CHAPTER 1

INTRODUCTION

The problem that many Restaurant businesses face today is to make sure that they attract new customers and also they keep their existing customers. The cost to attract a new customer is costlier than retaining the old customer. Therefore, there is an argument that for a business, existing customer is worthier than a new customer. In this industry, a customer is likely to return to the restaurant in the future if they received an excellent customer service as well as appetizing food . However, if they had to wait for an unreasonable amount of time or there was a mistake in the order, it’s very unlikely the customer would return.This chapter gives an introduction to the project by defining the problems encountered by restaurants, the main objectives that the system expects to achieve and a brief introduction to existing solutions The project, Restaurant Management System is a web-based application that allows the restaurant manager to handle all restaurant activities online.

Interactive GUI and the ability to manage various food and table bookings and make this system very flexible and convenient. The restaurant manager is a very busy person and does not have the time to sit and manage the entire activities manually on paper. This application gives him the power and flexibility to manage the entire system from a single online system. The system is hence useful for both customers, managers and also for the employees to portable manage the restaurant activities.

1.1 Project Objective

The objective of this project is to build an electronic restaurant management systemusing all of the skills and techniques from the field ensuring that no common development mistakes are reproduced. Project management is critical to all software engineering projects and keeping to a project plan will be of similar importance. One of the main objectives of any business is to maximize profit by increasing efficiency and decreasing overheads without compromising customer satisfaction. Currently, many restaurants use a paper-based system to communicate between the restaurant and kitchen which can be shown to be one of the least efficient approaches. Even though this approach is implemented in successful profitable restaurants, there are several problems which could be seen as reducing the restaurant’s efficiency: By introducing an online restaurant management system these problems can be avoided or improved leading to an increase in profits. Maximizing the profit is one of main objectives of any business. This can achieve by increasing efficiency and decreasing overheads without compromising customer satisfaction. Through better application of daily operations restaurant can increase the efficiency and can offerimproved services to the customers. Because almost all processes are manual and time consuming, all the processes should be automated. The Main Objective: To build a web based restaurant management system for Restaurant. In order to fulfill the main objective following goals have to be achieved.

Improve customer relationship management Proposed system enables visual confirmation to the customers that the order was placed correctly and will decrease difficulties. When the order is ready, kitchen can update the food order status as ready. Same time, customer and the cashier will be notified. This will reduce miscommunication and workload of the cashier. Reducing waiting time of the customer will improve the customer satisfaction. Avoid long queues This solution will help to increase the efficiency of restaurant’s staff. It eliminates paper work and increase level of accuracy. Staff can handle more customers in little time because web based solution can improve speed of service, sales volume and customer satisfaction. Bulk Processing Order retrieval is simple and kitchen can see the order as bulks. Then kitchen can process more orders because they can complete several same type of orders at the same time. Customer feedback Customers can give feedback which is very valuable for improvements of the restaurant. Stock Control All the kitchen ingredient stock levels can be maintained through the system. Proposed system will facilitate restaurant to maintain kitchen stock. Menuitem management. Kitchen can maintain possible meals and can update which item cannot be provided in relevant day. Discounts Providing special discounts and promotions. They can assign,” Item of the Day” for special discount. System can identify whether the customer is new customer or regular customer. For the regular customer, system can provide special discounts. Increase the customer satisfaction and make them retain with the According to a research article written by Horizons in 2006 within restaurant sector was worth billion,any restaurant generating a good business reputation could lead to the making of a very successful and profitable business. The problem for many businesses is to ensure that they not only attract new customers but to ensure they maintain their existing clientele. It has been argued many times that an existing customer is worth more to a business than a new customer as the cost to attract a new customer can be up to five times the cost to retain an old customer. An online article by Paul Lemberg discusses the pros and cons of this argument. Within the restaurant sector, a customer is likely to return to the restaurant in the future if they received an excellent customer service as well as appetising food. However, if theyhad to wait for an unreasonable amount of time or there was a mistake in the order, it's very unlikely the customer would return. Restaurant uses least efficient methods such as paper-based or verbal method to communicate between the restaurant and kitchen. Even though this approach is implemented in successful profitable restaurants, there are several problems which could be seen as reducing the restaurant's efficiency that can be identified using above scenario:

- Sometimes handwriting can lead to miscommunication.
- Order logging is unmanageable.
- Inefficient restaurant-kitchen communication.
- Difficult order tracking and time management.
- Difficult stock management.
- Limited statistical output

Online system that proposed here will simplify the ordering process and it will be helpful to both restaurant and customer. The proposed system will be developed with interactive menus, pop-up messages etc. for the easiness of the user. Customer can select the required food item with available customizations and the can change the order at any time before checking out online. If order is confirmed, it will display a pop-up message as confirmation to the customer. When the order is placed it's recorded in the database and retrieved in real time manner. Through this option Restaurant Employees are allowed to quickly go through the orders which they have received. Therefore, they can process all orders efficiently and effectively with minimal delays and confusion. Therefore a solution to this problem would be to minimise mistakes within the order and bill, and help eradicate delays as well as encouraging team work and communication within the team. The next section will go into the objectives of the proposed solution.

