nos

Bus.objects.filter(id=book.busid).update(rem=rem\_r)

#nos\_r = book.nos - seats\_r

Book.objects.filter(id=id\_r).update(status='CANCELLED')

Book.objects.filter(id=id\_r).update(nos=0)

messages.success(request," Your Ticket Has Been Cancel")

return redirect(seebookings)

except Book.DoesNotExist:

messages.error(request,"Sorry You have not booked that bus")

return render(request, 'app/error.html', context)

else:

return render(request, 'app/home.html')

@login\_required(login\_url='signin')

def seebookings(request,new={}):

context = {}

id\_r = request.user.id

book\_list = Book.objects.filter(userid=id\_r)

if book\_list:

return render(request, 'app/booklist.html', locals())

else:

messages.error(request,"Sorry no buses booked")

return render(request, 'app/home.html', context)

def signup(request):

context = {}

if request.method == 'POST':

name\_r = request.POST.get('name')

email\_r = request.POST.get('email')

password\_r = request.POST.get('password')

user = User.objects.create\_user(name\_r, email\_r, password\_r, )

if user:

login(request, user)

return render(request, 'app/thank.html')

else:

context["error"] = "Provide valid credentials"

return render(request, 'app/index.html', context)

else:

return render(request, 'app/index.html', context)

def signin(request):

context = {}

if request.method == 'POST':

name\_r = request.POST.get('name')

password\_r = request.POST.get('password')

user = authenticate(request, username=name\_r, password=password\_r)

if user:

login(request, user)

# username = request.session['username']

context["user"] = name\_r

context["id"] = request.user.id

return render(request, 'app/success.html', context)

# return HttpResponseRedirect('success')

else:

messages.error(request,"Please Enter Correct ID/Password")

return render(request, 'app/index.html', context)

else:

messages.error(request,"You are not logged in")

return render(request, 'app/index.html', context)

def signout(request):

context = {}

logout(request)

messages.error(request,"You Log Out")

return render(request, 'app/index.html', context)

def success(request):

context = {}

context['user'] = request.user

return render(request, 'app/success.html', context)

@login\_required(login\_url='signin')

def cancel(request,id):

d=Book.objects.get(id=id)

return render(request,'app/cancel.html',{'c':d})

@login\_required(login\_url='signin')

def detail(request,id):

d=Book.objects.get(id=id)

return render(request,'app/detail.html',{'l':d})

@login\_required

def info(request):

if request.method == "POST":

name = request.POST.get("name")

pickup = request.POST.get("pickup")

time = request.POST.get("time")

phone = request.POST.get("phone")

email = request.POST.get("email")

price = request.session['price']

context = {'name': name, 'pickup': pickup,

'phone': phone, 'email': email, 'price': price}

return render(request, "invoice.html", context)

else:

try:

price = request.session['price']

except:

return redirect('index')

# retrieves saved user info

username = request.user.username

user = User.objects.get(username=username)

fullname = user.first\_name+' '+user.last\_name

email = user.email

phone = user.profile.phone

context = {'name': fullname, 'email': email, 'phone': phone}

return render(request, "info.html", context)

def signup(request):

if request.method == "POST":

username = request.POST.get("username")

password = request.POST.get("password")

first\_name = request.POST.get("fname")

last\_name = request.POST.get("lname")

email = request.POST.get("email")

phone = request.POST.get("phone")

if User.objects.filter(username=username).exists() or User.objects.filter(email=email).exists() or Profile.objects.filter(phone=phone).exists():

context = {'msg': "User exists, Please Login"}

return render(request, 'signup.html', context)

else:

u = User(username=username, first\_name=first\_name,

last\_name=last\_name, email=email)

u.set\_password(password)

u.save()

p = Profile(user=u, phone=phone)

p.save()

user = authenticate(username=username, password=password)

