

Acknowledgment

We would like to extend our heartfelt appreciation to all those who have been instrumental in the realization of this project.

First and foremost, we want to express our deep gratitude to each other, **Makwana Jaydeep K. and Makwana Jaydeep D.** . Our collaboration, dedication, and complementary skills have been the cornerstone of this project's success. Working together has allowed us to combine our strengths and create something truly remarkable.

We also want to express our appreciation to our project supervisor, prof. Diler Sakshi, for their unwavering support, guidance, and invaluable insights. Their expertise played a pivotal role in shaping the direction of this research.

Lastly, we wish to acknowledge the countless individuals who participated in surveys, interviews, or provided data for this research. Your contributions were invaluable in ensuring the successful completion of our project.

We are grateful to everyone who played a part, no matter how big or small, in making this endeavor possible.

Thank you all.

Preface

In an era defined by rapid technological advancement and evolving consumer preferences, the transportation industry has witnessed a profound transformation. Gone are the days of standing in lengthy queues or navigating through traffic to secure a seat on a bus for our journeys. The digital age has ushered in a new era of convenience, accessibility, and efficiency in the way we book tickets for our travels.

This project report unveils the culmination of our efforts in designing and developing an "Online Bus Ticket Booking System." It represents a journey that commenced with a fundamental question: How can we simplify and enhance the process of booking bus tickets, making it more convenient and enjoyable for travelers?

The online bus booking system is not just a technological solution; it is a manifestation of our commitment to improving the travel experiences of individuals who rely on buses as their mode of transportation. It bridges the gap between passengers and their desired destinations, streamlining the ticket booking process and elevating the overall bus travel experience.

Our gratitude also goes to the potential users of this system—the travelers—whose feedback and insights have played a crucial role in shaping the development of this platform. It is our aspiration that this system will not only meet but exceed your expectations, providing you with a seamless and enjoyable bus ticket booking experience. Safe travels!

The purpose and significance of the project

The "Online Bus Ticket Booking System" project is fueled by a clear mission: to redefine the bus ticket booking process and introduce unparalleled convenience to users. Its primary goal is to simplify the often intricate task of securing bus tickets. By eliminating the need for physical queues and facilitating bookings through digital platforms, this system provides users with a level of convenience they have long desired. In addition to improving convenience, the system offers real-time information on bus schedules, seat availability, and pricing. This real-time accessibility empowers users to swiftly make informed decisions, ensuring they can reserve tickets for their preferred routes without the challenges associated with traditional ticket booking methods. Moreover, the project aims to streamline ticket management, minimizing errors and ensuring a more seamless overall ticketing experience. With a user-friendly interface and intuitive features like seat selection and secure payment processing, the project's mission is to make the entire bus travel experience enjoyable from start to finish.

Beyond convenience, the "Online Bus Booking" project carries significant implications for both travelers and the transportation industry as a whole. One of its key advantages is the enhancement of customer satisfaction, addressing the challenges associated with traditional bus booking methods. This improvement in the passenger experience has the potential to foster loyalty and encourage repeat business. For bus operators, the project emerges as a pivotal revenue-generating tool. It optimizes seat utilization, facilitates dynamic pricing strategies, and mitigates the risk of overbooking, ultimately leading to increased profitability. Additionally, the project's robust data collection and analytics capabilities provide valuable insights into user preferences and industry trends, enabling more effective route planning, marketing strategies, and overall improvements in the bus travel experience.

Index

No.	Topic	Page No.
1	Introduction	5
2	Project Profile	9
3	System Development life cycle	20
4	Analysis Requirement	26
5	Feasibility Study	28
6	Hardware & software	35
7	Flow chart	36
8	Bar chart	38
9	Time Line	39
10	Gantt Chart	40
11	Data Flow Diagram	41
12	Use Case Diagram	47
13	Data Structure	49
14	Screen Layout and Body	52
15	Error and Solution	80
16	Testing	81
17	Limitation	82
18	Conclusion	83
19	Reference	84

Introduction

Online Bus Booking

The Online Bus Booking Application, developed with ASP.NET and supported by SQL Server, is a user-friendly web tool that streamlines the reservation of bus tickets. This application enables passengers to easily book tickets in advance, access real-time information on bus schedules, choose preferred seats, and enjoy a smooth booking experience.

Effortless Ticket Reservation: Users can seamlessly reserve bus tickets online, eliminating the need for time-consuming physical queues and providing a convenient booking solution.

1. What is our project

- Our project involves developing an "Online Bus Booking Solution."
- It is a digital platform designed to enable users to book bus tickets conveniently and efficiently through the internet.
- The "Online Bus Booking Solution" is a sophisticated web-based tool designed to transform the way individuals reserve and purchase tickets for their bus journeys. Leveraging cutting-edge technology, it provides users with a seamless, user-friendly interface accessible through web browsers and mobile devices.

2. Aim of our project

- The core objective of our project is to introduce a modern and efficient ticket booking solution that caters to the evolving needs of today's travelers. By integrating real-time data feeds, users can instantly access comprehensive information on current bus schedules, upcoming journeys, and corresponding seat availability and pricing details. This empowers users to make informed decisions swiftly, optimizing their bus travel experience..
- Furthermore, our project aims to revolutionize ticket management for bus operators. It introduces an intuitive backend system that enables operators to effortlessly manage ticket sales, monitor seat occupancy, and implement dynamic pricing strategies. Through this, we aim to enhance operational efficiency and revenue optimization for bus service providers.

3. Advantages of our project

- **Convenience:** Our system offers unparalleled convenience by allowing users to browse bus schedules, select preferred departure times, and choose specific seats, all from the comfort of their homes or while on the move. This eliminates the need for physical presence at bus terminals and reduces waiting times.
- **Real-Time Access:** Users gain instant access to a dynamic database of bus schedules, ensuring that they are always up-to-date with the latest offerings. This feature guarantees that users can secure tickets for their desired journeys, even for high-demand routes or limited-capacity services.

- **Efficient Ticket Management:** For bus operators, our system presents an integrated solution that minimizes manual intervention and reduces the likelihood of errors. Through the intuitive dashboard, operators can monitor ticket sales, track seat occupancy, and implement customized pricing strategies based on demand trends.
- **Enhanced User Experience:** The system offers a range of features designed to elevate the overall user experience. This includes interactive route maps, allowing users to select their preferred routes, as well as secure payment processing, which guarantees a seamless and trustworthy transaction process.

4. Disadvantages of our project

- **Dependence on Technology:** Dependence on Technology: While our system provides a convenient solution for tech-savvy individuals, it may inadvertently exclude potential users who lack access to the necessary digital infrastructure, such as stable internet connections or compatible devices.
- **Security Concerns:** As with any online platform handling sensitive information, security is paramount. Robust security measures, including encryption protocols and secure authentication processes, must be implemented to safeguard user data from potential cyber threats.
- **Technical Issues:** Despite our best efforts, users may encounter occasional technical glitches or system downtimes. These issues could arise from factors such as server outages, software updates, or unexpected spikes in user traffic. It is imperative to have a responsive support system in place to address and resolve such concerns promptly.

5. Reference websites

- in.redbus.com
- goibibo.com
- makemytrip.com/bus
- yatra.com/buses
- paytm.com/bus
- cleartrip.com/buses
- ixigo.com/bus

Project Profile

Project Title	<i>Online Bus Booking</i>
Project Description	<i>Online Bus Booking</i> is a user-friendly web tool that streamlines the reservation of bus tickets.
Class	<i>BCA sem-6</i>
Front-End	<i>ASP.NET, JavaScript, CSS, Bootstrap, JQuery</i>
Back-End	<i>C#, SQL Server</i>
Project Guide	<i>Prof.Diler Sakshi</i>
Developed By	<i>Makwana Jaydeep K. & Makwana Jaydeep D.</i>
Software	<i>Visual Studio 2013</i>
Other Software	<i>Microsoft Word, Microsoft Excel, Microsoft Power Point</i>
Submitted To	<i>Saurashtra University, Rajkot</i>

Software Study

What is C#?

C# (pronounced "C sharp") is a versatile, modern, and object-oriented programming language developed by Microsoft within its .NET framework. Here are some key aspects and features that define C#:

- 1. Object-Oriented:** C# is a fully object-oriented programming language, meaning it supports principles like encapsulation, inheritance, and polymorphism. Objects, which encapsulate data and behaviour, are fundamental building blocks in C# programming.
- 2. Managed Code:** C# is a managed code language, and its programs are executed in a runtime environment called the Common Language Runtime (CLR). The CLR provides services such as memory management, exception handling, and security, making C# programs more robust and secure.
- 3. Syntax:** C# syntax is similar to that of other C-based languages like C++ and Java. This makes it relatively easy for developers familiar with those languages to transition to C#. The syntax is designed to be readable and expressive.
- 4. Type-Safe:** C# is a strongly-typed language, which means variable types must be declared before they are used. This helps catch type-related errors at compile-time rather than runtime, contributing to better code reliability.
- 5. Platform-Independent:** While initially developed for Windows applications, C# has become more platform-independent with the introduction of .NET Core and later .NET 5 and .NET 6. These versions enable the development of cross-platform applications that can run on Windows, Linux, and macOS.
- 6. Integrated Development Environment (IDE):** Visual Studio is the primary IDE (Integrated Development Environment) for C# development. It offers a rich set of tools, a powerful debugger, and features that enhance productivity.

- 7. Component-Oriented:** C# supports component-oriented programming through features like properties, events, and delegates. This enables the creation of reusable and modular components, facilitating the development of scalable applications.
- 8. Asynchronous Programming:** C# provides asynchronous programming features using the ``async`` and ``await`` keywords. This allows developers to write non-blocking code, enhancing the responsiveness of applications, especially in scenarios involving I/O operations.
- 9. LINQ (Language-Integrated Query):** LINQ is a powerful feature in C# that enables developers to query data from various sources (such as databases, collections, or XML) directly within the language, enhancing code readability and reducing the need for complex loops.
- 10. Community Support:** C# has a vibrant and active community, with a wealth of resources, forums, and documentation available. The community contributes to the language's growth, and developers often share their knowledge and best practices.

Overall, C# is a versatile and robust programming language used for developing a wide range of applications, including desktop applications, web applications, mobile apps (using Xamarin), and game development (using Unity3D).

What is SQL Server?

SQL Server is a relational database management system (RDBMS) developed by Microsoft. It is a robust and comprehensive platform designed to efficiently manage and organize large amounts of data. SQL Server is widely used in enterprise environments for various applications, including data storage, retrieval, and management within organizations.

Here are key features and aspects of SQL Server:

- 1. Relational Database Management System (RDBMS):** SQL Server is a relational database system, meaning it organizes data into tables with rows and columns, and establishes relationships between these tables. This relational structure provides a flexible and efficient way to store and retrieve structured data.
- 2. Structured Query Language (SQL):** SQL Server uses the SQL language for database management. SQL allows users to define, manipulate, and query the data stored in the database. Common SQL operations include selecting, inserting, updating, and deleting records.
- 3. Data Integrity and Constraints:** SQL Server ensures data integrity through the implementation of various constraints, such as primary keys, foreign keys, unique constraints, and check constraints. These constraints help maintain the accuracy and consistency of the data stored in the database.
- 4. Transaction Management:** SQL Server supports transactions, which are sequences of one or more SQL statements treated as a single unit of work. This ensures that either all the operations in the transaction are completed successfully, or none of them are applied, maintaining the consistency of the database.
- 5. Scalability and Performance:** SQL Server is designed to handle large volumes of data and high concurrent user access. It provides features like indexing, partitioning, and parallel processing to optimize query performance. Additionally, SQL Server supports clustering and replication for achieving high availability and scalability.
- 6. Security:** SQL Server offers robust security features to protect data. This includes authentication, authorization, encryption, and auditing capabilities. Users and roles can be defined with specific permissions to control access to databases and their objects.

- 7. Business Intelligence (BI):** SQL Server includes a suite of tools for business intelligence, data warehousing, and data analysis. These tools include SQL Server Analysis Services (SSAS), SQL Server Reporting Services (SSRS), and SQL Server Integration Services (SSIS), enabling users to perform tasks related to analytics, reporting, and data integration.
- 8. Integration with Development Tools:** SQL Server integrates seamlessly with various development tools, with Visual Studio being one of the primary environments for developing applications that interact with SQL Server databases. It also supports languages like C# for creating stored procedures and functions.
- 9. Cloud Integration:** Microsoft offers cloud-based versions of SQL Server, such as Azure SQL Database and Azure SQL Managed Instance. These cloud services provide scalable and managed database solutions, allowing organizations to leverage the benefits of cloud computing.
- 10. Community Support and Documentation:** SQL Server has a large and active community, and Microsoft provides extensive documentation, forums, and resources to support developers and administrators working with SQL Server databases.

In summary, SQL Server is a powerful RDBMS that offers a wide range of features for efficiently managing and querying relational data, making it a popular choice for organizations of all sizes.

What is ASP.NET?

ASP.NET is a web development framework developed by Microsoft for building modern, dynamic, and scalable web applications. ASP.NET enables developers to build robust and interactive websites and web services by using a combination of programming languages, such as C# or Visual Basic, and a variety of web development tools and technologies.

Here are key aspects and features of ASP.NET:

- 1. Model-View-Controller (MVC) Architecture:** ASP.NET MVC is a framework within ASP.NET that promotes the separation of concerns in web application development. It divides the application into three main components: Model (data and business logic), View (user interface), and Controller (handles user input and manages flow).
- 2. Web Forms:** ASP.NET Web Forms is another programming model within ASP.NET that follows a more traditional event-driven model. It allows developers to create web pages with a drag-and-drop approach, making it easier to build complex user interfaces.
- 3. .NET Framework Integration:** ASP.NET is built on top of the .NET Framework, providing access to a rich set of libraries and features. Developers can use languages like C# or Visual Basic to build web applications, leveraging the benefits of a strongly-typed, object-oriented programming model.
- 4. Server Controls:** ASP.NET provides a wide range of server controls that encapsulate the behavior and appearance of common HTML elements. These controls simplify the development process by providing a higher level of abstraction and reducing the amount of code developers need to write.
- 5. State Management:** ASP.NET offers various mechanisms for managing state in web applications, including client-side and server-side options. This helps developers maintain information between different page requests and ensure a smooth user experience.
- 6. Security Features:** ASP.NET includes built-in security features such as authentication, authorization, and support for secure communication (HTTPS). It also provides features to prevent common web vulnerabilities, like Cross-Site Scripting (XSS) and Cross-Site Request Forgery (CSRF).

- 7. ASP.NET Core:** ASP.NET Core is the cross-platform, open-source version of ASP.NET. It is designed to run on Windows, Linux, and macOS and provides a modular and lightweight framework for building modern, cloud-based, and microservices-oriented applications.
- 8. Web API:** ASP.NET Web API allows developers to build RESTful APIs to enable communication between different web applications or mobile apps. It supports a variety of formats such as JSON and XML for data interchange.
- 9. Razor Pages:** Razor Pages is a lightweight alternative to the MVC pattern in ASP.NET, simplifying the development of single-page applications. It uses the Razor view engine and integrates seamlessly with existing ASP.NET features.
- 10. Integration with Front-End Technologies:** ASP.NET can be integrated with client-side libraries and frameworks like Angular, React, or Vue.js. This enables developers to build interactive and responsive user interfaces by combining the strengths of both server-side and client-side technologies.
- 11. Visual Studio Integration:** Microsoft Visual Studio is the preferred IDE for ASP.NET development, providing a rich set of tools for designing, coding, testing, and deploying web applications.

ASP.NET is widely used for building a variety of web applications, ranging from small websites to large enterprise-level applications. Its versatility, integration with the .NET ecosystem, and support for various development models make it a popular choice among developers.

What is Bootstrap?

Bootstrap is an open-source front-end framework developed by Twitter. It is a popular and widely used toolkit for building responsive, mobile-first web applications and websites. Bootstrap provides a set of HTML, CSS, and JavaScript components and styles that help developers create consistent, visually appealing, and functional web interfaces with minimal effort.