1.2 Existing System

There are many computerised restaurant management systems available but for each system there exist disadvantages or missing features. The most common type of restaurant management system contains a static order entry computer system usually in the shape of a desktop computer with a touch screen. Typically this common approach is adequate to the restaurants requirements but still requires handwritten orders to be relayed to the order entry computer system.

- The existing system is manual
- The restaurant management has to keep record manually
- Lack of security of data.
- More manpower.
- Time Consuming.
- Need manual calculations.
- To overcome the problems of manual system, online Restaurant management system is proposed.
- The main purpose of Restaurant management is to provide online facility to food order and reserve tables.
- Restaurant administrator has to manage other activities like employee data, food orders etc.

The need for a new system arises due to the limitations and inefficiencies of the existing system. The current system may be manual, time-consuming, error-prone, and difficult to manage as the workload increases. A new system can overcome these issues by introducing automation, improving speed, enhancing accuracy, and ensuring better data security. It can also provide a user-friendly interface and make data access and management more efficient. Furthermore, the new system can support future growth and changes, making it more adaptable and scalable. Therefore, implementing a new system is essential for improving overall performance and achieving organizational goals effectively.

The need for a new system becomes essential when the existing system fails to meet current business or organizational requirements. Traditional or outdated systems often suffer from several issues such as slow processing speed, manual data entry, high chances of human error, poor data security, lack of integration with modern technologies, and limited scalability. As organizations grow, the complexity of managing operations using the old system increases, leading to inefficiency, frustration among users, and loss of productivity. In contrast, a new and improved system can offer automation of repetitive tasks, real-time data processing, advanced reporting tools, and user-friendly dashboards. It enhances data accuracy, reduces operational costs, and ensures secure and centralized storage of information. Moreover, a modern system is more adaptable to future technological advancements and can easily be updated or expanded as per the changing needs of the organization. Hence, to remain competitive, efficient, and customer-focused, the development and implementation of a new system is not just a need, but a strategic necessity.

1.3 Scope of the Project

The scope of the project defines the boundaries, objectives, and deliverables of the system being developed. It clearly outlines what the project will include and what it will not. This project aims to design and implement a system that improves the current process by making it more efficient, accurate, and user-friendly. The scope includes gathering user requirements, designing the interface, developing the core functionality, testing the system, and deploying it for real-time use. It also involves providing user training and documentation to ensure smooth adoption. However, the scope does not include future upgrades, third-party system integration, or maintenance beyond the initial deployment phase unless specified. Defining the scope helps avoid misunderstandings, ensures proper resource allocation, and keeps the project on track by focusing only on the planned features and goals.

The scope of the project outlines the specific objectives, features, and functions that the new system will cover. It defines the overall boundaries within which the system will be developed and operated. This project aims to replace or enhance the existing manual or outdated system with a more efficient, accurate, and user-friendly digital solution. Key activities included in the scope are requirement gathering, system design, coding, testing, deployment, and basic user training. The system will automate key processes, reduce human errors, improve data storage and retrieval, and offer better reporting and analysis capabilities.

The scope also clarifies what is not included in the project to avoid scope creep and manage expectations. It does not cover major future upgrades, integration with external or third-party applications, or long-term system maintenance unless specified in the plan. Any features or functionalities not discussed in the initial requirement phase will be considered out of scope. By clearly defining the scope, project stakeholders can focus on the core goals, ensure efficient use of time and resources, and deliver a system that meets the agreed-upon objectives within the timeline and budget.

1.Objectives of the Project

The main objective of the project is to design and develop a system that improves efficiency, accuracy, and user satisfaction compared to the existing process. It will automate manual tasks, reduce human errors, and streamline data management. The system will be designed with a user-friendly interface and easy navigation, ensuring smooth usability.

2. Features Included in the Scope

The scope includes gathering user requirements, designing the user interface, developing the core functions, testing the system for bugs or errors, deploying it on a live environment, and providing basic user training. The system will support functionalities like data entry, record management, search and filter options, report generation, and login authentication (if needed).

3. **Deliverables of the Project**

Deliverables include a fully functioning software system, user documentation, training materials, and a project report. All these deliverables will be completed and handed over within the defined timeline. Additionally, a demonstration will be conducted to explain how the system works.

4. **Limitations / Out of Scope**

The scope does **not** include long-term maintenance, future system upgrades, integration with third-party software, or mobile app versions (unless mentioned separately). Only the functionalities agreed upon during the planning phase will be developed. Any new features requested later will be treated as separate enhancements.

5. **Importance of Defining the Scope**

Clearly defining the scope helps in managing time, resources, and expectations effectively. It prevents delays, avoids confusion, and ensures the team stays focused on achieving the desired outcomes without deviating from the original plan.

CHAPTER 2

BACKGROUND STUDY AND ANALYSIS

This chapter gives an insight to Restaurant Management system similar in nature to that of the one being developed in this project. It also gives a brief introduction to the importance of requirement gathering, a discussion on development methodologies available as well as a justification on the platform and software used in this project. This chapter presents a detailed study and analysis of the existing system or process, along with the problems and limitations it currently faces. Understanding the background is essential for identifying the gaps that the new system needs to address. The study includes how the current system works, who the users are, what challenges they face, and what improvements are required. By analyzing the strengths and weaknesses of the present method, this chapter helps to build a solid foundation for the proposed solution. It also highlights the importance of modernizing the system to meet present and future requirements efficiently. This analysis will guide the design and development of a more robust, user-friendly, and functional system.