**urls.py**

**from django.contrib import admin**

**from django.urls import path**

**from .views import \***

**urlpatterns = [**

**#path('admin/', admin.site.urls),**

**path('home/',home),**

**path('about/',about),**

**path('aboutus/',aboutus),**

**path('help/',help),**

**path('contact/',contact),**

**path('',home),**

**path('myprofile/',myprofile),**

**path('findbus/',findbus),**

**path('booking/<int:id>',booking),**

**path('booked/<int:id>',booked),**

**#My Tickets**

**path('seebookings/',seebookings),**

**path('cancellings/',cancellings),**

**path('detail/<int:id>',detail),**

**path('cancel/<int:id>',cancel),**

**path('signup/',signup),**

**path('signin/',signin),**

**path('success/',success),**

**path('signout/',signout),**

**]**

mportos

**"""**

**Django settings for yatrisafar project.**

**Generated by 'django-admin startproject' using Django 5.0.3.**

**For more information on this file, see**

**https://docs.djangoproject.com/en/5.0/topics/settings/**

**For the full list of settings and their values, see**

**https://docs.djangoproject.com/en/5.0/ref/settings/**

**"""**

**import os**

**from pathlib import Path**

**# Build paths inside the project like this: BASE\_DIR / 'subdir'.**

**BASE\_DIR = Path(\_\_file\_\_).resolve().parent.parent**

**# Quick-start development settings - unsuitable for production**

**# See https://docs.djangoproject.com/en/5.0/howto/deployment/checklist/**

**# SECURITY WARNING: keep the secret key used in production secret!**

**SECRET\_KEY = 'django-insecure-==m^$sc-j=od(ouig=p6t2g))2!6$7p&lb=^^10a5ryg0sj^lu'**

**# SECURITY WARNING: don't run with debug turned on in production!**

**DEBUG = True**

**ALLOWED\_HOSTS = []**

**# Application definition**

**INSTALLED\_APPS = [**

**'django.contrib.admin',**

**'django.contrib.auth',**

**'django.contrib.contenttypes',**

**'django.contrib.sessions',**

**'django.contrib.messages',**

**'django.contrib.staticfiles',**

**'app',**

**]**

**MIDDLEWARE = [**

**'django.middleware.security.SecurityMiddleware',**

**'django.contrib.sessions.middleware.SessionMiddleware',**

**'django.middleware.common.CommonMiddleware',**

**'django.middleware.csrf.CsrfViewMiddleware',**