Features of Bootstrap:

1. **Responsive Design:** Bootstrap is designed with a mobile-first approach, meaning it prioritizes the design and development of web applications for smaller screens (e.g., smartphones and tablets) and then scales up to larger screens (e.g., desktops). This approach ensures that web pages adapt seamlessly to various screen sizes and devices.
2. **Pre-built Components:** Bootstrap offers a rich collection of pre-designed UI components, such as navigation bars, buttons, forms, modals, carousels, and more. These components can be easily customized and incorporated into web projects, saving developers time and effort.
3. **Grid System:** Bootstrap includes a responsive grid system that simplifies layout design. Developers can create flexible, grid-based page layouts by using rows and columns, which automatically adjust to fit different screen sizes. This grid system supports various screen sizes and breakpoints.
4. **Typography and Styles:** Bootstrap provides a consistent and visually appealing typography system, along with a set of CSS classes for styling elements like headings, paragraphs, and links. It offers a range of predefined color schemes and themes, making it easy to achieve a polished look and feel.
5. **JavaScript Plugins:** Bootstrap includes JavaScript plugins for enhancing user interactions and functionality. Examples include dropdown menus, tooltips, popovers, modals, and more. These plugins can be easily integrated into web applications without extensive coding.
6. **Customization:** Bootstrap is highly customizable. Developers can use Bootstrap's built-in Sass variables and mixins to modify default styles and adapt the framework to their project's specific design requirements. This flexibility allows for unique and branded designs.

7. **Community and Documentation:** Bootstrap has a large and active community of developers and users. It offers comprehensive documentation, tutorials, and examples to assist developers in using the framework effectively.
8. **Accessibility:** Bootstrap strives to maintain accessibility standards, ensuring that web applications created with Bootstrap are usable by individuals with disabilities. It provides ARIA roles and attributes to enhance accessibility.
9. **Browser Compatibility:** Bootstrap is compatible with most modern web browsers, including Chrome, Firefox, Safari, Edge, and Internet Explorer. It helps developers create consistent experiences across different browsers.
10. **Open Source:** Bootstrap is an open-source project, available under the MIT License. This means it is free to use, modify, and distribute, making it accessible to a wide range of developers and organizations.

Bootstrap has become a popular choice for web developers looking to accelerate the development process, improve design consistency, and create responsive web applications that work well on a variety of devices. It has a strong community and ecosystem of themes and extensions that further extend its capabilities.

What is JQuery?

jQuery is a fast, lightweight, and feature-rich JavaScript library designed to simplify and streamline the process of client-side web development. It provides a set of functions and utilities that make it easier to manipulate HTML documents, handle events, create animations, and manage asynchronous operations, among other tasks.

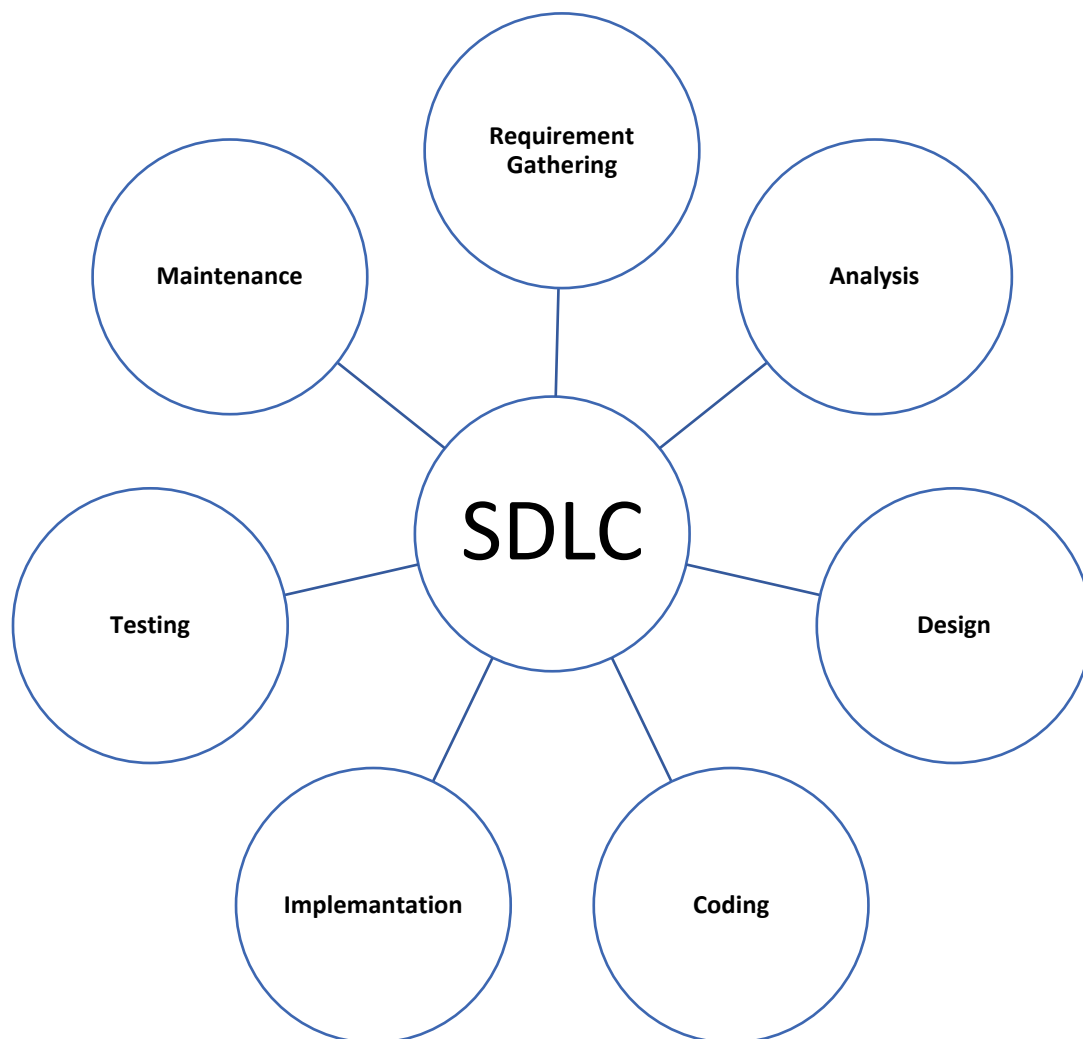
Key features and aspects of jQuery include:

1. **DOM Manipulation:** jQuery simplifies the process of manipulating the Document Object Model (DOM) of a web page. It provides easy-to-use methods for selecting, modifying, and traversing HTML elements.
2. **Event Handling:** jQuery simplifies the binding and handling of events, such as user clicks, keypresses, and form submissions. It allows developers to attach event handlers to elements with concise and readable code.
3. **Ajax (Asynchronous JavaScript and XML):** jQuery streamlines the implementation of Ajax functionality, making it easier to perform asynchronous requests to the server and update parts of a web page without requiring a full page reload.
4. **Animations and Effects:** jQuery includes built-in functions for creating animations and applying visual effects to elements on a web page. This includes features like fading, sliding, and toggling.
5. **Cross-Browser Compatibility:** jQuery abstracts many of the browser-specific differences, providing a consistent and reliable programming interface across various web browsers. This helps developers write code that works consistently on different platforms.
6. **Utilities:** jQuery offers a range of utility functions to simplify common programming tasks, such as working with arrays, handling asynchronous operations, and managing the execution of functions.
7. **AJAX Methods:** jQuery provides methods like `$.ajax()` that simplify the process of making asynchronous HTTP requests. This is useful for retrieving data from a server without reloading the entire page.
8. **Plugins:** jQuery has a vast ecosystem of plugins contributed by the community. These plugins extend the functionality of jQuery, offering additional features and capabilities.

9. **Chaining:** One of the distinctive features of jQuery is its method chaining. This allows developers to chain multiple method calls together, resulting in more concise and readable code.
10. **Easy Integration:** jQuery can be easily integrated into existing web projects. It works well with other JavaScript libraries and frameworks, allowing developers to use it alongside other tools.
11. **Open Source:** jQuery is open-source, which means its source code is freely available, and developers can modify and distribute it under the terms of the MIT License.

While jQuery has been widely used in the past, modern web development has seen the emergence of new JavaScript features and libraries. Developers often consider the use of more modern frameworks like React, Angular, or Vue.js for complex and interactive web applications. However, jQuery remains relevant and is still used in various projects, especially for quick prototyping and smaller-scale applications.

SYSTEM DEVELOPMENT LIFE CYCLE (SDLC)



SDLC (Software Development Life Cycle) is a systematic and structured approach to the development, deployment, and maintenance of software applications. It provides a framework for managing the entire software development process, from initial conception to final release and ongoing maintenance. SDLC emphasizes planning, design, coding, testing, and documentation to ensure the creation of high-quality, reliable, and maintainable software solutions. The specific phases and methodologies within SDLC may vary depending on the project's requirements and the chosen development approach.

is a systematic and structured approach to software development that outlines the stages, processes, and activities involved in designing, creating, testing, deploying, and maintaining software applications. SDLC is essential for ensuring that software projects are completed successfully, on time, and within budget, while also meeting the specified requirements and quality standards. It provides a framework for managing and controlling the software development process from start to finish.

Our project planning

No.	Project Task	No. of Days	Starting Date	Ending Date
1	Requirement Gathering	6	16-Dec-23	21-Dec-23
2	Requirement Analysis	6	22-Dec-23	27-jDec-23
3	Feasibility Study	6	28-Dec-23	02-Jan -24
4	Data Information	4	03-Jan -24	06-Jan -24
5	Data Structure	8	7-Jan -24	14-Jan -24
6	Data Flow Diagram: Admin & Client	5	15-Jan -24	19-Jan -24
7	Designing: Admin & Client	5	20-Jan -24	24-Jan -24
8	Coding	23	25-Jan -24	16-Feb -24
9	Error & Solutions	4	17-Feb -24	20-Feb -24
10	Testing	3	21-Feb -24	23-Feb -24
11	Implementation	2	24-Feb -24	25-Feb -24
12	Limitations	2	26-Feb -24	27-Feb-24
13	Conclusion	1	28-Feb-24	28-Feb-24

Software Development Life Cycle (SDLC) for Online Bus Booking

Project Overview:

1. Requirements Gathering and Analysis:

Conduct in-depth discussions with stakeholders to gather detailed project requirements.

Create a comprehensive requirements document that outlines functional and non-functional requirements.

Prioritize and validate requirements to ensure they align with project goals.

2. Planning:

Develop a project plan that includes scope, schedule, budget, and resource allocation.

Define project milestones, deliverables, and success criteria.

Identify potential risks and develop a risk management plan.

3. Design:

Create a technical architecture and system design based on the gathered requirements.

Design the user interface (UI) and user experience (UX) elements, if applicable.

Document the design decisions and technical specifications.

4. Implementation (Coding):

Begin the coding phase, writing code according to the design specifications.

Adhere to coding standards, best practices, and version control.

Conduct code reviews and testing throughout the development process.

5. Testing:

Develop and execute test plans, including unit testing, integration testing, system testing, and user acceptance testing (UAT).

Identify and document defects or issues and work on their resolution.

Ensure that the software meets the defined quality and performance standards.

6. Deployment (Release):

Prepare the software for deployment, including installation and configuration.

Perform data migration tasks, if applicable.

Plan and execute a controlled release to end-users or production environments.

7. Maintenance and Support:

Transition to the maintenance phase, where ongoing monitoring, bug fixes, and updates occur.

Provide technical support to users and address issues promptly.

Regularly assess system performance and security.

8. Documentation:

Maintain comprehensive documentation throughout the SDLC, including technical documentation, user manuals, and training materials.

Keep records of changes, updates, and version history.

9. Review and Evaluation:

Conduct regular project reviews and evaluations to assess progress, quality, and alignment with project goals.

Adjust project plans and activities based on feedback and lessons learned.

10. Closure:

Once the project is complete, formally close it by documenting the final deliverables, outcomes, and any remaining tasks.

Conduct a final project review to capture key insights and improvements for future projects.

Requirement Analysis

Requirement analysis is the process that focuses on task that determine the needs or condition to meet the new or altered product of project.

❖ **Project Scope:** The Online Bus Booking System will be a web-based application designed to facilitate users in browsing, booking, and managing bus tickets through an online platform. The system will provide a user-friendly interface allowing users to view available buses, schedules, and routes. Additionally, it will offer administrators the necessary tools to manage bus and route information efficiently.

❖ **User Requirements:**

- **User registration and login:** users should be able to create accounts, to login.
- **Search Buses:** users should be able to search Buses.
- **Booking Report:** users should be able to see their booking report.
- **Ticket Download:** users should be able to download tickets.

❖ **Administrator requirements:**

- **Add Bus :** admin should be able to add buses.
- **Route Details:** admin should be able to view route details.
- **Booking Report:** admin should be able to view all booking reports.
- **Bus Details Report:** admin should be able to view Bus Details Report.
- Admin should be able to add boarding points.
- Admin should be able to update bus details.
- Admin should be able to add bus schedule.

❖ **Technical analysis:**

- The system will be developed using a combination of web technologies including:
 - **Front-end: ASP.NET, Bootstrap, JQuery.**
 - **Back-end: C#, SQL Server**

❖ Security measures:

- Secure user authentication and authorization mechanism.
- Password hashing and encryption techniques for user data protection.

❖ Performance:

- The system should be able to handle a significant numbers of concurrent user requests without significant performance degradation.

❖ Compatibility:

- The application should be compatible with major web browsers and different devices, including desktops, tablets, and smartphones.

❖ Conclusion:

- the requirement analysis for the online Bus booking has identified the specific needs and expectations of users and administrators. This analysis will serve as a blueprint for the project and implement user-friendly application that meets the demands of users and provides an enhanced Bus booking experience.

Feasibility Study

A feasibility study is conducted to assess the viability and practicality of implementing a new system or project. In the case of an online Bus booking, the feasibility study evaluates various aspects to determine whether the project is worth pursuing. Here's a detailed feasibility study an online Bus booking system:

1. Technical Feasibility:

Assessment of the technical feasibility ensures that the required technology and infrastructure are available and can support the development and operation of the online Bus booking system.

- **Scalability and Performance:** Is the system scalable to handle a potentially large number of users, especially during peak booking times? Will it provide satisfactory performance with minimal downtime?

2. Economical Feasibility:

- Economical feasibility involves the estimation and the displaying the cost of expenses which was made to develop this website.
- The following tables shows the economical feasibility of our website.

Hardware Expenses

CPU	7500
Monitor	5500
Mouse	500
Keyboard	1000
Processer	11000

Software Expenses

Google Chrome	Open Source
Visual Studio	Open Source

3. Operational Feasibility:

- Operational feasibility measures how well organized your site is and how user will fill about the system.
- It measures that how well proposed the system solves the problem.
- Our research and study predict that all the functionalities provide to the user are extremely easy to use and our website is user friendly.
- The following are the functionality that are provide to the user and admin:

❖ Guest User:

- **Home**
- **About Us**
- **Search Buses**
- **Login**
- **Register**

❖ Registered User:

- **Home**
- **About Us**
- **Search Buses**
- **Book Tickets**
- **Booking Report**
- **Download Ticket**
- **Logout**

❖ Admin:

- **Add Bus**
- **Add Boarding Points**
- **Update Bus Details**
- **Add Bus Schedule**
- **Logout**

4. Resource Feasibility:

- In this feasibility we have to check we have complete resources for developing our project.
- If you don't have all resources you can't successfully develop your project.
- There are different types of resources like hardware or software.

Hardware Requirements

Processor	Intel core i3 and above
RAM	4GB
Hard Disk	200GB and above
Monitor	CRT or LCD
Keyboard	Normal or multimedia
Mouse	Compatible Mouse

Software Requirements

Front End	ASP.NET, CSS, JavaScript, Bootstrap, JQuery
Back End	C#, SQL Server
Operating System	Windows 7 or above
Browser	Google Chrome, Firefox

5.Schedule Feasibility:

- In this feasibility you have to create a chart in which you have to define functionality of your project and how much time that task take for successfully complete.