2.1 Overview

Use of Restaurant management system helps to manage staff, sales, customers in easier manner,Restaurant Management helps to run a restaurant in more efficient manner. It helps to manage orders, menus, inventory and sales. This kind of system usually contains hardware, such a cash register, large displays or touch panels, barcode scanner and receipt printer as well as software.It provides a comprehensive tool that allows to see restaurant and its needs at a glance, which can simplify the workload on a day-to-day basis. What is a restaurant management system? “A restaurant management system (RMS) is a type of point-of-sale (POS) software specifically designed for restaurants, bars, food trucks and others in the food service industry. Unlike a POS system, and RMS encompasses all back-end needs, such as inventory to staff management. A system typically includes software and hardware, such as registers, scanners and receipt printers.” Types of RMScurrently available Because every restaurant has unique needs, there are different RMS types to choose from. To determine which type best suits the restaurant, one has to focus on the features that business requires and how important each of those features [9].

1.End to end

This is the most robust and comprehensive type of RMS. Main features include core POS, inventory control, CRM, staff, menu, order and payment management, technical support, and reporting and analytics. Depending on the RMS vendor, developer may be able to mix and match features.

2. POS

This is the core of the system and allows to integrate it with third-party systems for inventory, accounting, marketing and other key systems.

3.iPad or Android only

Most systems are designed to run only on one device type to maintain the integrity of the system. Determine which device type will use in the restaurant.

4.General POS

This system is designed for businesses that have both retail and food services available. It offers seamless crossover with add-on modules

2.2 Comparison of development life cycles

It is very important to identify best method to do the development of the system. Variety of software development life cycle models are available. Success of any project is depending on selecting a suitable model for a specific development project. In the background study, waterfall, spiral, incremental/iterative (agile), v model is considered as the alternative software development model . It helps to manage orders, menus, inventory and sales. This kind of system usually contains hardware, such a cash register, large displays or touch panels, barcode scanner and receipt printer as well as software.It provides a comprehensive tool that allows to see restaurant and its needs at a glance, which can simplify the workload on a day-to-day basis. What is a restaurant management system? “A restaurant management system (RMS) is a type of point-of-sale (POS) software specifically designed for restaurants, bars, food trucks and others in the food service industry. Unlike a POS system, and RMS encompasses all back-end needs, such as inventory to staff management. A system typically includes software and hardware, such as registers, scanners and receipt printers.”

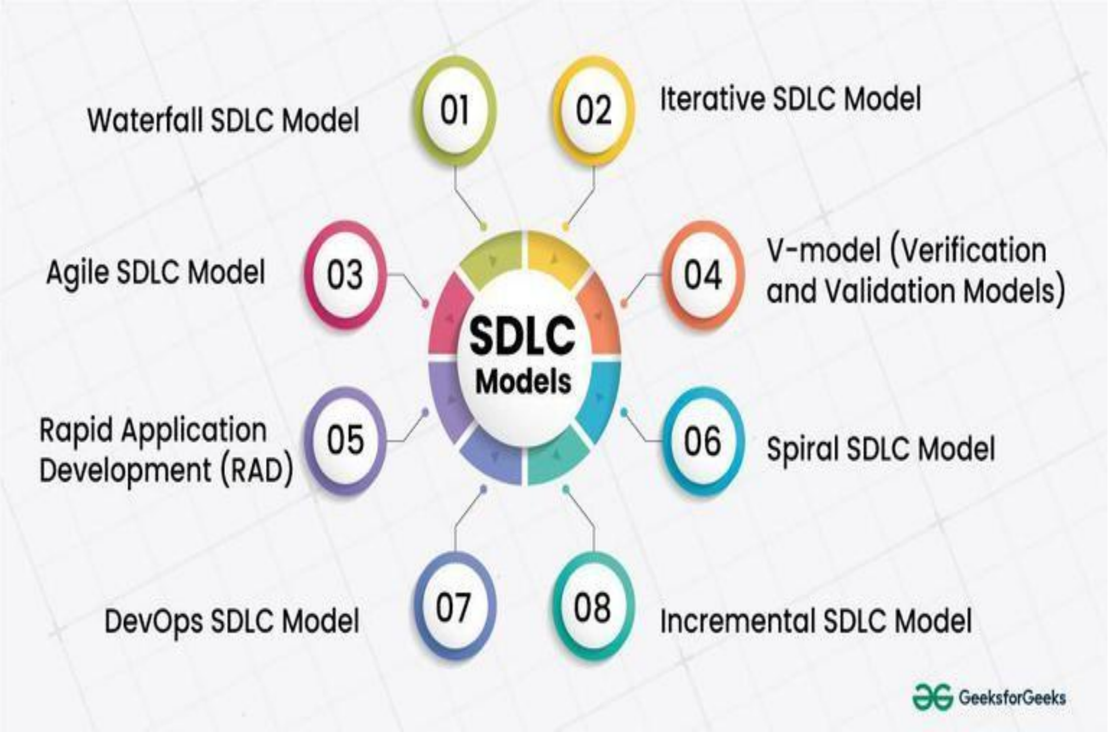


Fig 2.1 software development life

The Software Development Life Cycle (SDLC), which is a systematic process used to develop high-quality software in a structured and efficient manner. It consists of several well-defined phases, each with specific goals and deliverables that guide the development from start to finish.