**'django.contrib.auth.middleware.AuthenticationMiddleware',**

**'django.contrib.messages.middleware.MessageMiddleware',**

**'django.middleware.clickjacking.XFrameOptionsMiddleware',**

**]**

**ROOT\_URLCONF = 'yatrisafar.urls'**

**TEMPLATES = [**

**{**

**'BACKEND': 'django.template.backends.django.DjangoTemplates',**

**'DIRS': [os.path.join(BASE\_DIR,'templates')],**

**'APP\_DIRS': True,**

**'OPTIONS': {**

**'context\_processors': [**

**'django.template.context\_processors.debug',**

**'django.template.context\_processors.request',**

**'django.contrib.auth.context\_processors.auth',**

**'django.contrib.messages.context\_processors.messages',**

**],**

**},**

**},**

**]**

**WSGI\_APPLICATION = 'yatrisafar.wsgi.application'**

**# Database**

**# https://docs.djangoproject.com/en/5.0/ref/settings/#databases**

**DATABASES = {**

**'default': {**

**'ENGINE': 'django.db.backends.sqlite3',**

**'NAME': BASE\_DIR / 'db.sqlite3',**

**}**

**}**

**# Password validation**

**# https://docs.djangoproject.com/en/5.0/ref/settings/#auth-password-validators**

**AUTH\_PASSWORD\_VALIDATORS = [**

**{**

**'NAME': 'django.contrib.auth.password\_validation.UserAttributeSimilarityValidator',**

**},**

**{**

**'NAME': 'django.contrib.auth.password\_validation.MinimumLengthValidator',**

**},**

**{**

**'NAME': 'django.contrib.auth.password\_validation.CommonPasswordValidator',**

**},**

**{**

**'NAME': 'django.contrib.auth.password\_validation.NumericPasswordValidator',**

**},**

**]**

**# Internationalization**

**# https://docs.djangoproject.com/en/5.0/topics/i18n/**

**LANGUAGE\_CODE = 'en-us'**

**TIME\_ZONE = 'UTC'**

**USE\_I18N = True**

**USE\_TZ = True**

**# Static files (CSS, JavaScript, Images)**

**# https://docs.djangoproject.com/en/5.0/howto/static-files/**

**STATIC\_URL = 'static/'**

**MEDIA\_URL="/media/"**

**MEDIA\_ROOT=os.path.join(BASE\_DIR,'media')**

**STATICFILES\_DIRS=[**

**os.path.join(BASE\_DIR,'static'),**

**]**

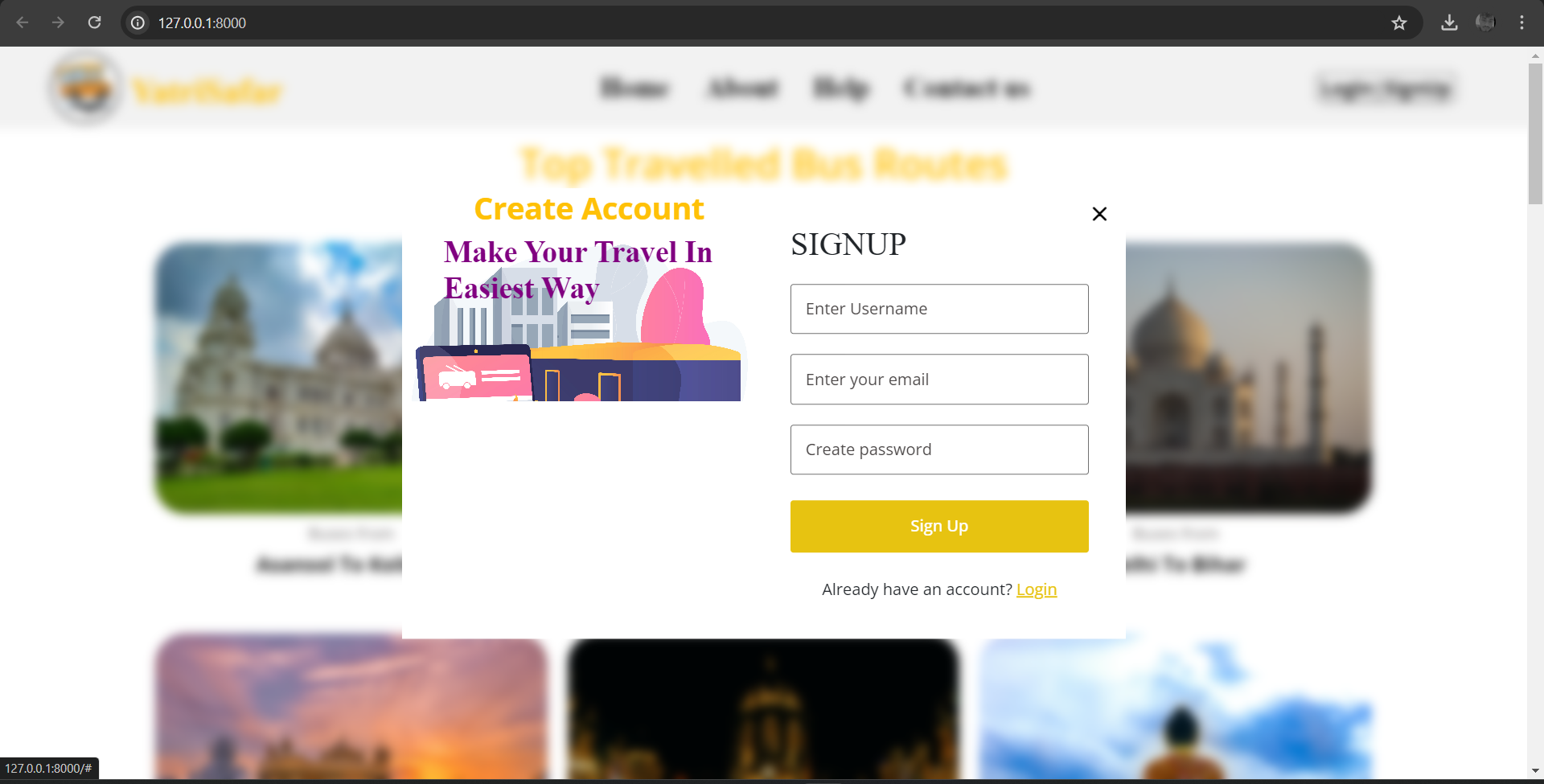
**# Default primary key field type**

**# https://docs.djangoproject.com/en/5.0/ref/settings/#default-auto-field**

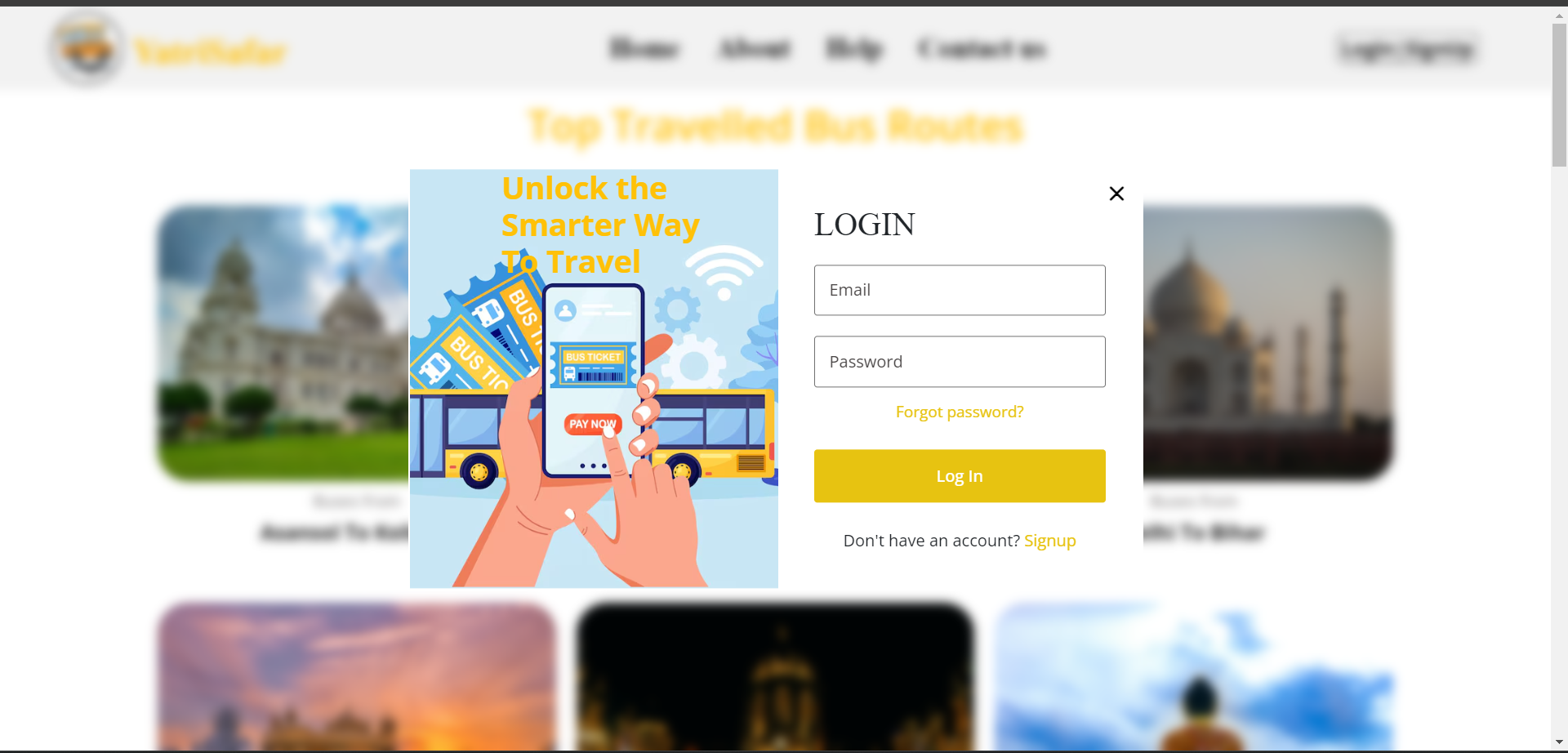
**DEFAULT\_AUTO\_FIELD = 'django.db.models.BigAutoField'**