No.	Project Task	No. of Days	Starting Date	Ending Date
1	Requirement Gathering	6	16-Dec-23	21-Dec-23
2	Requirement Analysis	6	22-Dec-23	27-jDec-23
3	Feasibility Study	6	28-Dec-23	02-Jan -24
4	Data Information	4	03-Jan -24	06-Jan -24
5	Data Structure	8	7-Jan -24	14-Jan -24
6	Data Flow Diagram: Admin & Client	5	15-Jan -24	19-Jan -24
7	Designing: Admin & Client	5	20-Jan -24	24-Jan -24
8	Coding	23	25-Jan -24	16-Feb -24
9	Error & Solutions	4	17-Feb -24	20-Feb -24
10	Testing	3	21-Feb -24	23-Feb -24
11	Implementation	2	24-Feb -24	25-Feb -24
12	Limitations	2	26-Feb -24	27-Feb-24
13	Conclusion	1	28-Feb-24	28-Feb-24

❖ Conclusion:



Following a comprehensive feasibility study, if the online bus booking system is determined to be technically, economically, legally, and operationally viable, within a reasonable timeframe for development and deployment, it is deemed feasible to proceed with the project. The study serves as a crucial tool for making well-informed decisions and proactively addressing potential challenges during the early stages of development, thereby enhancing the likelihood of a successful implementation of the bus booking system.

Hardware & Software

Hardware

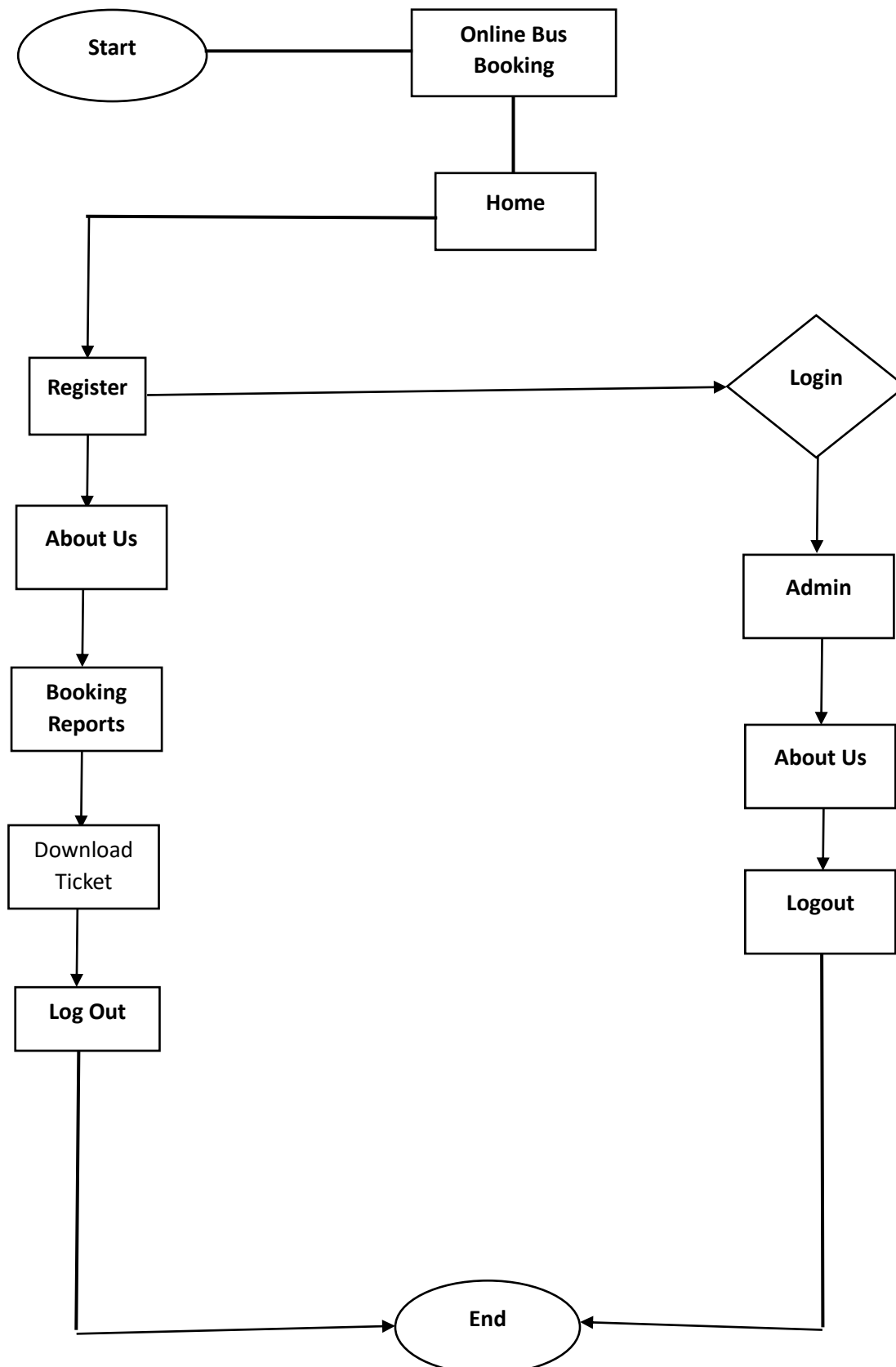
Processor	Intel core i3 and above
RAM	4GB
Hard Disk	200GB and above
Monitor	CRT or LCD
Keyboard	Normal or multimedia
Mouse	Compatible Mouse

Software

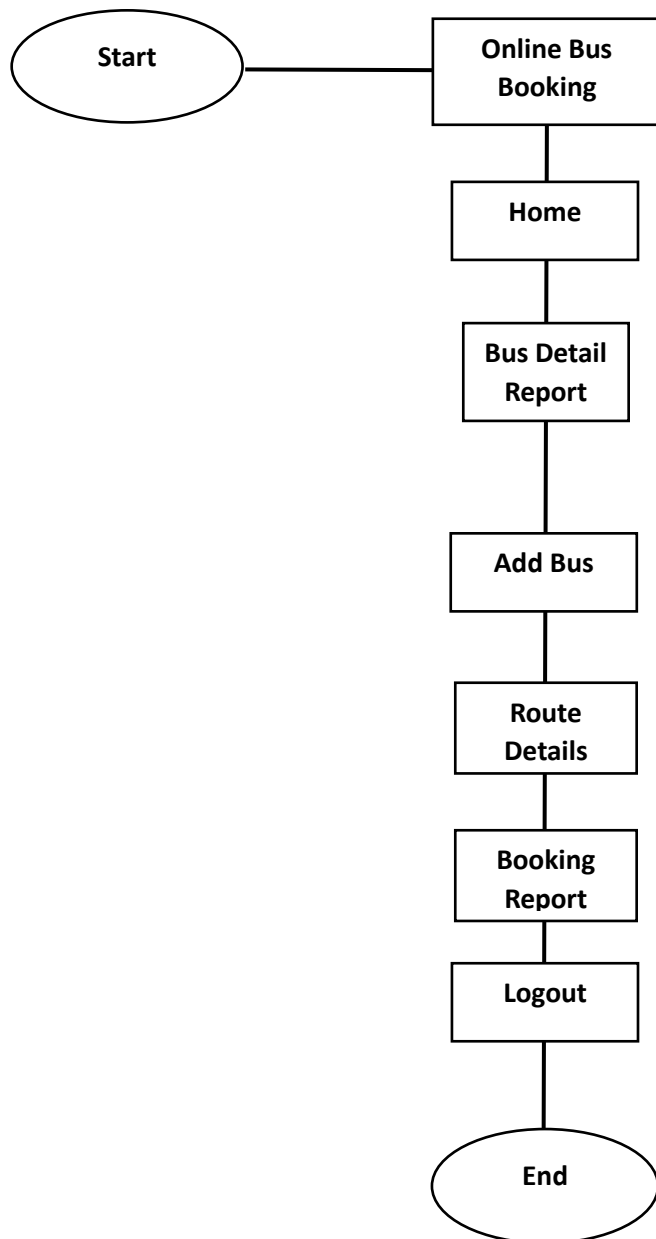
Front End	ASP.NET, CSS, JavaScript, Bootstrap, JQuery
Back End	C#, SQL Server
Operating System	Windows 7 or above
Browser	Google Chrome, Firefox

Flow Chart

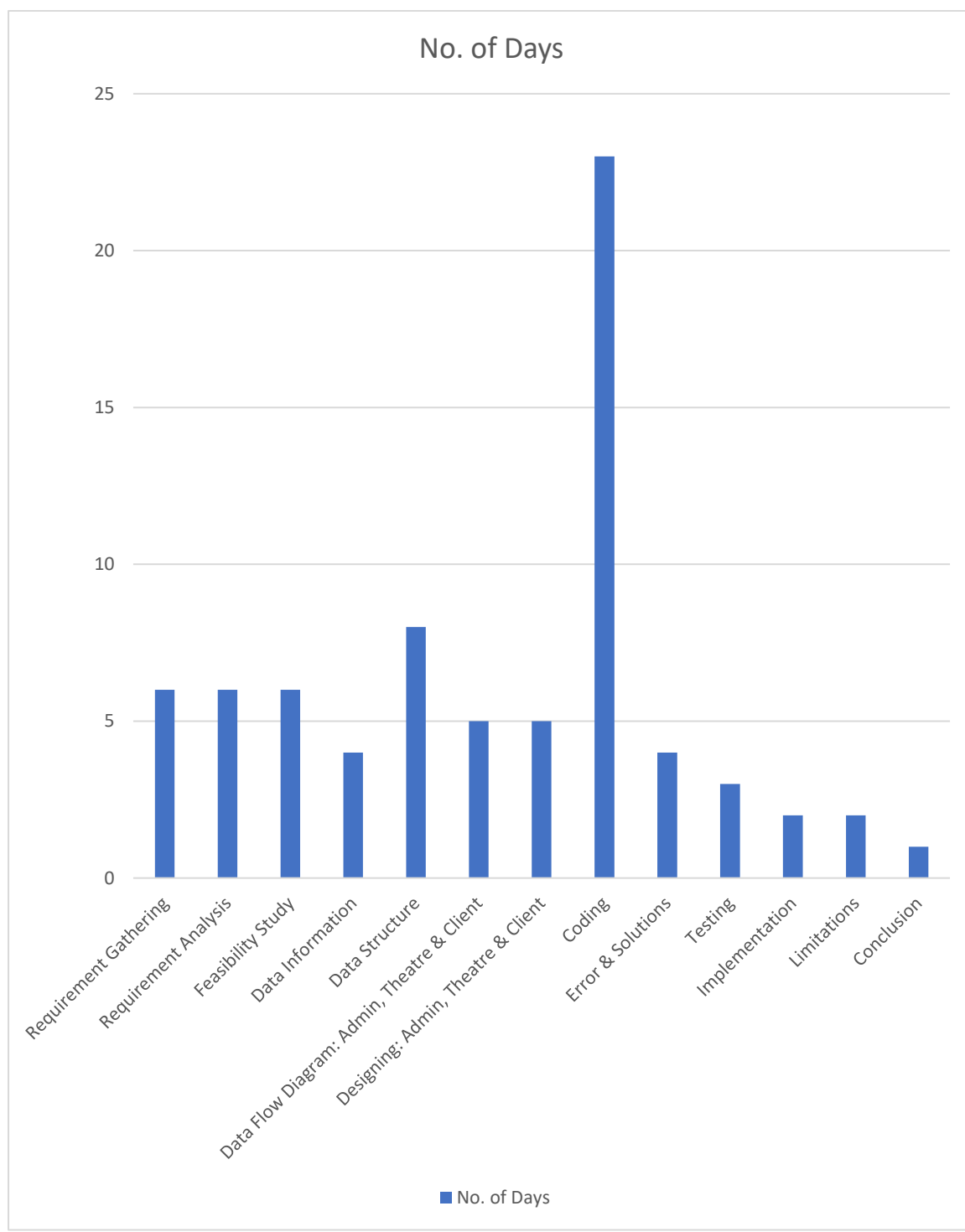
1.Client



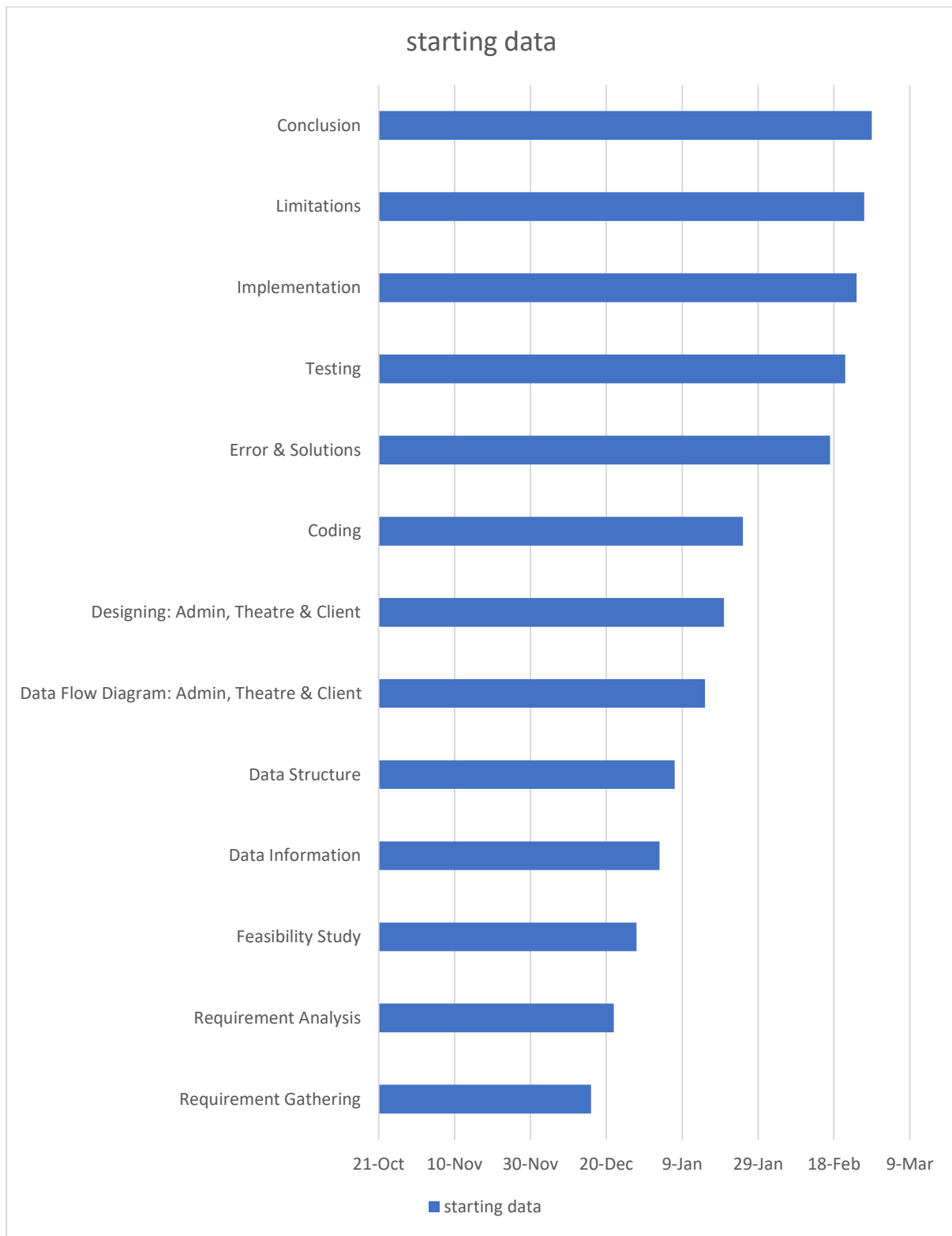
2.Admin



Bar Chart



Time Line



Gantt Chart



Data Flow Diagram

What is data flow diagram?

- Data flow diagram show the way of information flow through a process or system.
- It includes data input and output, data store, and various of processes the data moves through.
- Data flow diagram visually represents systems and processes that would be describe in a chunk of text.