2.3 Need for the Proposed System

The food and hospitality industry is rapidly evolving, with increasing demand for speed, convenience, and accuracy in service. In many restaurants, the traditional methods of managing operations—such as taking orders on paper, manually booking tables, or relying on verbal communication between staff—are becoming outdated and inefficient. These manual processes often lead to common issues like misplaced or incorrect orders, overlapping table reservations, delayed food preparation, and poor coordination between kitchen and serving staff. During busy hours or peak seasons, these problems multiply, resulting in longer wait times, customer dissatisfaction, and loss of business. Furthermore, with the growing reliance on digital services, customers now prefer the ability to book tables online, view menus from their phones, and place orders without delays or confusion. Therefore, there is a strong need for a digital solution that can simplify these processes and bring more efficiency to restaurant management. The proposed system aims to meet this need by offering a centralized platform where customers can interact with the restaurant digitally, and staff can manage daily operations more effectively. It not only reduces human errors and workload but also improves service quality, ensures faster order processing, and enhances the overall dining experience. By adopting such a system, restaurants can modernize their services, stay competitive in the market, and build stronger relationships with customers through reliable and seamless service.

2.4 Literature Review

A literature review is an important part of any research or project report because it helps us understand what has already been studied or developed in the same area. In the case of developing a restaurant management or online food ordering system, the literature review focuses on exploring existing systems, previous research, technologies used, and the strengths and weaknesses of those solutions. Many studies and projects have shown that digital platforms significantly improve the speed and accuracy of restaurant services. For example, research has shown that using a web-based ordering system reduces waiting times and errors in orders. Other systems, like mobile apps for food delivery or online reservation platforms, have also proven to make the customer experience more convenient and efficient.

A literature review helps to build a strong foundation for any project by exploring the research, technologies, and solutions that have already been developed in the same domain. In the context of restaurant management and online food ordering systems, many researchers, developers, and organizations have explored various approaches to improving the efficiency and reliability of restaurant operations. For example, several studies highlight how digital menu systems and online reservation platforms not only reduce staff workload but also improve customer satisfaction by allowing users to view menus, place orders, and book tables from their mobile devices or computers. Research papers and real-world case studies often show that automating the order-taking and food preparation process minimizes human error, prevents order duplication, and streamlines communication between the kitchen and service staff.

CHAPTER 3

SYSTEM DESIGN AND ANALYSIS

The system design of the Restaurant Menu Page, which is a key component of the overall restaurant management system. System design is a crucial phase that translates the user requirements and functional specifications into a structured solution. In this chapter, the layout, functionality, and flow of the menu page are discussed in detail. It includes both the front-end design**,** which is responsible for user interaction and display, and the back-end structure, which handles data processing and storage. The goal is to create a user-friendly interface that allows customers to easily browse menu items, view categories, prices, and descriptions, and select their desired items efficiently. This design ensures that the system is both visually appealing and functionally effective, providing a smooth experience for both users and administrators.

The system design for the Restaurant Menu Page, which plays a vital role in enhancing the digital dining experience for customers. The design phase serves as the bridge between the planning and implementation stages of software development. It involves creating a blueprint that defines how the system will function, look, and interact with users. For a restaurant menu page, the design must ensure easy navigation, clear presentation of food items, categorization (e.g., starters, main course, desserts, beverages), price display, item descriptions, and optional images of dishes to attract customers.

The system design includes both logical design (such as database schemas, data flow diagrams, and relationships between components) and physical design (such as the visual layout and user interface elements). Additionally, the design takes into account responsiveness for various devices like mobiles and tablets, as many customers may access the menu through smartphones. The system also considers potential features like filtering items by category, adding items to a cart or order list, and integration with other modules such as online ordering or table booking systems. Security, performance, scalability, and ease of maintenance are also considered during the design process. By the end of this chapter, the complete structural overview of the restaurant menu page is outlined, which sets the foundation for development and implementation.

3.1 Objectives of the Design

The primary objective of designing the Restaurant Menu Page is to create an intuitive, user-friendly, and visually appealing interface that enhances the overall dining experience for customers, whether they are ordering online or browsing in-store through digital displays. The design aims to ensure that users can easily view all available food items, along with details such as names, descriptions, images, prices, and categories (e.g., starters, main course, desserts, beverages), with minimal effort and maximum clarity.Another important objective is to provide smooth navigation and quick access to menu sections using filters or tabs, making the selection process efficient and hassle-free.

The design should also be responsive, meaning it should adapt well to various screen sizes, such as mobile phones, tablets, and desktops, offering a consistent experience across devices. Additionally, the system design should support backend integration so that restaurant administrators can easily add, edit, or remove menu items without technical expertise. The design should also consider future scalability, allowing new features like item customization, availability status (e.g., "out of stock"), and online ordering to be integrated later without major changes. Finally, the design must ensure data accuracy, security, and performance. Information displayed to the user should be dynamically fetched from the database to avoid inconsistencies, and the interface must load quickly, even with a large number of menu items. By meeting these objectives, the system design lays a strong foundation for building a reliable and effective restaurant menu application.