**14.7. Screenshots of Forms**

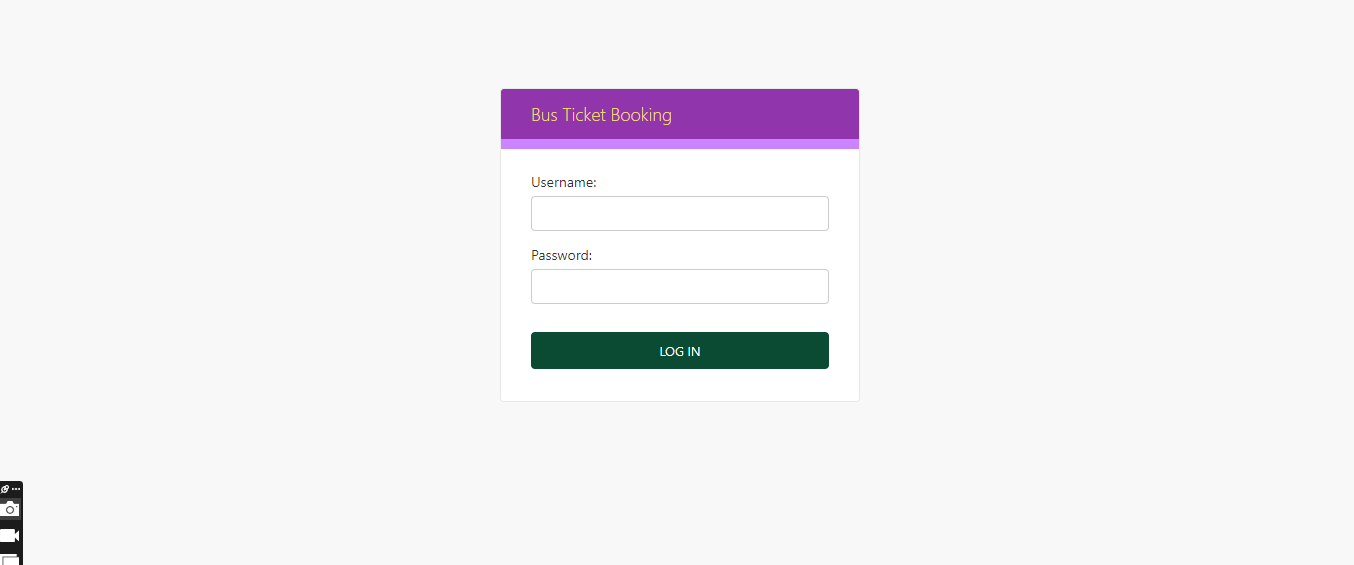
**User Signup Form**

****

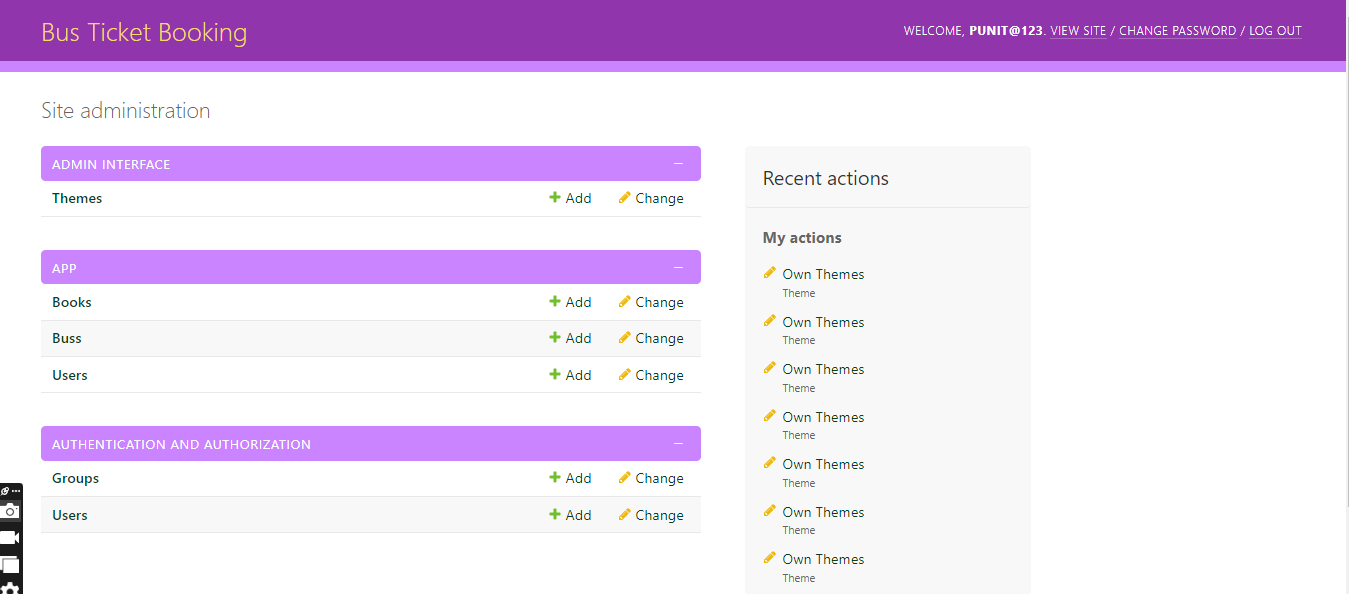
**User Login Form**

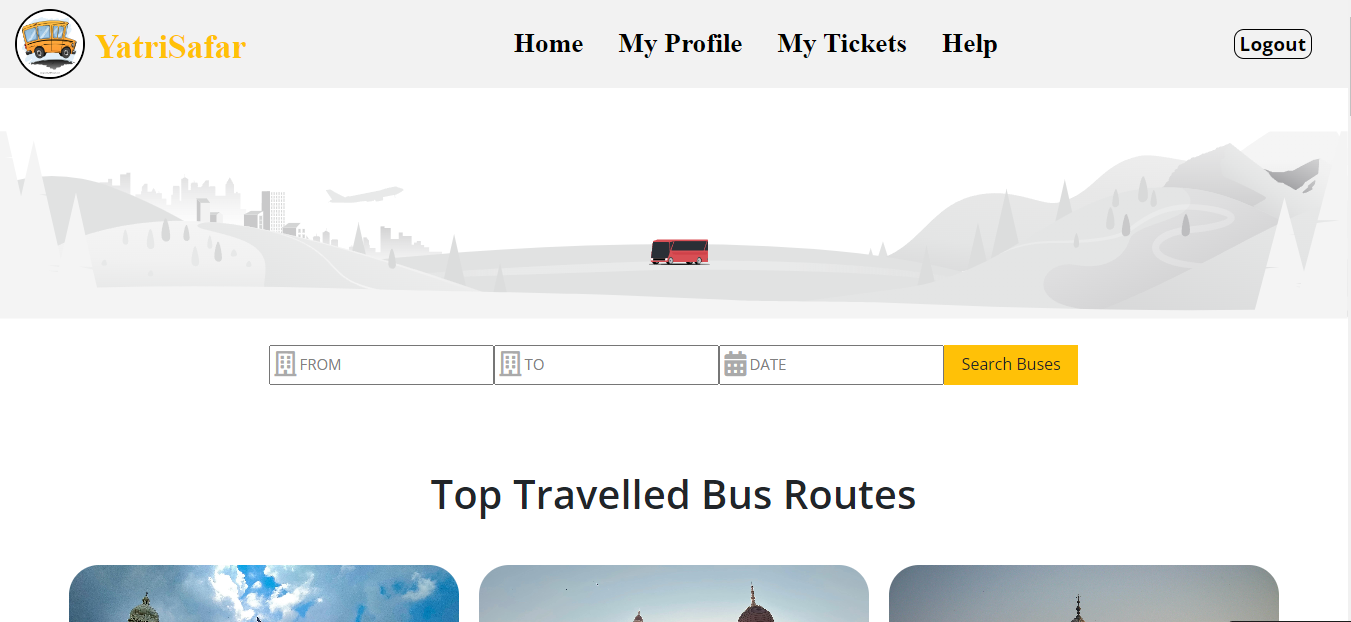


**Super Admin Login**

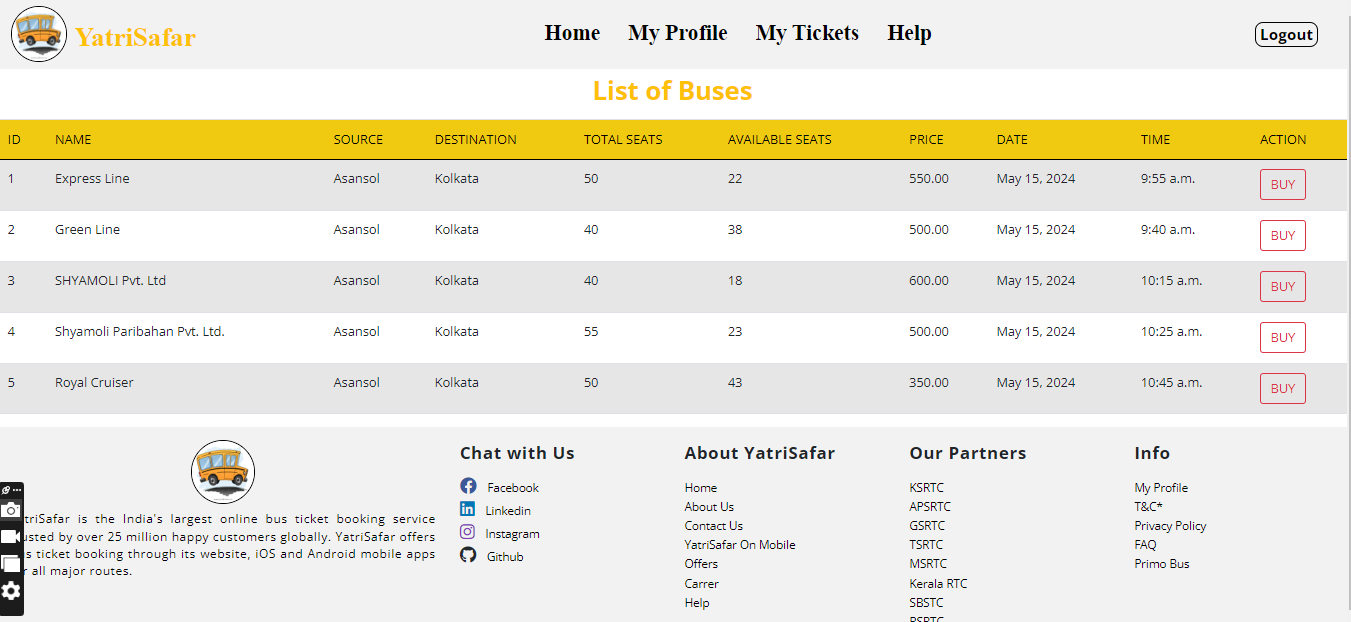
****

**Super Admin Home Page**

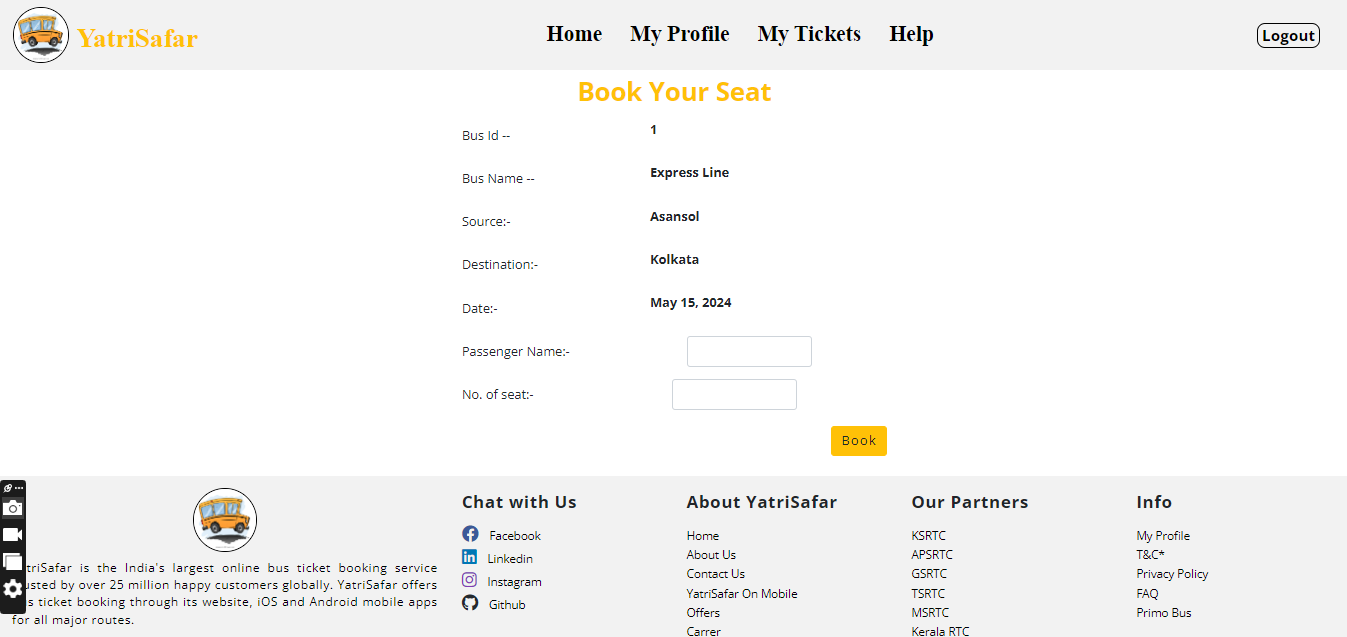
****

**User Home page**

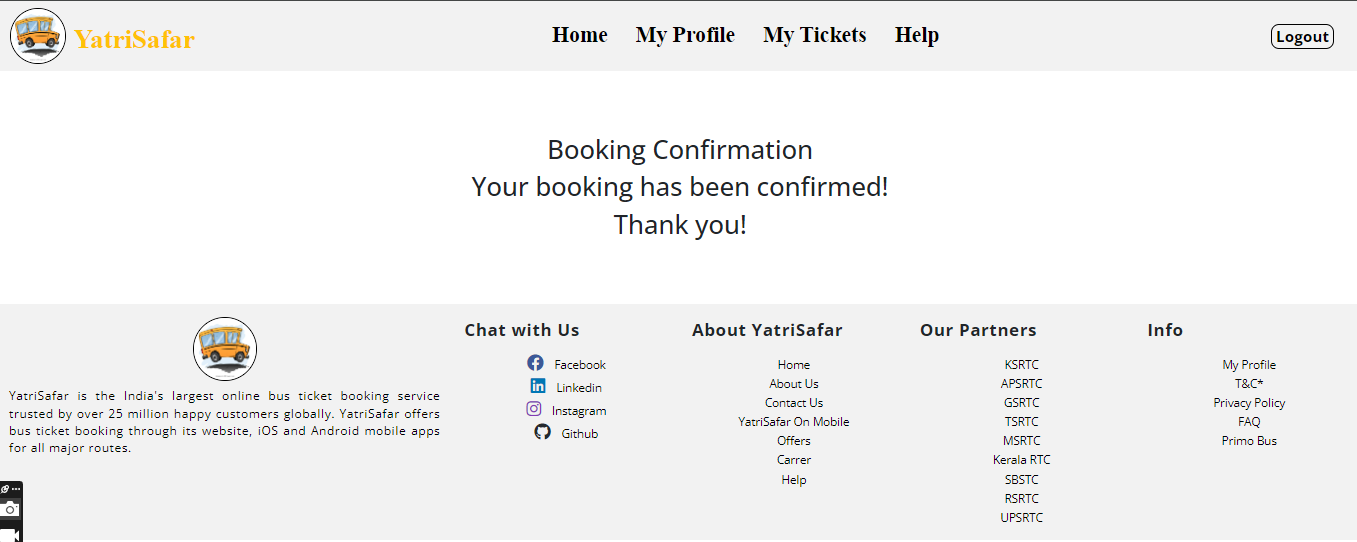
**Searches Bus List**

****

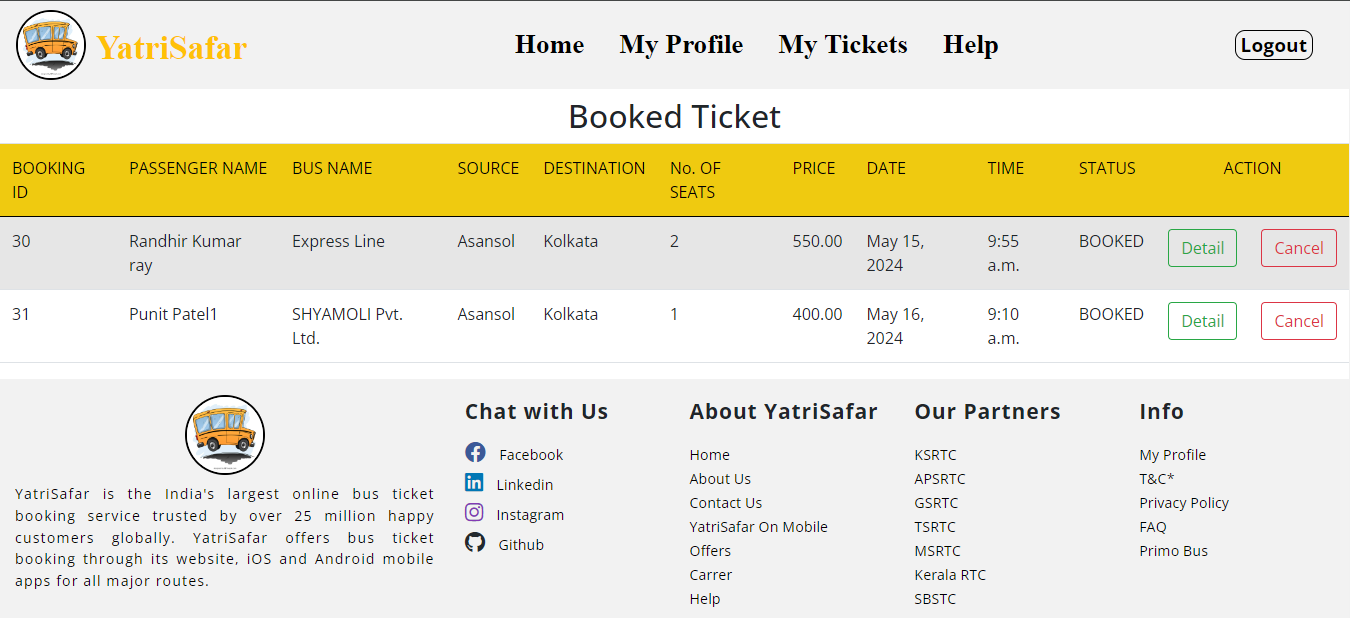
**Ticket Buying Process Page**

****

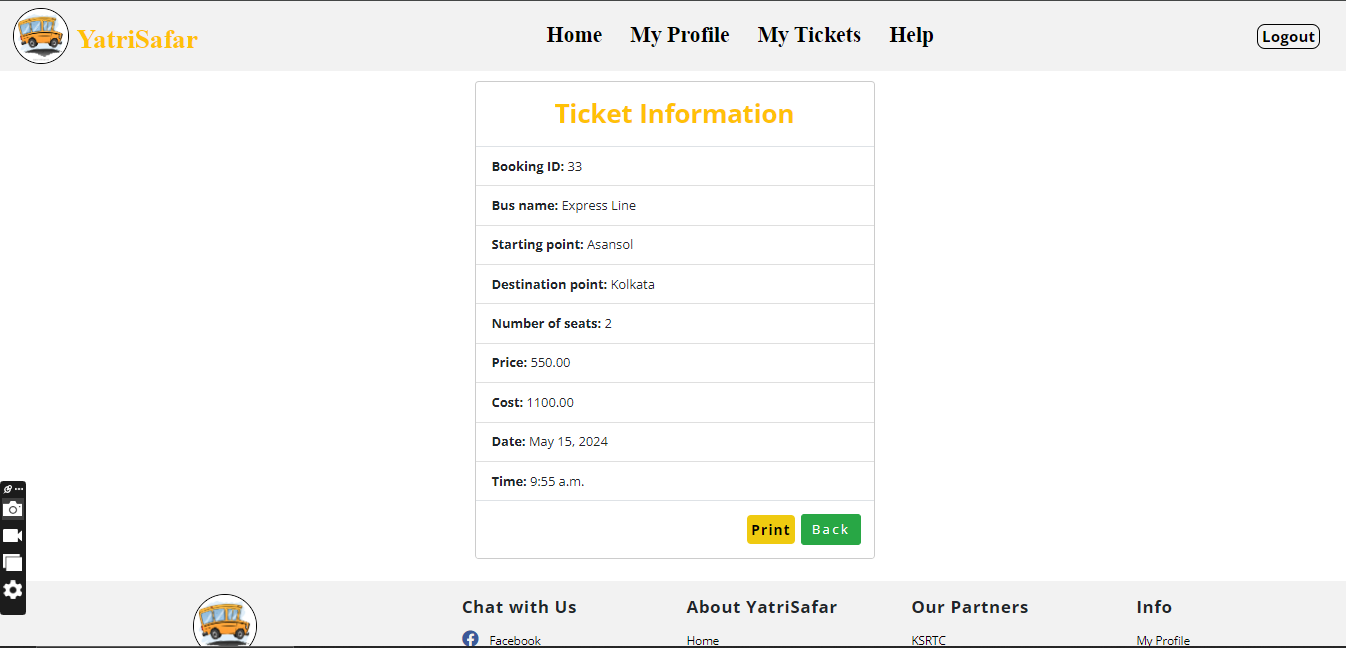
**Ticket Booking Confirmation Page**

****

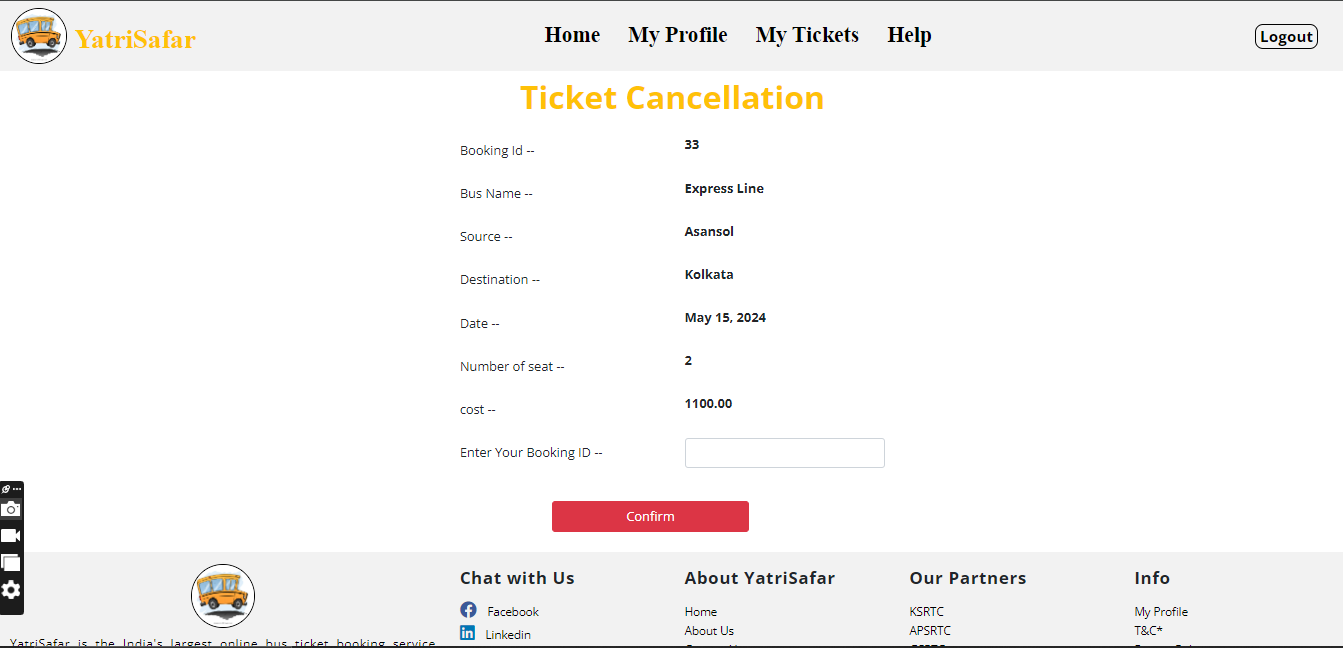
**Booked Ticket Page**

****

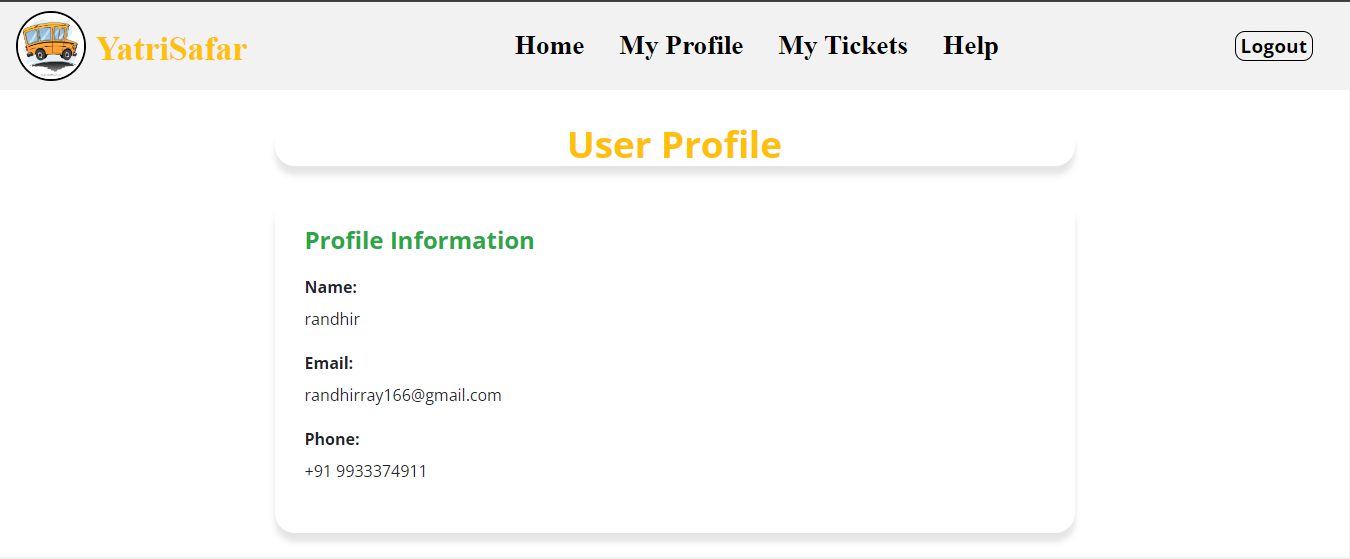
**Ticket Details Page**

****

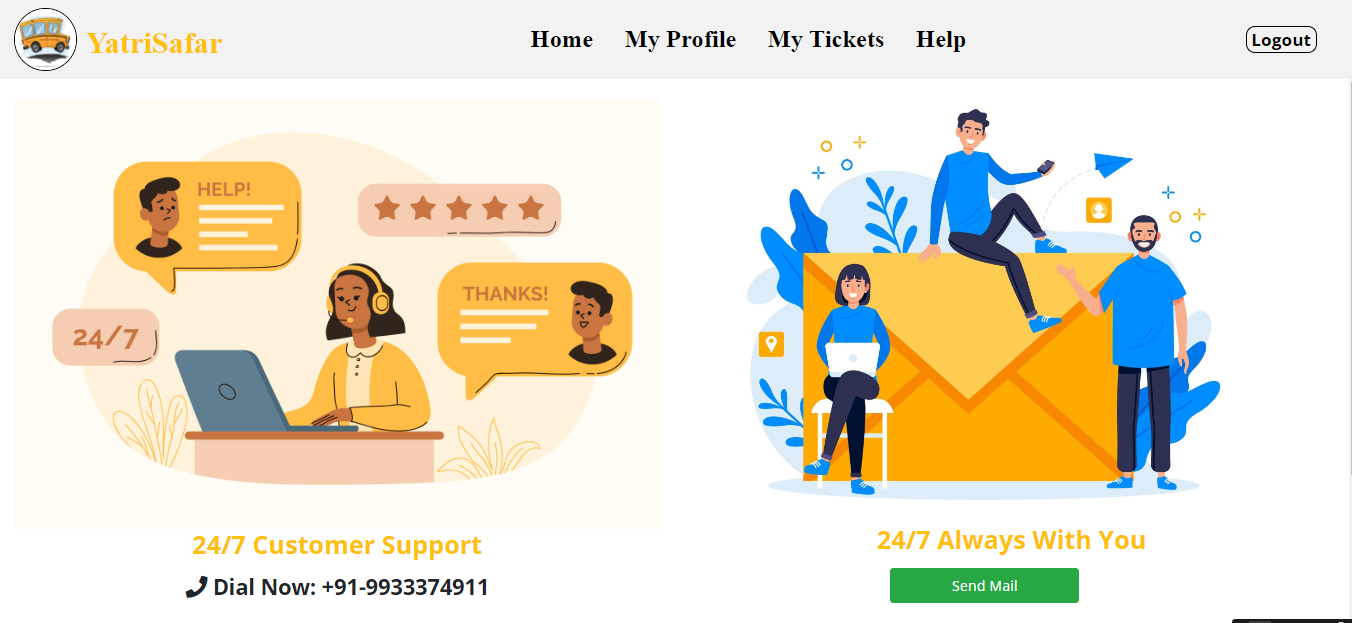
**Ticket Cancelling Page**

****

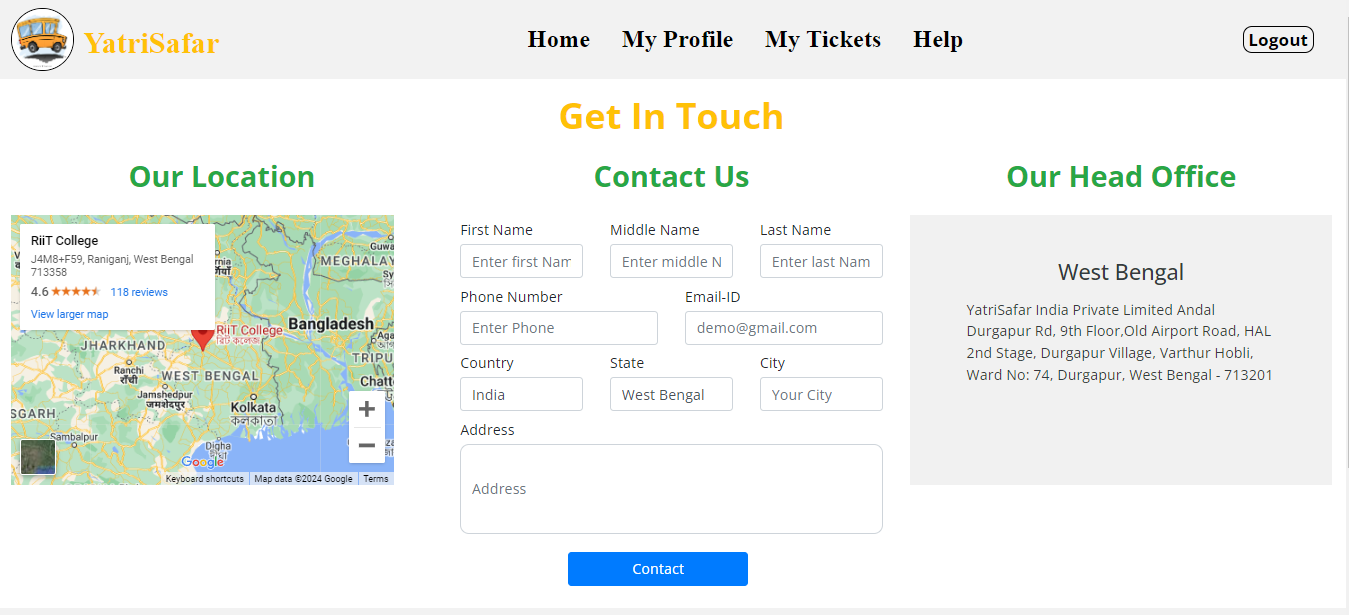
**User Profile Page**

****

**Help Page**

****

**Contact Us Page**

****

**14.8. Methodology used for testing**

* **Unit Testing:** A Unit corresponds to a form/class in the package. Unit testing focuses on verification of the corresponding form or class. In this level we have tested all our forms/classes individually. This testing includes testing of control paths, interfaces, local data structures, logical decisions, boundary conditions, and error handling. From this testing we were able to save, retrieve, update, delete and the search records on a table.