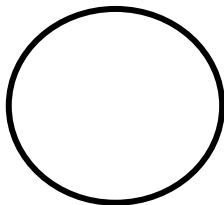
Symbols of data flow diagram



Flow Of Data



Input and output data



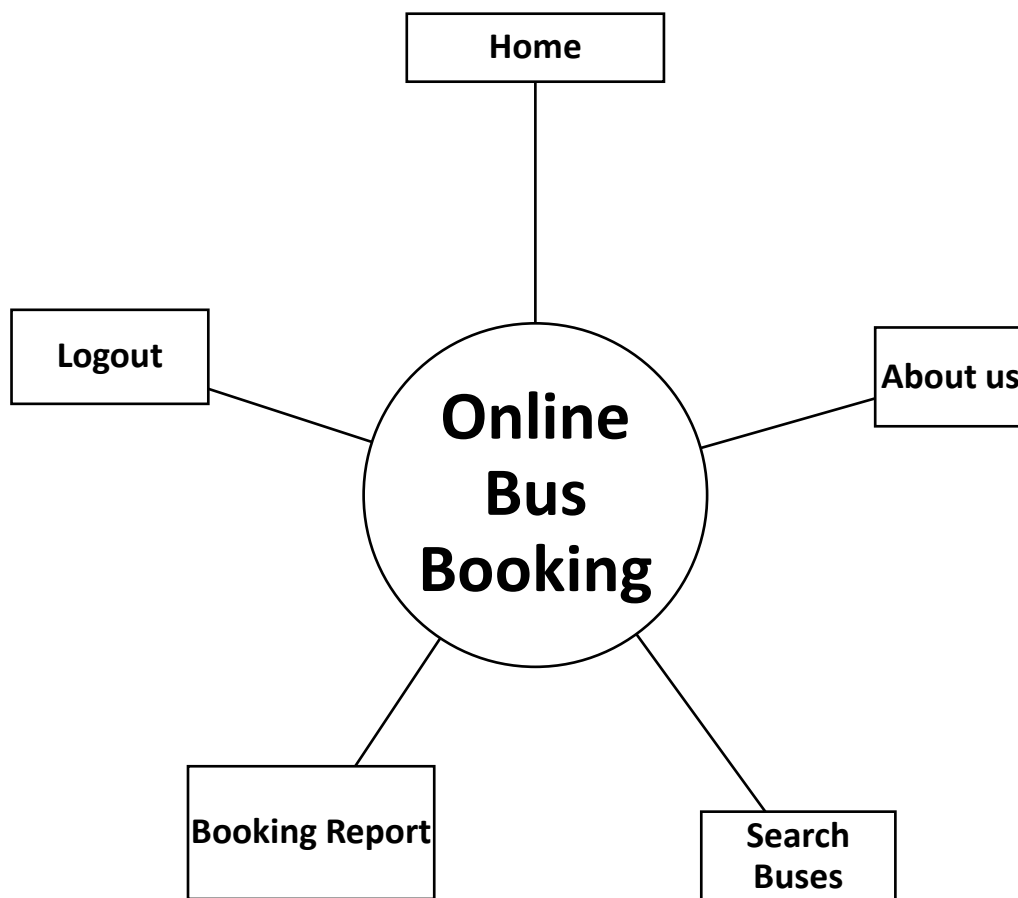
Process of data



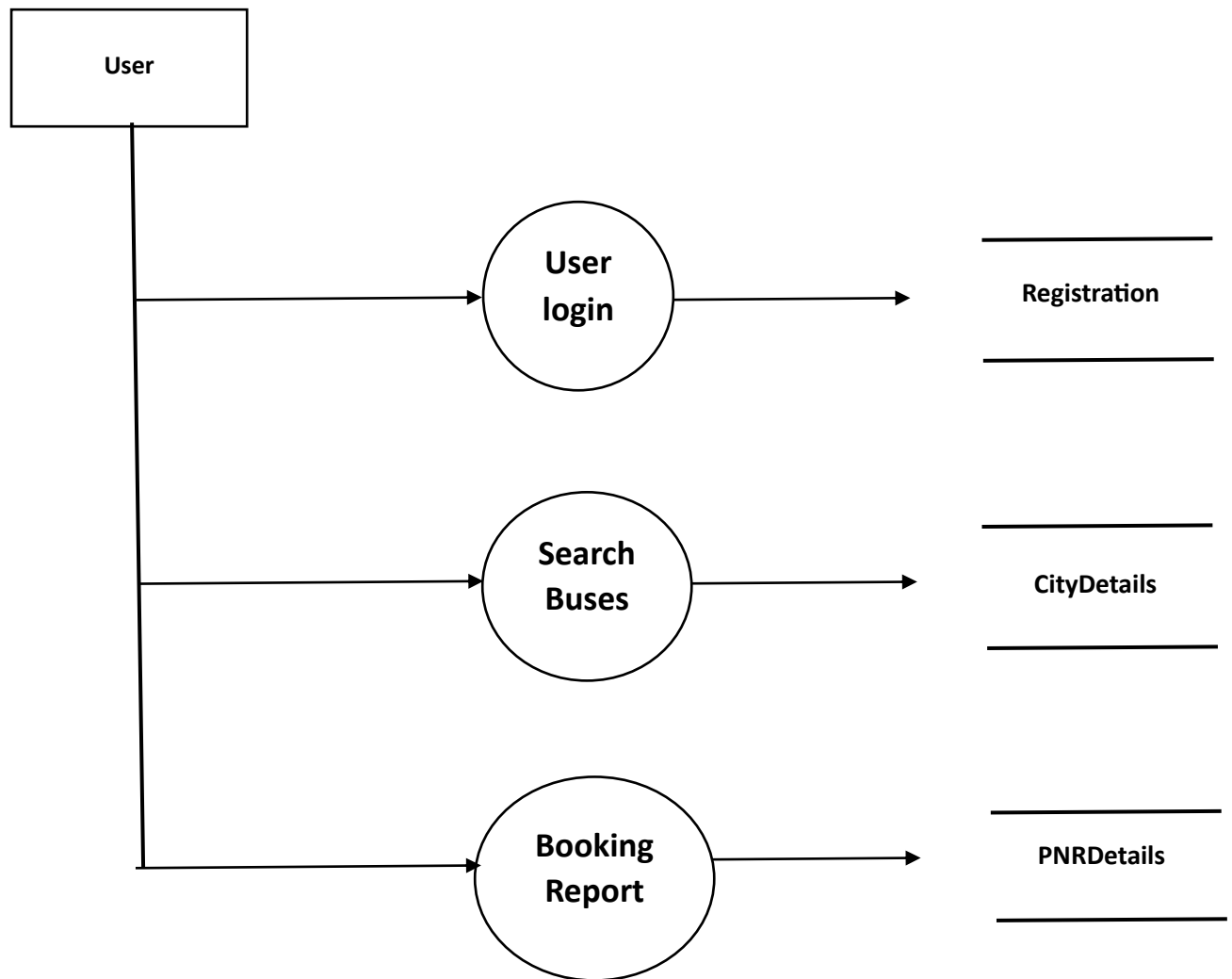
Database of data store

Users:

0 level data flow diagram

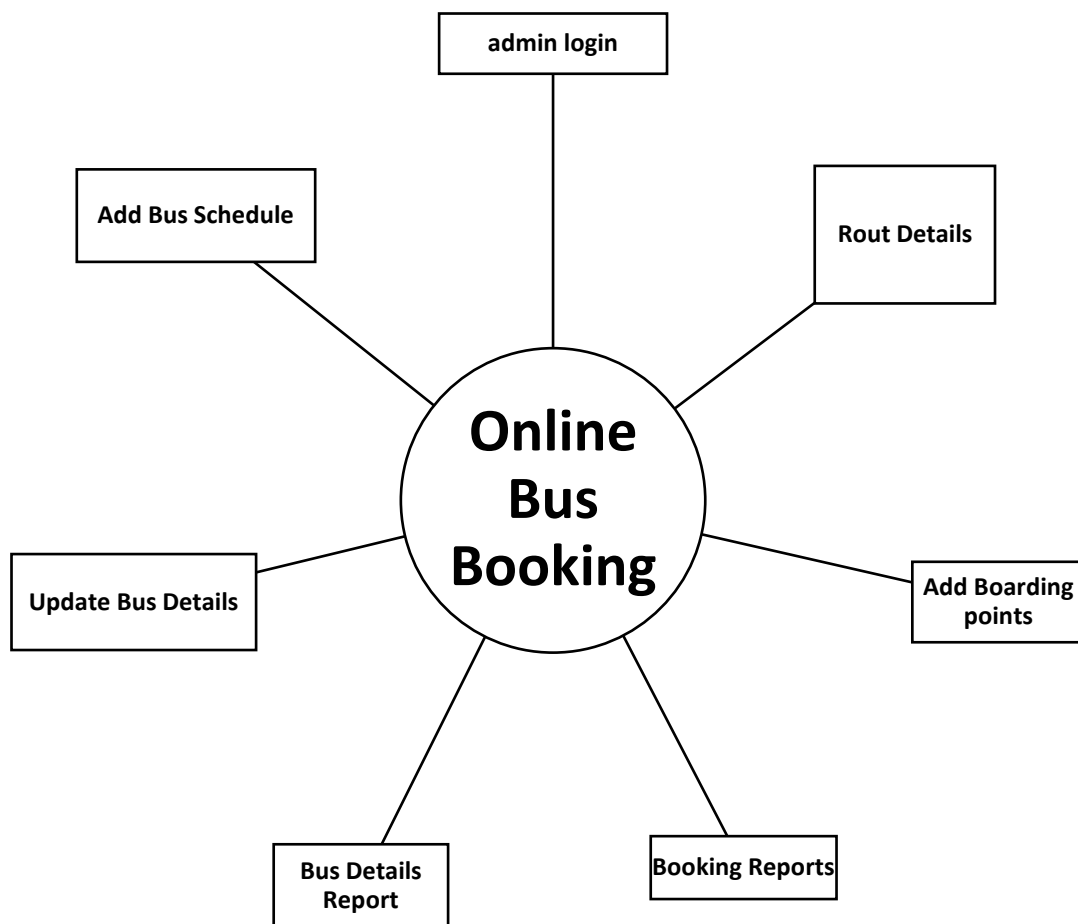


1st level data flow diagram

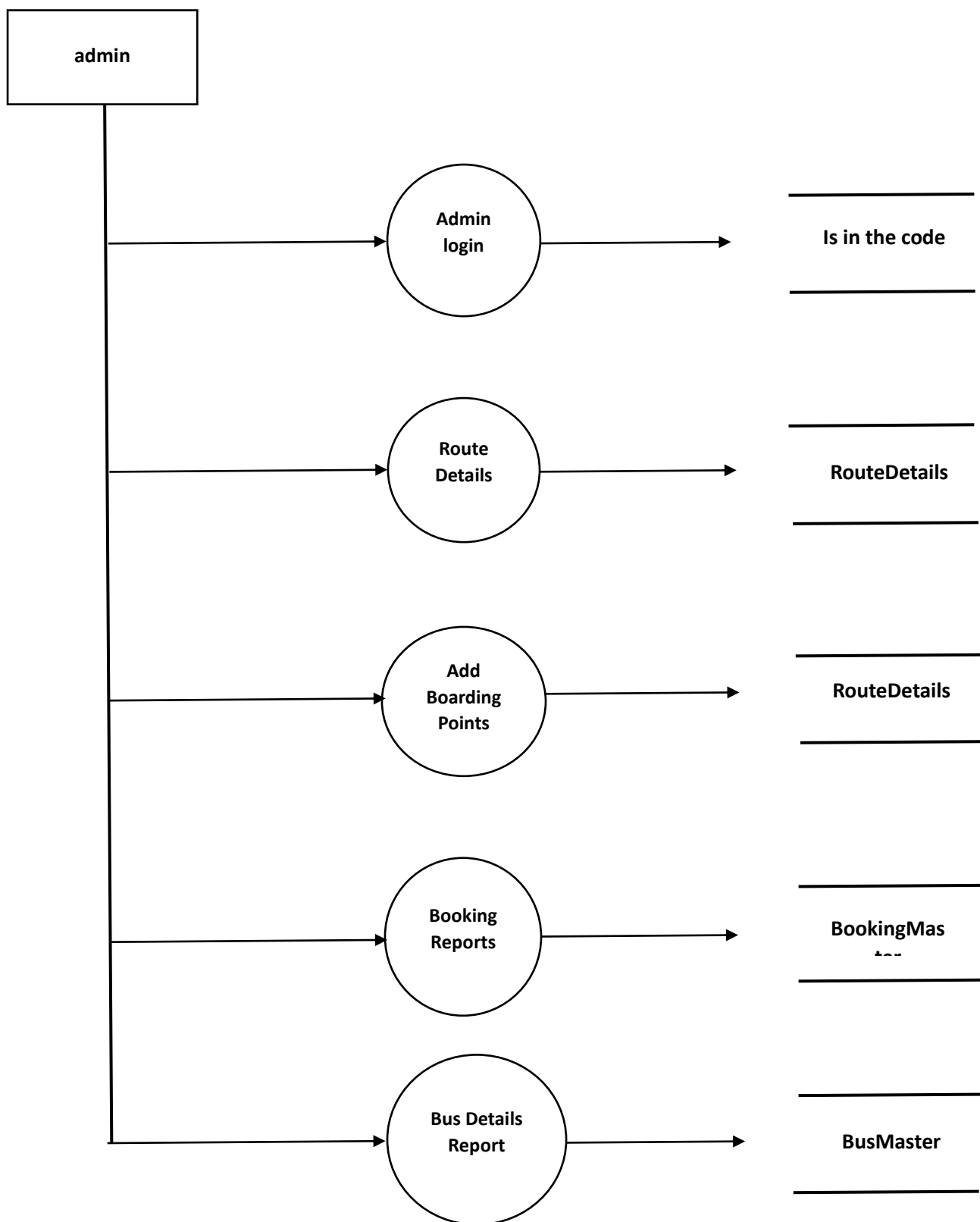


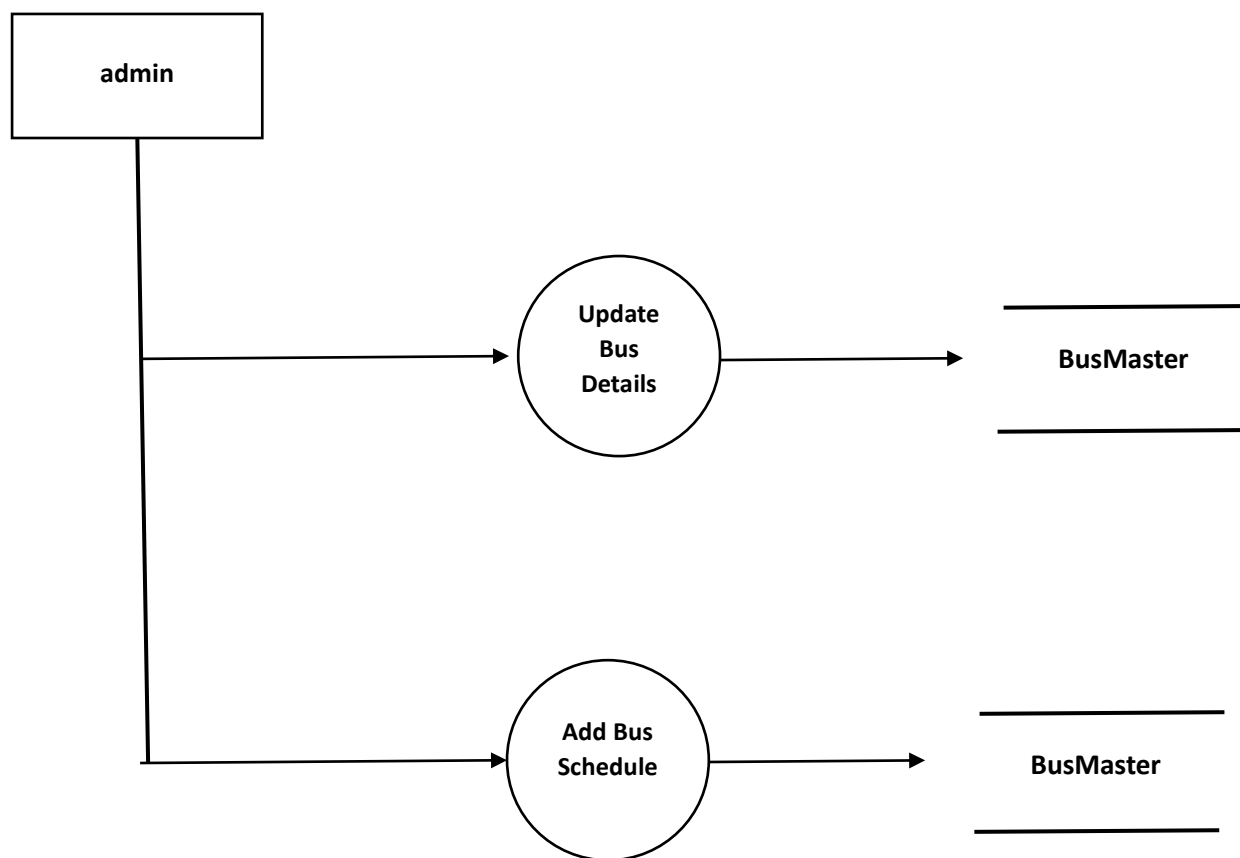
Admin:

0 level data flow diagram

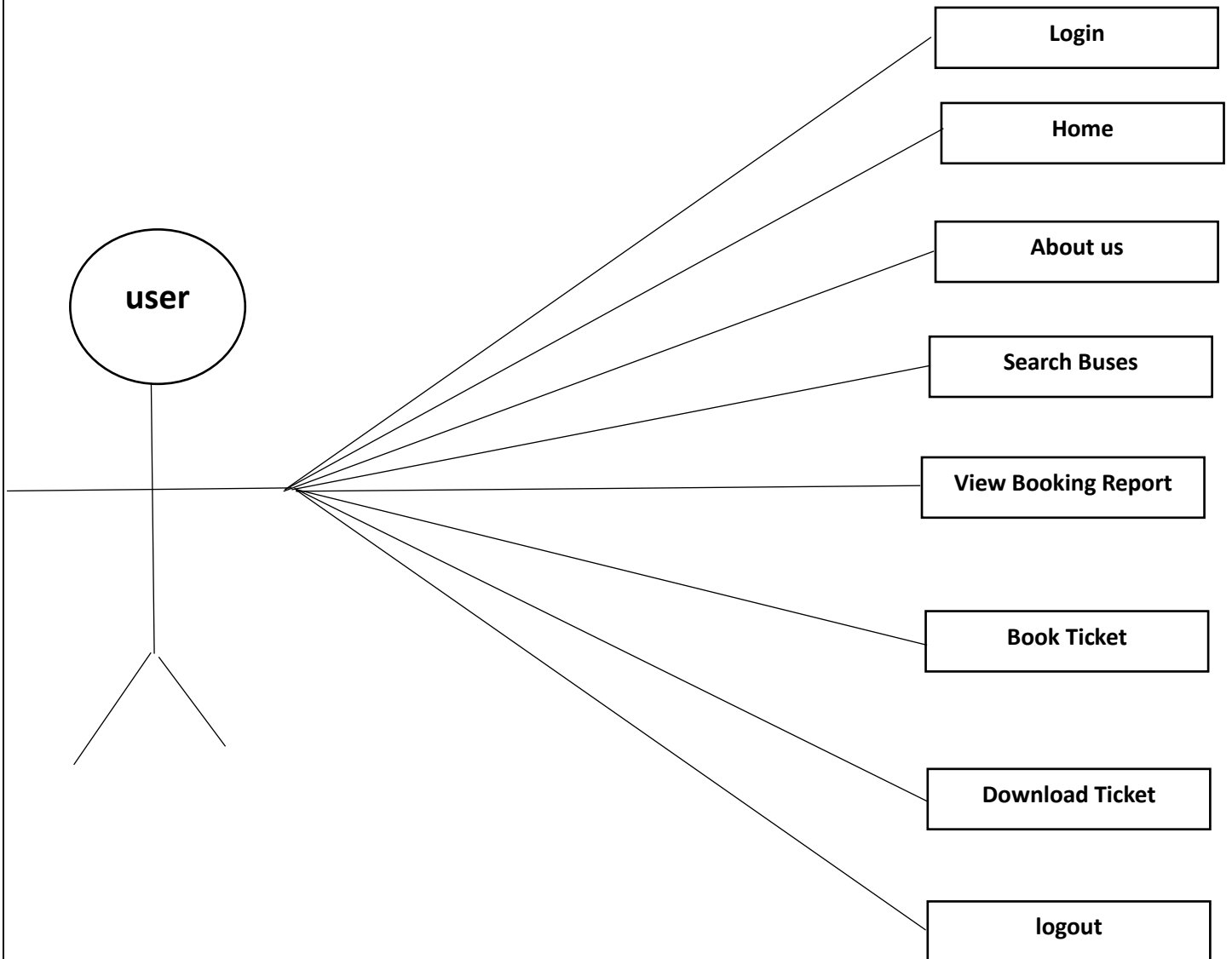


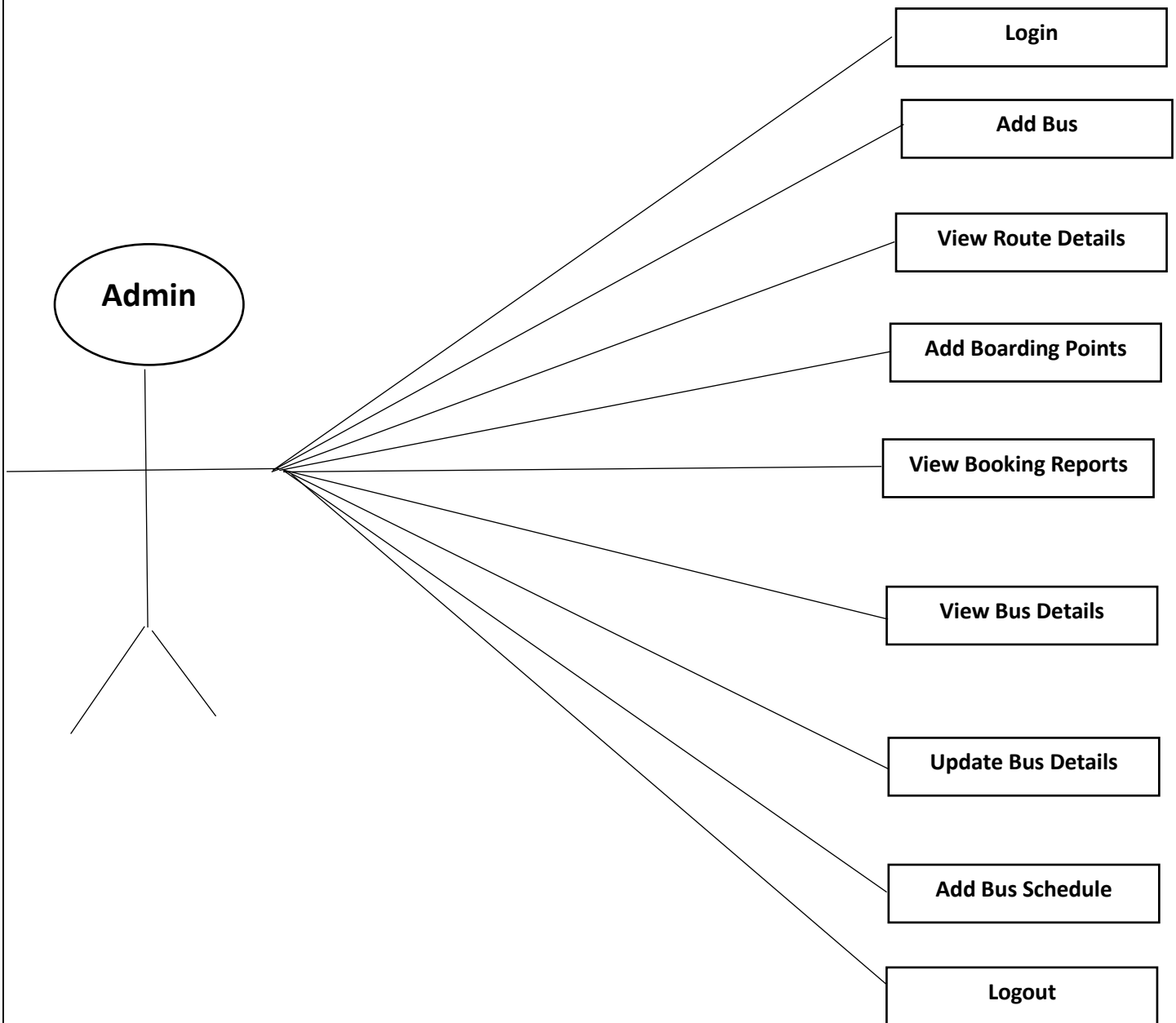
1st level data flow diagram





Use case diagram





Data Structure

BookingMaster:

Name	Type	Size	Constraints
BookingId	Int	-	Primary-key Auto-increment
RegId	Int	-	Not-null
BusId	Int	-	Not-null
Fname	Varchar	50	Not-null
Lname	Varchar	50	Not-null
Email	Varchar	50	Not-null
Contact	Varchar	50	Not-null
City	Varchar	50	Not-null
SeatNo	Varchar	50	Not-null
TravelDate	Varchar	50	Not-null
Origin	Varchar	50	Not-null
Destination	Varchar	50	Not-null
BoardingID	Int	-	Not-null
Fare	Decimal	18,2	Not-null
PNRNo	Varchar	50	Not-null
ScheduleID	Int	-	Not-null

BusMaster:

Name	Type	Size	Constraints
BusId	Int	-	Primary-key Auto-increment
BusNo	Varchar	50	Not-null
BusType	Varchar	50	Not-null
TotalSeat	Int	-	Not-null
SeatColumn	Int	-	Not-null
SeatRow	Int	-	Not-null
BookedSeat	Int	-	Not-null
AvailableSeats	Int	-	Not-null
BusName	Varchar	50	Not-null

CardDetails:

Name	Type	Size	Constraints
CardID	Bigint	-	Primary-key Auto-increment
UserID	Int	-	Not-null
CardType	Varchar	20	Not-null
BankName	Varchar	50	Not-null
CVVNo	Varchar	10	Not-null
CardNo	Nvarchar	100	Not-null
TotalAmount	Decimal	18,2	Not-null
CreatedBy	Int	-	Not-null

CityDetails:

Name	Type	Size	Constraints
CityID	Bigint	-	Primary-key Auto-increment
CityName	varchar	50	Not-null

PickUpStand:

Name	Type	Size	Constraints
StandId	Int	-	Primary-key Auto-increment
Routeld	Int	-	Not-null
PlaceName	Varchar	50	Not-null
PlaceTime	Varchar	50	Not-null
BusID	Int	-	Not-null

PNRDetails:

Name	Type	Size	Constraints
PNRDetailsID	Bigint	-	Primary-key Auto-increment
PNRNo	Varchar	50	Not-null
TotalAmount	Decimal	18,2	Not-null
TotalTickets	Int	-	Not-null
CreatedBy	Int	-	Not-null

Registration:

Name	Type	Size	Constraints
regId	Int	-	Primary-key Auto-increment
Fname	Varchar	50	Not-null
Lname	Varchar	50	Not-null
EmailId	Varchar	50	Not-null
Address	Varchar	50	Not-null
City	Varchar	50	Not-null
PinCode	Varchar	50	Not-null
Contact	Varchar	50	Not-null
Password	Varchar	50	Not-null
CreatedDate	Datetime	-	Not-null

RouteDetails:

Name	Type	Size	Constraints
RouteID	Int	-	Primary-key Auto-increment
Origin	Varchar	50	Not-null
Destination	Varchar	50	Not-null
CreatedDate	Datetime	-	Not-null
BusID	Int	-	Not-null

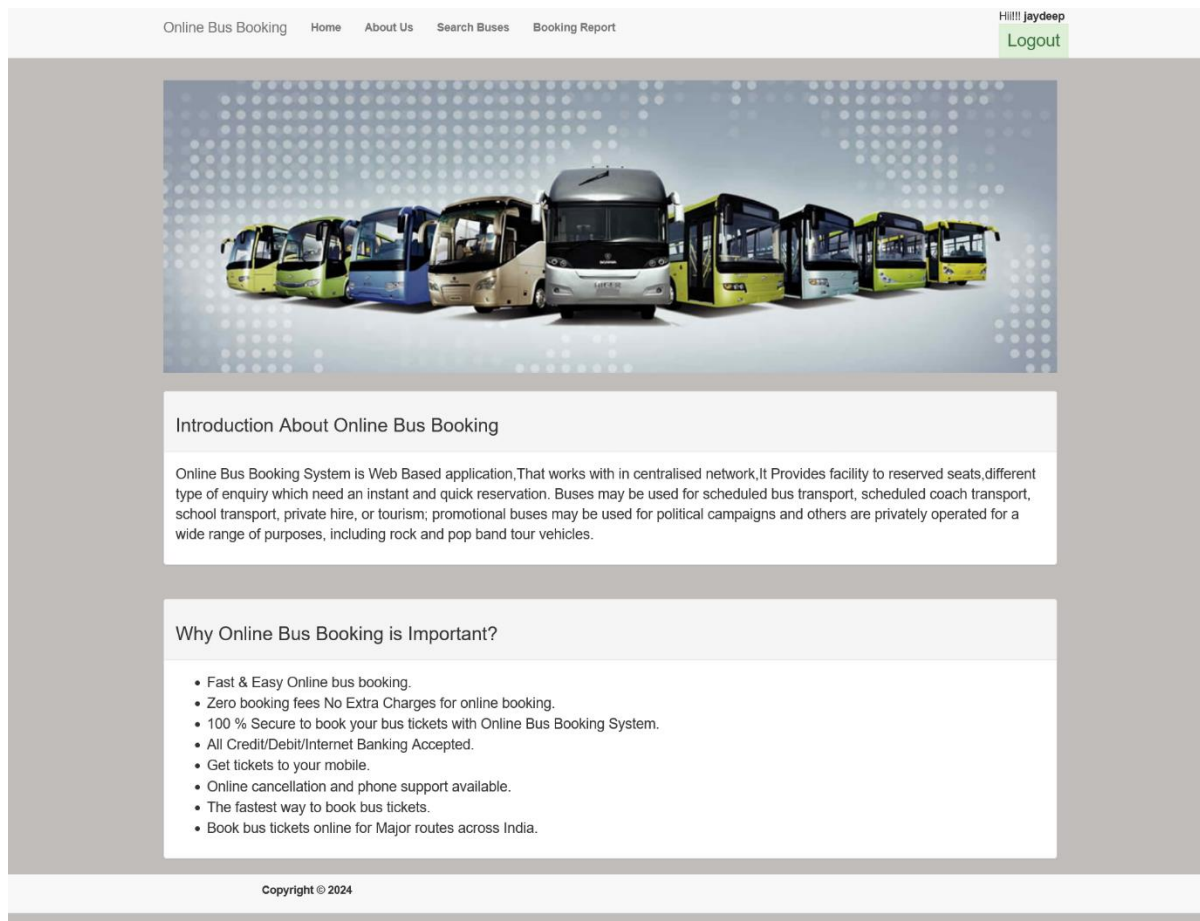
ScheduleMaster:

Name	Type	Size	Constraints
BusId	Int	-	Primary-key Auto-increment
Date	Varchar	50	Not-null
Fare	Decimal	18,2	Not-null
EstimetedTime	Varchar	50	Not-null
ArivalTime	Varchar	50	Not-null
DepartureTime	Varchar	50	Not-null
RouteID	Int	-	Not-null
BookedSeats	Int	-	Not-null
AvailableSeats	Int	-	Not-null