3.2 Functional Requirements

Functional requirements are the specific features, actions, and operations that a system or application must be able to perform. They define to meet the needs of its users and stakeholders. In software development, these requirements describe the interactions between the system and its users — such as receiving input, processing it, and producing expected results. For example, in a restaurant menu website, functional requirements would include displaying food items with details like name, price, and image, allowing users to filter items by category (like Starters or Desserts), or letting customers search for specific dishes like "Pizza". They may also include backend tasks like enabling the admin to add or update menu items. Functional requirements are essential because they serve as the foundation for system design, coding, and testing. Without them, developers wouldn't have a clear understanding of the features the system must provide.

3.2 METHODOLOGY

This is focused about stakeholders of the system, and requirements of the system. Also this chapter will look at the system design. System design is explained with the aid of diagrams to illustrate graphically certain sections of the software system.

This chapter presents the methodology adopted for the development of the system, focusing on the identification of stakeholders, collection of system requirements, and the design approach used to create a functional and efficient application. Stakeholders such as restaurant owners, customers, staff members, and system administrators were first identified to understand their specific needs and expectations. Requirements were gathered through interviews, observations, and questionnaires, and categorized into functional requirements—such as menu display, order management, and table booking—and non-functional requirements including usability, performance, security, and reliability. Once the requirements were defined, system design was undertaken to translate them into a logical and implementable structure. Various diagrams were used to represent the system design visually and to clarify the components and their interactions. These included context diagrams to show the relationship between the system and external entities, use case diagrams to highlight user interactions, entity-relationship diagrams (ERDs) to model the database structure, data flow diagrams (DFDs) to trace the movement of data, and architecture diagrams to depict the overall system framework. This structured design approach ensured that each module of the system was well-defined, maintainable, and scalable. The development process followed a systematic model, such as the waterfall or iterative approach, progressing through stages of requirement analysis, design, implementation, testing, and deployment. This methodology ensured that the final system not only met user expectations but was also robust, flexible, and prepared for future enhancements.

3.3 Kitchen Staff and serving staff

The restaurant environment, kitchen staff and serving staff face several operational challenges that can impact service quality and efficiency. One of the key difficulties faced by kitchen staff is managing and remembering a large number of food orders, especially during peak hours. Often, the same food item is ordered multiple times by different customers, prompting staff to prepare those items in bulk. However, without a structured system, it becomes difficult to track which orders are complete and which still need to be packed or served. A system that categorizes orders based on food items and displays them in bulk can significantly improve workflow in the kitchen. This allows cooks to prepare large quantities of the same dish at once and mark items as ready in a more organized manner.Additionally, for takeaway orders, packing becomes a complex task if individual orders are not clearly separated or labeled. A feature that allows staff to view and manage orders individually, especially for take-away, ensures better accuracy and minimizes the risk of mixing up customer orders. Clear separation of orders enhances packing efficiency and customer satisfaction. On the other hand, serving staff benefit greatly from a system that allows them to arrange and deliver orders based on the time they were placed. Implementing a “first come, first served” view helps streamline the serving process and ensures fairness, reducing wait times for customers. Furthermore, digital notifications or order status updates can keep both kitchen and serving staff informed in real-time, reducing confusion and improving coordination between departments. By integrating these features into the restaurant system, both kitchen and service operations become more manageable, efficient, and customer-focused.

3.3.1 Cashiers

Currently they have to remember all the item codes, if there is a new employee it is very difficult to bill the order in cash register and its time consuming. If there is a method to see the items in category wise that would be easy and efficient. By observing the working environment, it is very clear about the drawbacks of the current system. Current method causes customer dissatisfaction and due to that competitors also getting advantages.

The Cashier is a key user role in the restaurant management system. Cashiers are responsible for handling customer transactions, processing payments, and managing order receipts. The system must allow cashiers to access an intuitive interface where they can view active orders, calculate total bills including taxes or discounts, and process payments through various methods such as cash, card, or digital wallets. Additionally, the cashier should be able to generate printed or digital receipts and update the order status once the payment is complete. For security and accountability, the system should also track login sessions, associate transactions with specific cashier accounts, and allow supervisors to review cashier activity. The cashier module should prioritize ease of use, speed, and accuracy to ensure smooth customer service during busy hours.

3.4 Nature of the current method

The current method is based on papers. Only way to track order is cash register and the bill that given to the customer. Menu cards are also paper based or verbal. Those menu cards are printed and if any change is required restaurant must redesign the menu cards and reprint them. This is a big waste and it is costly and it is impossible to reprint every time even for a miner change. Current system is time consuming. Customers have to waste their time in queue and place the order. Several case of serving a wrong order are also happened due to miscommunication. Therefore, briefly, For placing any orders customers have to visit restaurant to know about food items and then place order and pay. In this method time and manual work is required. It is difficult to ensure whether the order has placed correctly or not because cash register codes are unknown to the customer. Sometime there are miscommunication between kitchen and front desk. No database is present; therefore, analysis is impossible. MenuManagement Function Provides functionality for the Admin User only. It will not be available to any other 23 users of the system like Restaurant Employees or Customers. Using a graphical interface, it will allow an Admin to manage the menu that is displayed to users of the web ordering system: Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs for Microsoft Windows, as well as web sites, web applications and web services. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code. Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs for Microsoft Windows, as well as web sites, web applications and web services. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Name	Description	Task/Responsibilities
Admin	Refers privileged to the actor who has control over the whole system	Admin can manage employee, food order, table reservation, parties, leave, etc.
Customer	Customer of the system	Customer can order food and reserve table etc.
Staff	Employee of Restaurant	Employee can edit their profile and apply for leave etc.