* **Integration Testing**: Integration testing is used to verify the combination of the software modules. In this level, we have tested by combining all unit tested forms into a subsystem. Here we found that the subsystems are performing well.
* **System Testing:**System testing is used to verify, whether the developed system meets the requirements.
* **Acceptance Testing:**Acceptance is the part of the project by which the customer accepts the product. The system under consideration is tested for user acceptance by constantly keeping in touch with the system users at time of developing and making changes whenever required.

**14.9Test Report**

**Unit Testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test caseRef No** | | **TCT-001** | | |
|  | |  |  | |
| Functionality | | : | Login to the System(Online Bus Ticket Booking System) | |
|  | |  |  | |
| Expectedoutcome | | : | Theuser should not login to member’s area  .and some error message follow | |
|  | | | | |
| StepNo. | DataUsed | | | ActualOutcome |
| 1. | Click anthology in button  Withoutenteringusereamer password | | | An alert message cametoenter  username |
| 2. | Clickon the log in button  Afterenteringsome user name leaving password field blank | | | Analertmessagecame to enter  Password |
| 3. | Click on thelog in button  Afterenteringsomepassword but leaving user name field blank | | | Analert message cameto enter  username |
| 4. | Click on the log in button  Afterenteringsomewronguser namebut correct Password | | | Amessage displayed on Log in  Pageabout this |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Testcase Ref No** | | **TCT-002** | | |