Screen layout and body

User

Default.aspx



```
<%@ Page Title="" Language="C#" MasterPageFile="~/BusBookingMaster.Master" AutoEventWireup="true"
CodeBehind="Default.aspx.cs" Inherits="BusBookingProject.Default" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
```

```
</asp:Content>
```

```
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
```

```
<div class="container" style="margin-top:6%">
```

```
<div class="row">
```

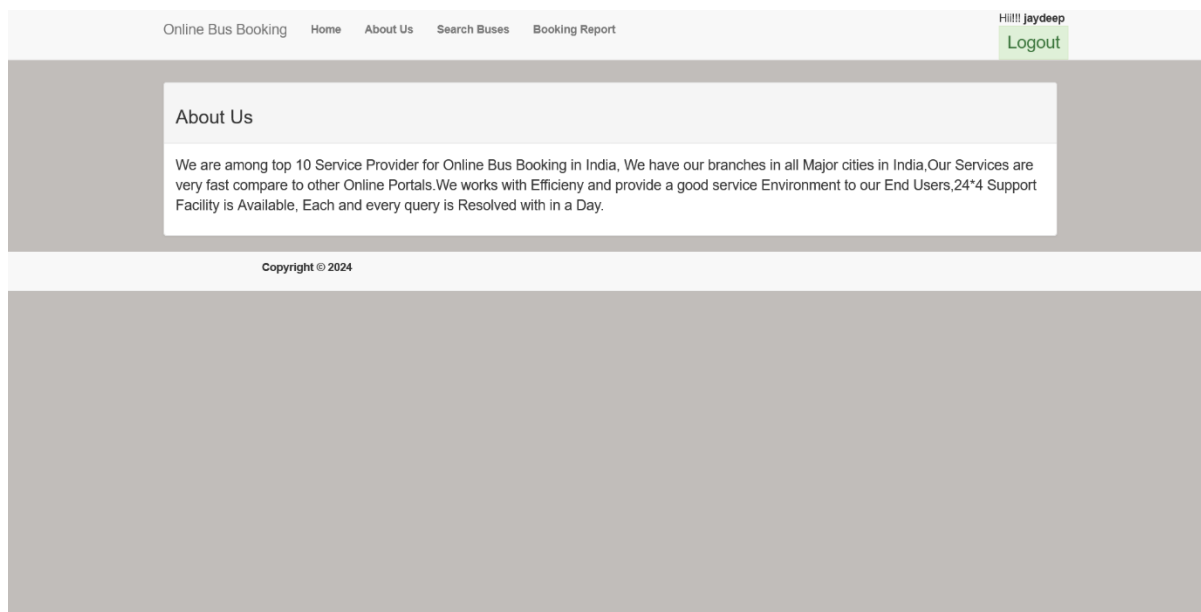
```

<div class="col-lg-12">
<asp:Image ID="imgBus" ImageAlign="AbsMiddle" ImageUrl="~/img/banner-home-1.jpg" style="width:100%"
runat="server" />
</div>
<div class="col-lg-12" style="margin-top:2%">
<div class="panel panel-default">
<div class="panel-heading">
<div class=" panel-title">
<h3>Introduction About Online Bus Booking</h3>
</div></div>
<div class="panel-body">
<p style="font-size:large">
Online Bus Booking System is Web Based application,That works with in centralised network,It Provides facility to
reserved seats,different type of enquiry which need an instant and quick reservation.

Buses may be used for scheduled bus transport, scheduled coach transport, school transport, private hire,
or tourism; promotional buses may be used for political campaigns and others are privately operated for a wide range
of purposes,
including rock and pop band tour vehicles.
</p></div></div></div>
<div class="col-lg-12" style="margin-top:2%">
<div class="panel panel-default">
<div class="panel-heading">
<div class=" panel-title">
<h3>Why Online Bus Booking is Important?</h3></div></div>
<div class="panel-body" style="font-size:large"><ul>
<li>Fast & Easy Online bus booking.</li><li>Zero booking fees No Extra Charges for online booking.</li><li>
100 % Secure to book your bus tickets with Online Bus Booking System.</li><li>All Credit/Debit/Internet Banking
Accepted.</li><li>Get tickets to your mobile.</li><li>
Online cancellation and phone support available.</li><li>
The fastest way to book bus tickets.</li><li>
Book bus tickets online for Major routes across India.</li></ul>
</div></div></div></div></div>
</asp:Content>

```

AboutUs.aspx



```
<%@ Page Title="" Language="C#" MasterPageFile="~/BusBookingMaster.Master"
AutoEventWireup="true" CodeBehind="AboutUs.aspx.cs" Inherits="BusBookingProject.AboutUs" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
<div class="container" style="margin-top:6%">
<div class="panel panel-default">
<div class="panel-heading">
<div class="panel-title">
<h3>About Us</h3>
</div></div>
<div class="panel-body">
<p style="font-size:large">
We are among top 10 Service Provider for Online Bus Booking in India,
We have our branches in all Major cities in India,Our Services are very fast
compare to other Online Portals.We works with Efficiency and provide a
good service Environment to our End Users,24*4 Support Facility is Available,
Each and every query is Resolved with in a Day.
</p></div></div></div>
</asp:Content>
```

Search Buses (Home.aspx)

```
<%@ Page Title="" Language="C#" MasterPageFile="~/BusBookingMaster.Master"
AutoEventWireup="true" CodeBehind="Home.aspx.cs" Inherits="BusBookingProject.Home" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
<div class="container" style="margin-top:8%">
<div class="row ">
<div class="col-lg-4 col-sm-4 col-md-2 col-sm-offset-2 col-md-offset-2" style="margin-left:30%">
<div class="panel panel-default">
<div class="panel-heading">
<h3 class="panel-title">Search For Available Buses</h3>
</div>
<div class="panel-body">
<div id="divMessage" runat="server" />
<div class="col-xs-12 col-sm-12 col-md-12">
```

```

<div class="form-group">
<asp:Label ID="lblAccType" runat="server" Text="From" Font-Bold="true"></asp:Label>
<asp:DropDownList ID="ddlOrigin" class="form-control input-sm floatlabel" runat="server">
</asp:DropDownList>
</div></div>

<div class="col-xs-12 col-sm-12 col-md-12">
<div class="form-group">
<asp:Label ID="Label1" runat="server" Text="To" Font-Bold="true"></asp:Label>
<asp:DropDownList ID="ddlDestination" class="form-control input-sm floatlabel" runat="server">
</asp:DropDownList>
</div></div>

<div class="col-xs-12 col-sm-12 col-md-12">
<div class="form-group">
<asp:Label ID="lblDate" runat="server" Text="Travel Date" Font-Bold="true"></asp:Label>
<asp:TextBox ID="txtDate" runat="server" placeholder="DD/MM/YYYY Format" class="form-control
input-sm floatlabel"></asp:TextBox>
</div></div>

<div class="col-xs-12 col-sm-12 col-md-12">
<div class="form-group">
<asp:Button ID="btnSearch" runat="server" Text="Search Buses" class="btn btn-info btn-block"
OnClick="btnSearch_Click"/>
</div></div></div></div></div></div></div>
</asp:Content>

```

Search Buses (Home.aspx.cs)

```

using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
using System.Linq;

```



```

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace BusBookingProject{

public partial class Home : System.Web.UI.Page{

#region Global Variable

SqlConnection connString = new
SqlConnection(ConfigurationManager.ConnectionStrings["OnlineBusBookingConnectionString"].ToString());

#endregion

protected void Page_Load(object sender, EventArgs e){

if(!IsPostBack){

bindOriginCity();

bindDextinationCity();}}

private void bindOriginCity(){

DataSet dsOrigin = getCity();

if(dsOrigin.Tables[0].Rows.Count>0){

ddlOrigin.DataSource = dsOrigin.Tables[0];

ddlOrigin.DataTextField = "CityName";

ddlOrigin.DataValueField = "CityName";

ddlOrigin.DataBind();}

ddlOrigin.Items.Insert(0, new ListItem("-Select City--","0"));}

private void bindDextinationCity(){

DataSet dsOrigin = getCity();

if (dsOrigin.Tables[0].Rows.Count > 0){

ddlDestination.DataSource = dsOrigin.Tables[0];

ddlDestination.DataTextField = "CityName";

ddlDestination.DataValueField = "CityName";

ddlDestination.DataBind();}

ddlDestination.Items.Insert(0, new ListItem("-Select City--", "0"));}

private DataSet getCity(){

```

```
DataSet dsGetData = new DataSet();

SqlCommand sqlCmd = new SqlCommand();

if (connString.State == ConnectionState.Closed){
    connString.Open();}

sqlCmd.CommandType = CommandType.StoredProcedure;

sqlCmd.CommandText = "ispGetCity";

sqlCmd.Connection = connString;

SqlDataAdapter sda = new SqlDataAdapter(sqlCmd);

sda.Fill(dsGetData);

return dsGetData;}

protected void btnSearch_Click(object sender, EventArgs e){

Response.Redirect("BusBookingSearchDetails.aspx?Origin="+ddlOrigin.SelectedItem.Text+"&Destination="+ddlDestination.SelectedItem.Text+"&TravelDate="+txtDate.Text);

}}}
```

BookingReport.aspx

Online Bus Booking

Home

About Us

Search Buses

Booking Report

Hi!!!! jaydeep

Logout

Sr.No	PNR No	Total Booked Tickets	Booking Amount	Action
1	Q4W49K	1	150.00	Download Ticket

Copyright © 2024

```

<%@ Page Title="" Language="C#" MasterPageFile="~/BusBookingMaster.Master"
AutoEventWireup="true" CodeBehind="BookingReport.aspx.cs"
Inherits="BusBookingProject.BookingReport" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="container" style="margin-top:6%">

<asp:GridView ID="gdTicketReport" runat="server" EmptyDataText="No Record Found...."
AutoGenerateColumns="False" AllowPaging="true" PageSize="20" CssClass=""

Width="100%" Font-Size="12" OnRowCommand="gdTicketReport_RowCommand">

<Columns>

<asp:TemplateField HeaderText="Sr.No">

<ItemTemplate>

<%# Container.DataItemIndex+1 %>

</ItemTemplate>

</asp:TemplateField>

<asp:BoundField HeaderText="PNR No" DataField="PNRNo" />

<asp:BoundField HeaderText="Total Booked Tickets" DataField="TotalTickets" />

<asp:BoundField HeaderText="Booking Amount" DataField="TotalAmount" />

<asp:TemplateField HeaderText="Action">

<ItemTemplate>

<asp:LinkButton ID="lnkbtndownloadTicket" runat="server" CommandName="Download Ticket"
CommandArgument='<%# Container.DataItemIndex %>'>Download Ticket</asp:LinkButton>

<asp:HiddenField ID="hdnPNRNo" runat="server" Value='<%# Eval("PNRNo") %>' />

</ItemTemplate>

</asp:TemplateField></Columns>

</asp:GridView>

<div id="ticket" runat="server" visible="false">

<h1 style="font-weight: bold">Online Bus Booking</h1><br /><br />

<h2 style="font-weight: bold">PNR Details</h2>

<div class="table-bordered">

<table class="table-striped" id="tbtPNR" runat="server" style="width: 100%; color: green; font-size:
large"><tr>

```

```

<td style="font-weight: bold">PNR No</td><td>
<asp:Label ID="lblTransactionNo" runat="server"></asp:Label></td></tr><tr>
<td style="font-weight: bold">Total Tickets</td><td>
<asp:Label ID="lblTotalTickets" runat="server"></asp:Label></td></tr><tr>
<td style="font-weight: bold">Total Amount</td><td>
<asp:Label ID="lblTotalAmount" runat="server"></asp:Label></td></tr><tr>
<td style="font-weight: bold">Bus Name</td><td>
<asp:Label ID="lblBusName" runat="server"></asp:Label></td></tr>
</table></div><br /><br />
<h2 style="font-weight: bold">Passenger's Details</h2>
<asp:GridView ID="gdPaxDetails" runat="server" EmptyDataText="No Record Found...."
AutoGenerateColumns="False" AllowPaging="true" PageSize="25" CssClass="table table-hover table-
bordered" Style="margin-top: 5%" Width="100%"><Columns>
<asp:TemplateField HeaderText="Sr.No">
<ItemTemplate>
<%# Container.DataItemIndex+1 %>
</ItemTemplate>
</asp:TemplateField>
<asp:BoundField HeaderText="First Name" DataField="Fname" />
<asp:BoundField HeaderText="Last Name" DataField="Lname" />
<asp:BoundField HeaderText="Contact" DataField="Contact" />
<asp:BoundField HeaderText="SeatNo" DataField="SeatNo" />
<asp:BoundField HeaderText="Travel Date" DataField="TravelDate" />
<asp:BoundField HeaderText="From" DataField="Origin" />
<asp:BoundField HeaderText="To" DataField="Destination" />
<asp:BoundField HeaderText="Boarding Place" DataField="PlaceName" />
</Columns>
</asp:GridView><br /><br />
<p style="text-align: center; font-style: italic; font-size: 10pt">
Thank you for booking Ticket with us,have a pleasant journey!!!!
</p></div></div>
</asp:Content>

```

BookingReport.aspx.cs

```

using iTextSharp.text;
using iTextSharp.text.pdf;
using iTextSharp.tool.xml;
using System;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
using System.IO;
using System.Reflection;
using System.Text;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace BusBookingProject{
public partial class BookingReport : System.Web.UI.Page{

SqlConnection connString = new
SqlConnection(ConfigurationManager.ConnectionStrings["OnlineBusBookingConnectionString"].ToString());

protected void Page_Load(object sender, EventArgs e){
if(!IsPostBack){
if(Session["UserID"]!=null){
bindPnrDetails();

if (Request.QueryString["Msg"] != null){

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "alertMessage", "alert("'" +
Convert.ToString(Request.QueryString["Msg"]) + "'", true);

PropertyInfo isreadonly =
typeof(System.Collections.Specialized.NameValueCollection).GetProperty(
"IsReadOnly", BindingFlags.Instance | BindingFlags.NonPublic);
isreadonly.SetValue(this.Request.QueryString, false, null);

```

```
this.Request.QueryString.Clear();
}}else{
Response.Redirect("Login.aspx");}}}

private void bindPnrDetails(){
DataSet dsGetData = new DataSet();
SqlCommand sqlCmd = new SqlCommand();
if (connString.State == ConnectionState.Closed){
connString.Open();}

sqlCmd.CommandType = CommandType.StoredProcedure;
sqlCmd.Parameters.AddWithValue("@UserID", Convert.ToInt32(Session["UserID"]));
sqlCmd.CommandText = "ispGetPNRDetails";
sqlCmd.Connection = connString;
SqlDataAdapter sda = new SqlDataAdapter(sqlCmd);
sda.Fill(dsGetData);
if (dsGetData.Tables != null){
gdTicketReport.DataSource = dsGetData.Tables[0];
gdTicketReport.DataBind();}else{
gdTicketReport.DataSource = null;
gdTicketReport.EmptyDataText = "No Records Found";
gdTicketReport.DataBind();}}

public override void VerifyRenderingInServerForm(Control control){ }

private void printTicket(string transactionNo){
ticket.Visible = true;

DataSet dsGetData = new DataSet();
SqlCommand sqlCmd = new SqlCommand();
if (connString.State == ConnectionState.Closed){
connString.Open();}

sqlCmd.CommandType = CommandType.StoredProcedure;
sqlCmd.Parameters.AddWithValue("@PNRNo", transactionNo);
sqlCmd.CommandText = "GetPassengerDetails";
sqlCmd.Connection = connString;
```

```

SqlDataAdapter sda = new SqlDataAdapter(sqlCmd);
sda.Fill(dsGetData);
gdPaxDetails.DataSource = dsGetData.Tables[0];
gdPaxDetails.DataBind();
lblTransactionNo.Text = Convert.ToString(dsGetData.Tables[1].Rows[0]["PNRNo"]);
lblBusName.Text = Convert.ToString(dsGetData.Tables[1].Rows[0]["BusName"]);
lblTotalAmount.Text = Convert.ToString(dsGetData.Tables[1].Rows[0]["Amount"]);
lblTotalTickets.Text = Convert.ToString(dsGetData.Tables[1].Rows[0]["TotalTickets"]);
using (StringWriter sw = new StringWriter()){
using (HtmlTextWriter hw = new HtmlTextWriter(sw)){
StringBuilder sb = new StringBuilder();
ticket.RenderControl(hw);
StringReader sr = new StringReader(sw.ToString());
Document pdfDoc = new Document(PageSize.A4, 10f, 10f, 10f, 0);
PdfWriter writer = PdfWriter.GetInstance(pdfDoc, Response.OutputStream);
pdfDoc.Open();
pdfDoc.NewPage();
XMLWorkerHelper.GetInstance().ParseXHtml(writer, pdfDoc, sr);
pdfDoc.Close();
Response.ContentType = "application/pdf";
Response.AddHeader("content-disposition", "attachment;filename=Ticket" + ".pdf");
Response.Cache.SetCacheability(HttpCacheability.NoCache);
Response.Write(pdfDoc);
Response.End();
}}}

protected void gdTicketReport_RowCommand(object sender, GridViewCommandEventArgs e){
if (e.CommandName == "Download Ticket"){
int index = Convert.ToInt32(e.CommandArgument);
GridViewRow row = gdTicketReport.Rows[index];
printTicket(row.Cells[1].Text);
}}}}