3.1 SYSTEMREQUIREMENT PHASE

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs for Microsoft Windows, as well as web sites, web applications and web services. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

● Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a forms designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that enhance the functionality at almost everylevel—including adding support for source-control systems (like Subversion) andadding new tool sets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle(likethe Team Foundation Server client: Team Explorer).

CHAPTER 4

SYSTEM COMPONENT DESIGN AND DEVELOPMENT

This chapter provides a detailed explanation of the key modules and technologies that make up the ResqConnect system. Each module is designed to handle specific tasks such as registering emergency complaints, providing chatbot assistance, displaying real-time alerts, and sending notifications. To support these features, ResqConnect uses a combination of modern web technologies, APIs, and database solutions.

The chapter is divided into two main sections. The first section describes the core modules of the application and their functionalities. The second section highlights the technologies used to build the platform, including the frontend and backend tools, database system, APIs for external data and communication, and tools used during development. Together, these modules and technologies form a reliable and efficient emergency response system.

4.1 Modules Description

Modules Description is a part of the Software Requirements Specification (SRS) document or system design report. This section provides a detailed breakdown of **each major component (module)** that makes up the entire system. It helps developers, designers, and stakeholders understand how the system works as a whole by dividing it into manageable, functional parts.

1. Menu Management Module

Purpose: To allow administrators to manage food items available in the restaurant.

Features:

Interface to add, update, or remove menu items.Categorization of dishes (e.g., Starters, Main Course, Desserts, Beverages). Real-time updates of menu data in the database. Image and price management for each item.

2. Order Processing Module

Purpose: Facilitate the placement, tracking, and updating of customer orders.

Features:

User-friendly interface for selecting menu items and placing orders. Real-time order tracking and status updates (e.g., Preparing, Ready, Served).Support for both dine-in and takeaway orders. Integration with the kitchen display system (optional).

3. Billing and Payment Module

Purpose: Handle customer billing and manage multiple payment options.

Features:

Automatic bill generation with tax and discount calculations.Support for various payment methods (cash, card, UPI, digital wallets).Receipt printing and digital invoice generation. Payment tracking linked with order ID and table number.

4. Admin Dashboard Module

Purpose: Enable administrators to monitor and manage overall restaurant operations.

Features:

View and manage all current and past orders. Monitor sales, reservations, and staff activity. Generate daily, weekly, or monthly reports.Manage menu items, table status, and user roles securely.

5. Customer Feedback Module

Purpose: Collect and manage customer feedback to improve service quality.

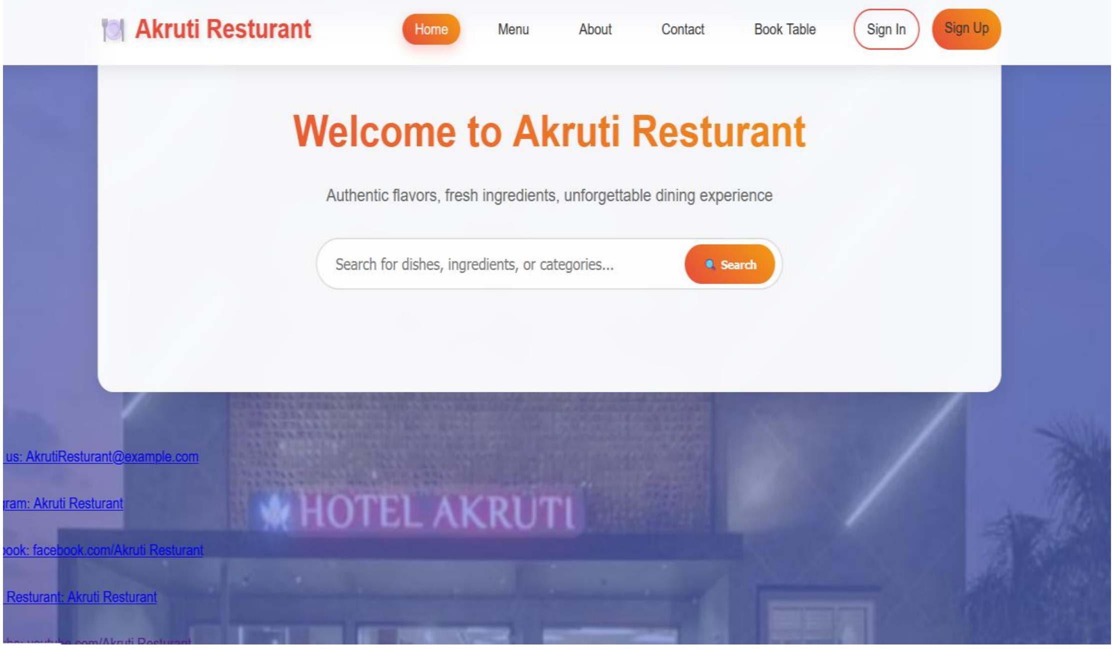
Features:

Feedback form accessible after order completion. Rating system for food quality, service, and ambiance.Admin access to view and analyze customer reviews.