|  | |  |  | |
| Functionality | | : | Enter valid Data for package searching | |
|  | |  |  | |
| Expectedoutcome | | : | The user will enter the dates, start and destination to search for package. The user also has to select the travel mode. | |
|  | | | | |
| StepNo. | DataUsed | | | ActualOutcome |
| 1. | Click on the search button  Withoutentering valid details | | | An alertmessage application ear son the vacant fields. |
| 2. | Click on the search button  Afterentering valid details | | | Package details will application ear in front of the user for selection. Once the user can select the package and book, the invoice will be generated. |

**14.10. Performance and Efficiency issues**

Code efficiency has been achieved through proper validation using various methods in Pythoncoding.Firstnodatacanbeadded, viewed, edited andelateddatabasewithoutloginorsession.Forthiswehave implanted session tracking techniques through Python. Codlings havebeen used to validate various forms to ensure correct data should enter in database.

**14.10.1.Conclusion**

In conclusion, the development and implementation of an online bus ticket booking system offer significant benefits to both passengers and bus operators. Through the integration of modern technologies and user-centric design, such a system streamlines the booking process, enhances convenience, and improves the overall travel experience.

Passengers benefit from the convenience of being able to search for routes, check seat availability, and make reservations from the comfort of their own devices, anytime and anywhere. They can choose their preferred seats, make secure payments, and receive instant confirmation of their bookings. Additionally, real-time information on schedules and fares empowers passengers to make informed decisions and plan their trips more efficiently.

For bus operators, an online booking system brings operational efficiencies, cost savings, and revenue opportunities. Automated ticketing processes reduce manual errors and administrative overhead, while real-time data analytics provide insights into customer preferences, booking trends, and revenue performance. By optimizing seat occupancy and improving customer satisfaction, bus operators can enhance their competitiveness in the market and maximize profitability.

**14.10.2. Future scope of the project**

**Integration of a payment gateway to offer diverse online payment options, enhancing user convenience and expanding revenue streams.**

**Implementation of email verification for enhanced security, reducing the risk of fraudulent activities and ensuring a trustworthy booking process.**

**Addition of a seat selection feature to provide users with greater control and personalization, improving overall user experience and satisfaction.**

**Incorporation of real-time chat support for immediate assistance, fostering better communication and resolving user queries promptly.**

**Enhancement of language support to cater to a wider audience, promoting inclusivity and extending the project's market reach.