```

Admin

(Add Bus) BusDetails.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Admin/Admin.Master"
AutoEventWireup="true" CodeBehind="BusDetails.aspx.cs"
Inherits="BusBookingProject.Admin.BusDetails" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="container" style="margin-top: 5%"></div>

<div class="row centered-form" style="margin-top: 5%">

<div class="col-lg-8 col-sm-8 col-md-2 col-sm-offset-2 col-md-offset-2">

<div class="panel panel-default">

<div class="panel-heading">

<h3 class="panel-title">Add Bus Details</h3></div>

<div class="panel-body">

<asp:ValidationSummary ID="vsRegister" runat="server" CssClass="alert alert-danger"
ShowSummary="true" ValidationGroup="vgRegister" />

<div id="divMessage" runat="server" />

<div class="col-xs-6 col-sm-6 col-md-6">

<div class="form-group">

<asp:Label ID="lblBus" runat="server" Text="Bus No" Font-Bold="true"></asp:Label>
```



```

<asp:TextBox ID="txtBusNo" runat="server" class="form-control input-sm floatlabel"/>

<asp:RequiredFieldValidator runat="server" ControlToValidate="txtBusNo" Display="None"
ID="rfvFirstName" ValidationGroup="vgRegister"

CssClass="text-danger" ErrorMessage="Bus No is required." /><br /></div>

<div class="form-group">

<asp:Label ID="lblSeatRow" runat="server" Text="No Of Seats Rows" Font-Bold="true"></asp:Label>

<asp:TextBox ID="txtSeatRows" runat="server" class="form-control input-sm floatlabel" />

<asp:RequiredFieldValidator runat="server" ControlToValidate="txtSeatRows" Display="None"
ID="rfVMobileNo" ValidationGroup="vgRegister"

CssClass="text-danger" ErrorMessage="Seats Row is required." /><br /></div>

<div class="form-group">

<asp:Label ID="Label1" runat="server" Text="Origin" Font-Bold="true"></asp:Label>

<asp:TextBox ID="txtOrigin" runat="server" class="form-control input-sm floatlabel" />

<asp:RequiredFieldValidator runat="server" ControlToValidate="txtOrigin" Display="None"
ID="RequiredFieldValidator1" ValidationGroup="vgRegister"

CssClass="text-danger" ErrorMessage="Origin is Required" /><br /></div>

<div class="form-group">

<asp:Label ID="Label3" runat="server" Text="BusName" Font-Bold="true"></asp:Label>

<asp:TextBox ID="txtBusName" runat="server" class="form-control input-sm floatlabel" />

<asp:RequiredFieldValidator runat="server" ControlToValidate="txtBusName" Display="None"
ID="RequiredFieldValidator3" ValidationGroup="vgRegister"

CssClass="text-danger" ErrorMessage="Bus Name is Required" /><br /></div></div>

<div class="col-xs-6 col-sm-6 col-md-6">

<div class="form-group">

<asp:Label ID="lblLastName" runat="server" Text="Bus Type" Font-Bold="true"></asp:Label>

<asp:DropDownList ID="ddlBusType" runat="server" class="form-control input-sm floatlabel">

<asp:ListItem Value="0" Text="Select Bus Type"></asp:ListItem>

<asp:ListItem Value="1" Text="Normal"></asp:ListItem>

<asp:ListItem Value="2" Text="AC"></asp:ListItem>

</asp:DropDownList>

<asp:RequiredFieldValidator runat="server" ControlToValidate="ddlBusType" Display="None"
ID="rfVLastName" ValidationGroup="vgRegister"

CssClass="text-danger" ErrorMessage="Last Name is required." /><br /></div>

```

```

<div class="form-group">

<asp:Label ID="lblPassword" runat="server" Text="Total Seats Column" Font-
Bold="true"></asp:Label>

<asp:TextBox ID="txtSeatColumn" runat="server" class="form-control input-sm floatlabel" />

<asp:RequiredFieldValidator runat="server" ControlToValidate="txtSeatColumn" Display="None"
ID="rfvPassword" ValidationGroup="vgRegister"

CssClass="text-danger" ErrorMessage="Seats Column is Require" /><br /></div>

<div class="form-group">

<asp:Label ID="Label2" runat="server" Text="Destination" Font-Bold="true"></asp:Label>

<asp:TextBox ID="txtDetination" runat="server" class="form-control input-sm floatlabel" />

<asp:RequiredFieldValidator runat="server" ControlToValidate="txtDetination" Display="None"
ID="RequiredFieldValidator2" ValidationGroup="vgRegister"

CssClass="text-danger" ErrorMessage="Destination is Required" /><br /></div></div>

<div class="col-xs-6 col-sm-6 col-md-6">

<div class="form-group">

<asp:Button ID="btnSave" runat="server" style="width:auto;margin-top:20px;" Text="Add Bus
Details" OnClick="btnSave_Click" class="btn btn-info " ValidationGroup="vgRegister"
CausesValidation="True" ViewStateMode="Inherit" />

</div></div></div></div></div>

</asp:Content>

```

(Add Bus) BusDetails.aspx.cs

```

using System;

using System.Collections.Generic;

using System.Configuration;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace BusBookingProject.Admin{

```

```

public partial class BusDetails : System.Web.UI.Page{

#region Global Variable

SqlConnection connString = new
SqlConnection(ConfigurationManager.ConnectionStrings["OnlineBusBookingConnectionString"].ToString());

#endregion

protected void Page_Load(object sender, EventArgs e){

if(!IsPostBack){

if(Session["UserName"]!=null){

if(Request.QueryString["BusID"]!=null){

int BusID = Convert.ToInt32(Request.QueryString["BusID"]);

FillData(BusID);

btnSave.Text = "Update";}

else{

btnSave.Text="Insert";}}

else{

Response.Redirect("AdminLogin.aspx");}}}

private int UpdateData(){

int ResultCout = 0;

SqlCommand sqlCmd = new SqlCommand();

if (connString.State == ConnectionState.Closed){

connString.Open();}

sqlCmd.CommandType = CommandType.StoredProcedure;

sqlCmd.Parameters.AddWithValue("@BusID", Convert.ToInt32(Request.QueryString["BusID"]));

sqlCmd.Parameters.AddWithValue("@BusNo", Convert.ToString(txtBusNo.Text));

sqlCmd.Parameters.AddWithValue("@BusName", Convert.ToString(txtBusName.Text));

sqlCmd.Parameters.AddWithValue("@BusType", Convert.ToString(ddlBusType.SelectedItem.Text));

sqlCmd.Parameters.AddWithValue("@seatColumn", Convert.ToInt32(txtSeatColumn.Text));

sqlCmd.Parameters.AddWithValue("@SeatRow", Convert.ToInt32(txtSeatRows.Text));

sqlCmd.Parameters.AddWithValue("@Origin", Convert.ToString(txtOrigin.Text));

sqlCmd.Parameters.AddWithValue("@Destination", Convert.ToString(txtDetination.Text));

sqlCmd.CommandText = "ispUpdateBusData";

```

```

sqlCmd.Connection = connString;

ResultCout = sqlCmd.ExecuteNonQuery(); ;

return ResultCout;}

private void FillData(int BusID){
    DataSet dsGetData = new DataSet();

    SqlCommand sqlCmd = new SqlCommand();

    if (connString.State == ConnectionState.Closed){
        connString.Open();}

    sqlCmd.CommandType = CommandType.StoredProcedure;

    sqlCmd.Parameters.AddWithValue("@BusID", BusID);

    sqlCmd.CommandText = "ispGetBusDataByBusID";

    sqlCmd.Connection = connString;

    SqlDataAdapter sda = new SqlDataAdapter(sqlCmd);

    sda.Fill(dsGetData);

    if(dsGetData.Tables[0].Rows.Count>0){

        txtBusName.Text = Convert.ToString(dsGetData.Tables[0].Rows[0]["BusName"]);

        txtBusNo.Text = Convert.ToString(dsGetData.Tables[0].Rows[0]["BusNo"]);

        ddlBusType.SelectedItem.Text = Convert.ToString(dsGetData.Tables[0].Rows[0]["BustType"]);

        txtSeatRows.Text = Convert.ToString(dsGetData.Tables[0].Rows[0]["SeatRow"]);

        txtSeatColumn.Text = Convert.ToString(dsGetData.Tables[0].Rows[0]["SeatColumn"]);

        txtOrigin.Text = Convert.ToString(dsGetData.Tables[0].Rows[0]["Origin"]);

        txtDetination.Text = Convert.ToString(dsGetData.Tables[0].Rows[0]["Destination"]);}

    private int AddBusDetails(){

        int ResultCout = 0;

        SqlCommand sqlCmd = new SqlCommand();

        if (connString.State == ConnectionState.Closed){

            connString.Open();}

        sqlCmd.CommandType = CommandType.StoredProcedure;

        sqlCmd.Parameters.AddWithValue("@BusNo", Convert.ToString(txtBusNo.Text));

        sqlCmd.Parameters.AddWithValue("@BusName", Convert.ToString(txtBusName.Text));

        sqlCmd.Parameters.AddWithValue("@BustType", Convert.ToString(ddlBusType.SelectedItem.Text));

```

```


sqlCmd.Parameters.AddWithValue("@SeatColumn", Convert.ToInt32(txtSeatColumn.Text));
sqlCmd.Parameters.AddWithValue("@SeatRow", Convert.ToInt32(txtSeatRows.Text));
sqlCmd.Parameters.AddWithValue("@Origin", Convert.ToString(txtOrigin.Text));
sqlCmd.Parameters.AddWithValue("@Destination", Convert.ToString(txtDetination.Text));
sqlCmd.CommandText = "ispAddBusDetails";
sqlCmd.Connection = connString;
ResultCout = sqlCmd.ExecuteNonQuery(); ;
return ResultCout;}

protected void btnSave_Click(object sender, EventArgs e){
if(btnSave.Text=="Insert"){
int result = AddBusDetails();
if (result == -1){
ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "alertMessage", "alert('Bus Details has
been added successfully')", true);

txtDetination.Text = "";
ddlBusType.SelectedValue = "0";
txtOrigin.Text = "";
txtBusNo.Text = "";
txtSeatColumn.Text = "";
txtSeatRows.Text = "";
txtBusName.Text = "";}else{
ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "alertMessage", "alert('Error occur
please contact your system administrator')", true);}}else{
int result = UpdateData();
if (result == -1){
Response.Redirect("BusDetailsReport.aspx");}else{
ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "alertMessage", "alert('Updation Failed
please contact your system administrator')", true);}}}}

```

RouteDetails.aspx

Online Bus Booking  Add Bus Route Details Booking Report Bus Details Report					Home Hello, Admin! Logout	
Sr.No	Route ID	Origin	Destination	Action	<div>Copyright © 2024</div>	
1	3	Mumbai	Pune	Add Boarding Points		
2	4	Nallasopara	Churchgate	Add Boarding Points		
3	5	Mumbai	Surat	Add Boarding Points		
4	6	Vasai	Virar	Add Boarding Points		

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Admin/Admin.Master" AutoEventWireup="true"
CodeBehind="RouteDetails.aspx.cs" Inherits="BusBookingProject.Admin.RouteDetails" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
```

```
</asp:Content>
```

```
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
```

```
<div class="container" style="margin-top: 8%">
```

```
<asp:GridView ID="gdRouteDetails" runat="server" EmptyDataText="No Record Found...." AutoGenerateColumns="False"
AllowPaging="true" PageSize="20" CssClass="table table-hover table-bordered"
```

```
Width="100%" Font-Size="12" OnRowDataBound="gdRouteDetails_RowDataBound"><Columns>
```

```
<asp:TemplateField HeaderText="Sr.No">
```

```
<ItemTemplate>
```

```
<%# Container.DataItemIndex+1 %>
```

```
</ItemTemplate>
```

```
</asp:TemplateField >
```

```
<asp:TemplateField HeaderText="Route ID">
```

```
<ItemTemplate>
```

```
<asp:Label ID="lblBusNo" runat="server" Text='<%# Eval("RouteID") %>'></asp:Label>
```

```
</ItemTemplate>
```

```
</asp:TemplateField>
```

```
<asp:TemplateField HeaderText="Origin">
```

```
<ItemTemplate>
```

```

</asp:TemplateField >
<asp:TemplateField HeaderText="Route ID">
<ItemTemplate>
<asp:Label ID="lblBusNo" runat="server" Text='<%# Eval("RouteID") %>'></asp:Label>
</ItemTemplate>
</asp:TemplateField>
<asp:TemplateField HeaderText="Origin">
<ItemTemplate>
<asp:Label ID="lblBusName" runat="server" Text='<%# Eval("Origin") %>'></asp:Label>
</ItemTemplate>
</asp:TemplateField>
<asp:TemplateField HeaderText="Destination">
<ItemTemplate>
<asp:Label ID="lblBusType" runat="server" Text='<%# Eval("Destination") %>'></asp:Label>
</ItemTemplate>
</asp:TemplateField>
<asp:TemplateField HeaderText="Action">
<ItemTemplate>
<asp:HyperLink ID="hlinkBoarding" runat="server" >Add Boarding Points</asp:HyperLink>
<asp:HiddenField ID="hdnRouteID" runat="server" Value='<%# Eval("RouteID") %>' />
<asp:HiddenField ID="hdnBusID" runat="server" Value='<%# Eval("BusId") %>' />
</ItemTemplate>
</asp:TemplateField></Columns>
</asp:GridView></div>
</asp:Content>

```

RouteDetails.aspx.cs

```
using System;

using System.Collections.Generic;

using System.Configuration;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace BusBookingProject.Admin{

public partial class RouteDetails : System.Web.UI.Page{

SqlConnection connString = new

SqlConnection(ConfigurationManager.ConnectionStrings["OnlineBusBookingConnectionString"].ToString());

protected void Page_Load(object sender, EventArgs e){

if(!IsPostBack){

if(Session["UserName"] !=null){

bindBoardingDetails();}else{

Response.Write("AdminLogin.aspx");}}}

private void bindBoardingDetails(){

DataSet dsGetData = new DataSet();

SqlCommand sqlCmd = new SqlCommand();

if (connString.State == ConnectionState.Closed){

connString.Open();}

sqlCmd.CommandType = CommandType.StoredProcedure;

sqlCmd.CommandText = "ispGetRouteDetails";

sqlCmd.Connection = connString;

SqlDataAdapter sda = new SqlDataAdapter(sqlCmd);

sda.Fill(dsGetData);

if (dsGetData.Tables[0].Rows.Count > 0){

gdRouteDetails.DataSource = dsGetData.Tables[0];

gdRouteDetails.DataBind();}else{
```