4.2 User Interface

1. Home page

The Home Page of the restaurant website serves as the welcoming gateway for visitors, designed to create a strong first impression with an attractive layout and vibrant images of delicious dishes. It provides a brief introduction to the restaurant, highlighting its ambiance, specialty cuisines, and commitment to quality service. Visitors can easily navigate through key sections such as the menu, table booking, and contact information. The homepage often includes promotional banners, customer reviews, and links to featured dishes or chef's specials. The goal of the Home Page is to engage users quickly, convey the restaurant’s brand identity, and guide them effortlessly to other parts of the site.


Akruti Resturant
Home
Menu
About
Contact
Book Table
Sign In
Sign Up

Welcome to Akruti Resturant

Authentic flavors, fresh ingredients, unforgettable dining experience

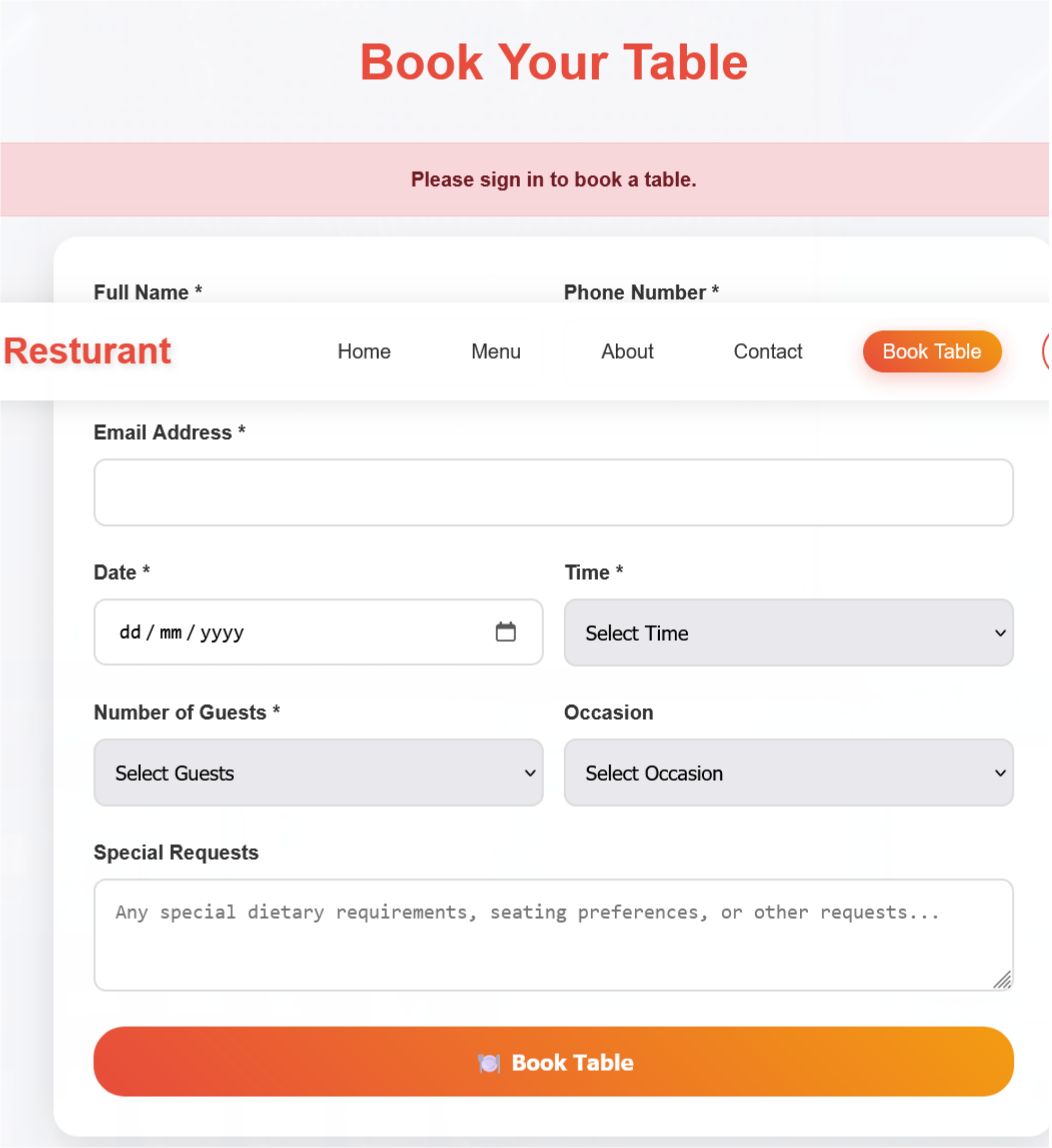
Search for dishes, ingredients, or categories...
Search
us: AkrutiResturant@example.com
ram: Akruti Resturant
ook: facebook.com/Akruti Resturant
Resturant: Akruti Resturant
HOTEL AKRUTI

Fig 4.1 Home page

The Home Page of the restaurant website is the central hub where visitors are introduced to the essence of the dining experience offered. It features an eye-catching banner or slideshow displaying signature dishes, special offers, or festive menus to immediately attract attention. A welcoming message or tagline sets the tone for the restaurant's brand—whether it's cozy and family-friendly, elegant and upscale, or trendy and modern. Users can quickly access the menu, make a table reservation, or explore ongoing promotions with clearly placed navigation buttons.