**

**Continuous improvement to address existing limitations, including refining payment processes, enhancing verification methods, providing more seat selection options, and optimizing user support features for a seamless booking experience.**

**Development of a mobile application for on-the-go booking convenience, catering to the increasing trend of mobile usage.**

**Implementation of analytics and reporting tools to gather insights on user behavior, booking patterns, and popular routes, aiding in decision-making and marketing strategies.**

**Introduction of a ratings and reviews system for users to share feedback and experiences, fostering transparency and trust within the community.**

**Expansion of the platform to include additional travel services such as hotel bookings, tour packages, or car rentals, creating a comprehensive travel ecosystem for users.**

**14.10.3.References**

* <http://www.asp.net/learn/data-access/tutorial-16-vb.aspx>
* [www.gliffy.com](http://www.gliffy.com)
* <http://www.locmetrics.com/>
* <http://en.wikipedia.org/wiki/Load_testing>
* <http://en.wikipedia.org/wiki/Unit_test>
* <http://en.wikipedia.org/wiki/Manual_testing>
* <https://docs.python.org/3>
* <https://w3school>
* <https://www.w3schools.com/>
* <https://youtube.com/>
* <https://docs.djangoproject.com/en/5.0/>
* <https://fontawesome.com/v4/icons/>
* <https://codepen.io/mosbth/pen/qBEeJpg>
* <https://getbootstrap.com/>
* <https://www.w3schools.com/css/>
* <https://www.w3schools.com/html/>
* <https://www.w3schools.com/js/>
* <https://www.w3schools.com/python/>