Sr.No	Bus Name	Passenger Name	Email ID	Contact No	Origin	Destination	Travel Date	Seat No	Fare	Booked By
1	Neeta Volvo	AnandSingh	abc@gmail.com	8291951419	Mumbai	Pune	25 Mar 2024	34	800.00	AnandSingh
2	Neeta Volvo	AnmoSharma	abc@gmail.com	9029451159	Mumbai	Pune	25 Mar 2024	35	800.00	AnandSingh
3	Neeta Volvo	AtulDubey	atul@gmail.com	98989898	Mumbai	Pune	25 Mar 2024	32	800.00	AtulDubey
4	Abhinav Transporters	AtulDubey	atul@gmail.com	9561454638	Mumbai	Surat	30 Mar 2024	31	800.00	AnandSingh
5	Abhinav Transporters	AmitDubey	a@gmail.com	8888245689	Mumbai	Surat	30 Mar 2024	32	800.00	AnandSingh
6	Abhinav Transporters	SurajDubey	a@gmail.com	8945621547	Mumbai	Surat	30 Mar 2024	33	800.00	AnandSingh
7	Abhinav Transporters	NehaDubey	neha@gmail.com	9029451159	Mumbai	Surat	28 Mar 2024	26	500.00	JitendraDubey
8	Neeta Volvo	AtulDubey	atul@gmail.com	9029451159	Mumbai	Pune	30 Mar 2024	11	250.00	JitendraDubey
9	Neeta Volvo	KiranDubey	kiran@tiwari.com	8888245689	Mumbai	Pune	30 Mar 2024	12	250.00	JitendraDubey
10	Neeta Volvo	jaydeepmakwana	jaydeep@gmail.com	66666666666666666666	Mumbai	Pune	25 Mar 2024	1	150.00	jaydeepmakwana

Copyright © 2024

```

<%@ Page Title="" Language="C#" MasterPageFile="~/Admin/Admin.Master" AutoEventWireup="true"
CodeBehind="BookingReport.aspx.cs" Inherits="BusBookingProject.Admin.BookingReport" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="container" style="margin-top:6%">

<asp:GridView ID="gdTicketReport" runat="server" EmptyDataText="No Record Found...."
AutoGenerateColumns="False" AllowPaging="true" PageSize="20" CssClass=""
Width="100%" Font-Size="12"><Columns>

<asp:TemplateField HeaderText="Sr.No">

<ItemTemplate>

<%# Container.DataItemIndex+1 %>

</ItemTemplate>

</asp:TemplateField>

<asp:BoundField HeaderText="Bus Name" DataField="BusName" />

<asp:BoundField HeaderText="Passenger Name" DataField="PaxName" />

<asp:BoundField HeaderText="Email ID" DataField="Email" />

<asp:BoundField HeaderText="Contact No" DataField="Contact" />

<asp:BoundField HeaderText="Origin" DataField="Origin" />

<asp:BoundField HeaderText="Destination" DataField="Destination" />

<asp:BoundField HeaderText="Travel Date" DataField="TravelDate" />

<asp:BoundField HeaderText="Seat No" DataField="SeatNo" />

<asp:BoundField HeaderText="Fare" DataField="Fare" />

<asp:BoundField HeaderText="Booked By" DataField="BookedBy" /></Columns>

</asp:GridView></div>

</asp:Content>

```

BookingReport.aspx.cs

```
using System;

using System.Collections.Generic;

using System.Configuration;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace BusBookingProject.Admin{

public partial class BookingReport : System.Web.UI.Page{

SqlConnection connString = new

SqlConnection(ConfigurationManager.ConnectionStrings["OnlineBusBookingConnectionString"].ToString());

protected void Page_Load(object sender, EventArgs e){

if(Session["UserName"] !=null){

bingBookingReport();}else{ Response.Redirect("Login.aspx");}}

private void bingBookingReport(){ DataSet dsGetData = new DataSet();

SqlCommand sqlCmd = new SqlCommand();

if(connString.State==ConnectionState.Closed){ connString.Open(); }

sqlCmd.CommandType = CommandType.StoredProcedure;

sqlCmd.CommandText = "ispGetBookingReportByAdmin";

sqlCmd.Connection = connString; SqlDataAdapter sda = new SqlDataAdapter(sqlCmd);

sda.Fill(dsGetData);

if(dsGetData.Tables[0].Rows.Count>0){ gdTicketReport.DataSource = dsGetData.Tables[0];

gdTicketReport.DataBind();}else{

gdTicketReport.DataSource =null;

gdTicketReport.EmptyDataText = "No Records Found";

gdTicketReport.DataBind();}}}
```

BusDetailsReport.aspx

Online Bus Booking Home Add Bus Route Details Booking Report Bus Details Report Home Hello, Admin! Logout									
Sr.No	Bus ID	Bus No	Bus Name	Bus Type	Seat Columns	Seat Row	Origin	Destination	Action
1	3	NV245632	Neeta Volvo	AC	7	5	Mumbai	Pune	Update Details Add Bus Schedule
2	4	64656	Atul Bus	AC	9	6	Nallasopara	Churchgate	Update Details Add Bus Schedule
3	5	A254635	Abhinav Transporters	AC	7	5	Mumbai	Surat	Update Details Add Bus Schedule
4	6	A254635	Vasai-Virar	Normal	9	6	Vasai	Virar	Update Details Add Bus Schedule

Copyright © 2024

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Admin/Admin.Master"
AutoEventWireup="true" CodeBehind="BusDetailsReport.aspx.cs"
Inherits="BusBookingProject.Admin.BusDetailsReport" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="container" style="margin-top:6%">

<asp:GridView ID="gdBusDetails" runat="server" EmptyDataText="No Record Found...."
AutoGenerateColumns="False" AllowPaging="true" PageSize="20" CssClass="table table-hover table-
bordered"

Width="100%" Font-Size="12" OnRowDataBound="gdBusDetails_RowDataBound">

<Columns>

<asp:TemplateField HeaderText="Sr.No"><ItemTemplate>

<%# Container.DataItemIndex+1 %></ItemTemplate>

</asp:TemplateField >

<asp:TemplateField HeaderText="Bus ID"><ItemTemplate>

<asp:Label ID="lblBusID" runat="server" Text='<%# Eval("BusId") %>'></asp:Label></ItemTemplate>

</asp:TemplateField>

<asp:TemplateField HeaderText="Bus No"><ItemTemplate>

<asp:Label ID="lblBusNo" runat="server" Text='<%# Eval("BusNo")
%>'></asp:Label></ItemTemplate>
```

```

</asp:TemplateField>

<asp:TemplateField HeaderText="Bus Name"><ItemTemplate>

<asp:Label ID="lblBusName" runat="server" Text='<%# Eval("BusName")
%>'></asp:Label></ItemTemplate>

</asp:TemplateField>

<asp:TemplateField HeaderText="Bus Type"><ItemTemplate>

<asp:Label ID="lblBusType" runat="server" Text='<%# Eval("BustType")
%>'></asp:Label></ItemTemplate>

</asp:TemplateField>

<asp:TemplateField HeaderText="Seat Columns"><ItemTemplate>

<asp:Label ID="lblSeatCol" runat="server" Text='<%# Eval("SeatColumn")
%>'></asp:Label></ItemTemplate>

</asp:TemplateField>

<asp:TemplateField HeaderText="Seat Row"><ItemTemplate>

<asp:Label ID="lblSeatRow" runat="server" Text='<%# Eval("SeatRow")
%>'></asp:Label></ItemTemplate>

</asp:TemplateField>

<asp:TemplateField HeaderText="Origin"><ItemTemplate>

<asp:Label ID="lblOrigin" runat="server" Text='< %# Eval("Origin") %>'></asp:Label></ItemTemplate>

</asp:TemplateField>

<asp:TemplateField HeaderText="Destination"><ItemTemplate>

<asp:Label ID="lblDestination" runat="server" Text='<%# Eval("Destination")
%>'></asp:Label></ItemTemplate>

</asp:TemplateField>

<asp:TemplateField HeaderText="Action"><ItemTemplate>

<asp:HyperLink ID="hlinkUpdate" runat="server" >Update Details</asp:HyperLink>

<asp:HyperLink ID="hlinkAddSchedule" runat="server" >Add Bus Schedule</asp:HyperLink>

<asp:HiddenField ID="hdnPNRNo" runat="server" Value='<%# Eval("BusId") %>' />

<asp:HiddenField ID="hdnRouteID" runat="server" Value='<%# Eval("RouteID") %>' />

</ItemTemplate></asp:TemplateField></Columns></asp:GridView></div>

</asp:Content>

```

BusDetailsReport.aspx.cs

```
using System;

using System.Collections.Generic;

using System.Configuration;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace BusBookingProject.Admin{

public partial class BusDetailsReport : System.Web.UI.Page{

SqlConnection connString = new

SqlConnection(ConfigurationManager.ConnectionStrings["OnlineBusBookingConnectionString"].ToString());

protected void Page_Load(object sender, EventArgs e){

if(!IsPostBack){

if(Session["UserName"] !=null){

bindBusDetailsReport();}

else{

Response.Redirect("AdminLogin.aspx");}}}

private void bindBusDetailsReport(){

DataSet dsGetData = new DataSet();

SqlCommand sqlCmd = new SqlCommand();

if (connString.State == ConnectionState.Closed){

connString.Open();}

sqlCmd.CommandType = CommandType.StoredProcedure;

sqlCmd.CommandText = "ispGetBusDetailsForUpdation";

sqlCmd.Connection = connString;

SqlDataAdapter sda = new SqlDataAdapter(sqlCmd);
```

```
sda.Fill(dsGetData);  
if(dsGetData.Tables[0].Rows.Count>0){  
    gdBusDetails.DataSource = dsGetData.Tables[0];  
    gdBusDetails.DataBind();}  
else{  
    gdBusDetails.DataSource = null;  
    gdBusDetails.EmptyDataText = "No Records Found";  
    gdBusDetails.DataBind();}  
protected void gdBusDetails_RowDataBound(object sender, GridViewRowEventArgs e){  
    if(e.Row.RowType==DataControlRowType.DataRow){  
        HyperLink klnikUpdate = (HyperLink)e.Row.FindControl("hlinkUpdate");  
        HiddenField hdnBusID = (HiddenField)e.Row.FindControl("hdnPNRNo");  
        HyperLink hlinkSchedule = (HyperLink)e.Row.FindControl("hlinkAddSchedule");  
        HiddenField hdnRouteID = (HiddenField)e.Row.FindControl("hdnRouteID");  
        klnikUpdate.NavigateUrl = "BusDetails.aspx?BusID=" + hdnBusID.Value;  
        hlinkSchedule.NavigateUrl = "BusScheduleDetails.aspx?BusID=" + hdnBusID.Value + "&RouteID=" +  
        hdnRouteID.Value;  
    }  
}
```

Error and solution

In ASP.NET development, developers may encounter various common errors during the coding, debugging, or deployment process. Here are some typical errors in ASP.NET along with brief descriptions:

- **Null Reference Exception:** Occurs when attempting to access a member (property, method, or field) of a null object reference. Developers need to ensure that objects are initialized before accessing their members.
- **Compilation Errors:** Syntax errors or issues with the structure of the code that prevent successful compilation. These errors need to be addressed before running or deploying the application.
- **404 - Page Not Found:** Indicates that the requested resource is not available on the server. Common reasons include incorrect URLs, missing files, or misconfigured routes in the application.
- **500 - Internal Server Error:** A generic error message indicating an internal server problem. Developers should check server logs for more detailed information, such as unhandled exceptions or configuration issues.
- **Authentication and Authorization Errors:** Occur when there are problems with user authentication or authorization. Developers need to ensure that users have the correct permissions to access certain resources.

To address these errors effectively, developers should use proper coding practices, implement exception handling, regularly test the application, and leverage debugging tools provided by the ASP.NET framework and development environments.

Testing

Software testing is the process of evaluating item to detect difference between given input and expected output also to access the feature of a software item.

There are two types of testing:

1. Black box testing

2. Whitebox testing

Parameter	Black box	White box
Test Coverage	The coverage in black box testing is related to functional aspects and user scenarios. Testers aim to cover various input combinations and user interactions to ensure the software behaves as expected.	Coverage in white box testing refers to the extent to which different parts of the code are executed during testing. Testers aim to achieve high code coverage to ensure that all code paths are tested.
Applicability	Black box testing is suitable for functional testing, integration testing, system testing, and acceptance testing. It is often used to validate that the software meets the specified requirements and behaves correctly.	White box testing is typically used for unit testing, code review, security testing, and performance testing. It helps in identifying issues related to code structure, security vulnerabilities, and optimization.

Limitation

Limitation of Online Bus Booking:

- **Internet Dependency:** Online bus booking is reliant on a stable internet connection. Users in areas with poor connectivity may face challenges accessing and using the system.
- **Technical Glitches:** Like any online platform, technical issues such as server outages, software bugs, or maintenance periods may temporarily disrupt the booking process, leading to inconvenience for users.
- **Security Concerns:** Handling sensitive personal and financial information online raises security considerations. Ensuring robust encryption and secure payment gateways is crucial to protect user data from potential cyber threats.
- **Limited Payment Options:** While many online bus booking platforms support common payment methods, some users may find limitations in terms of accepted payment options, potentially excluding those without access to specific financial services.
- **Device Compatibility:** Users may encounter compatibility issues on certain devices or browsers, affecting the overall user experience. Ensuring cross-device and cross-browser compatibility is essential for inclusivity.
- **Real-Time Seat Availability:** The real-time seat availability displayed on the platform may not always align with the actual availability due to simultaneous bookings or delays in updating the system, leading to discrepancies.
- **No Cancellation Policies:** Users may face restrictions and financial implications because we don't provide cancellation.
- **User Skill and Access:** Some potential passengers, particularly those less familiar with technology, may find it challenging to navigate online booking platforms. Ensuring user-friendly interfaces and providing adequate support is essential.
- **Limited Route Coverage:** Online bus booking systems may not cover all routes or regions, limiting choices for users in certain areas. Expanding route coverage is necessary for providing comprehensive services.
- **Dynamic Pricing:** Dynamic pricing strategies implemented by bus operators may lead to fluctuating ticket prices based on factors like demand, potentially resulting in higher costs during peak travel times.
- **Weather and External Factors:** Unforeseen circumstances such as adverse weather conditions, road closures, or external events may impact bus schedules and services, affecting the accuracy of online bookings.
- **User Privacy Concerns:** Users may have concerns about the privacy of their personal information stored on online platforms. Transparent privacy policies and robust data protection measures are essential to address these concerns.

Addressing these limitations and continuously improving the online bus booking system is vital to ensure a seamless, secure, and accessible experience for all users.

Conclusion

In conclusion, the emergence of online bus booking systems has significantly transformed the landscape of travel planning, offering unprecedented convenience and efficiency. These systems have redefined the way individuals organize their bus journeys, simplifying the process of booking tickets, choosing preferred seats, and ensuring seamless travel experiences.

As we've delved into the realm of online bus booking, it's evident that these systems bring numerous advantages. Users now have the flexibility to explore bus schedules, check seat availability, and make reservations from the comfort of their homes or while on the move. The option to pre-select desired seats enhances the overall travel experience, while secure online payment methods contribute an additional layer of convenience. From the perspective of bus service providers, these systems optimize operations, minimize manual ticket sales, and provide valuable insights into traveler preferences.

Nevertheless, it's important to recognize that online bus booking systems come with their own set of challenges and limitations. These may include reliance on internet connectivity, occasional technical glitches, security considerations, and variations in payment options. Addressing these challenges is crucial to ensure the system's accessibility, user-friendliness, and security for all passengers.

In the era of digital advancements, online bus booking systems have become integral to the travel industry, showcasing the transformative power of technology in enhancing our daily experiences. As these systems evolve, it remains paramount for providers to prioritize user experience, security, and accessibility, adapting to dynamic market trends and user expectations.

To summarize, online bus booking systems have revolutionized the way we approach travel, making the reservation process efficient and user-friendly. They have brought the convenience of planning bus journeys closer to people worldwide. With continuous technological progress, we can anticipate further innovations in this space, promising an even more enhanced and enjoyable travel experience for all.

Reference

Reference websites

- in.redbus.com
- goibibo.com
- makemytrip.com/bus
- yatra.com/buses
- paytm.com/bus
- cleartrip.com/buses
- ixigo.com/bus