2.Book Table page

The **Book Table** section of the restaurant website allows customers to easily reserve a table online without the need to call or visit in person. This feature is designed for convenience, offering a user-friendly form where guests can select the date, time, number of guests, and provide their contact details. It often includes options to leave special requests, such as seating preferences or celebration notes. Once the form is submitted, customers typically receive an instant confirmation via email or SMS.



Book Your Table

Please sign in to book a table.

Full Name *

Phone Number *

Email Address *

Date *

dd / mm / yyyy

Time *

Select Time

Number of Guests *

Select Guests

Occasion

Select Occasion

Special Requests

Any special dietary requirements, seating preferences, or other requests...

Fig.4.2 Book Table Page

The **Book Table** feature in the restaurant system allows customers to reserve a table in advance, ensuring convenience and better dining experience. This module captures essential customer details such as name, contact number, number of guests, preferred date, and time of reservation. Once the form is submitted, the system checks for table availability and confirms the booking.

CHAPTER 5

RESTAURANT TECHNOLOGY STACK

The design of the Restaurant Menu System is centered around creating a user-friendly, responsive, and efficient interface that enhances the overall dining and management experience. This system is developed with both customers and restaurant staff in mind, ensuring smooth navigation, real-time data handling, and an intuitive layout. From the customer's perspective, the design focuses on ease of use and visual clarity. The menu items are categorized clearly (e.g., Starters, Main Course, Desserts, Beverages) with appealing images, descriptions, prices, and availability status. Interactive features such as search bars, filters, and "Add to Cart" options provide a seamless browsing and ordering experience. The design also adapts to various devices through responsive web design, ensuring usability on desktops, tablets, and mobile phones. On the administrative side, the system includes a clean and functional admin dashboard where staff can manage menu items, update prices, mark items as available or out of stock, and review orders or reservations. The backend logic and database integration are designed to support real-time updates, allowing changes made by the admin to reflect instantly on the customer-facing interface. The overall design approach balances aesthetics and functionality, using modern web technologies such as HTML5, CSS3, JavaScript, PHP, and MySQL. It follows best practices in UI/UX design, ensuring accessibility, performance, and clarity. The system is modular in nature, allowing for future enhancements such as digital payments, feedback collection, or integration with third-party ordering platforms. In conclusion, the design of the Restaurant Menu System is aimed at delivering a fast, engaging, and error-free experience for all users, while streamlining restaurant operations through a well-structured backend interface.

5.1 Overview

This chapter presents the system design of the Restaurant Menu Management System. It aims to provide a comprehensive understanding of the system's internal structure, workflow, and interaction between various components through visual representations. The design phase serves as a blueprint for developers and ensures that the application is logically structured, scalable, and meets the functional requirements outlined in earlier chapters. The chapter includes multiple diagrammatic representations, each serving a specific purpose in visualizing how the system operates and how users interact with it. These diagrams include:

- Use Case Diagram – Illustrates the different actors (e.g., Customer, Admin) and the actions they can perform within the system, such as browsing the menu, placing orders, updating items, etc.
- Class Diagram – Represents the structure of key objects in the system, including MenuItem, Order, Reservation, and User, along with their attributes and relationships.
- Data Flow Diagram (DFD) – Demonstrates how data moves through the system, such as how user input is processed, stored in the database, and retrieved by the frontend.
- Entity Relationship Diagram (ERD) – Depicts the relationships between entities in the database, including tables like menu, orders, users, and reservations.
- Sequence Diagram – Shows step-by-step interactions between system components during operations like placing an order or making a reservation.
- System Architecture Diagram – Provides a high-level view of the system's architecture, including frontend, backend, database, and any APIs or external services.
- These design models not only help visualize system logic and flow but also guide implementation by clarifying how components interact and how data is handled securely and efficiently. By breaking down the system into diagrams, developers, testers, and stakeholders can better understand system behavior, spot potential issues early, and ensure the application meets both technical and user expectations.

This project has been designed using numerous diagrammatic techniques, the most general modelling language to describe both the structure and behaviour of a software system is Unified Modelling language (UML). Use case diagrams have already been used in the requirements analysis as a way to graphically overview the order process within the system. Other diagrams from the UML family are used in the design stage to show the structure and behaviour of numerous sophisticated design features.

5.2 Component Diagram

A component diagram is part of UML and its main purpose is to show the structural relationship between components in the system. A component diagram is useful for this system as it shows the higher architecture. Relational Database Management System (RDBMS) A relational database management system (RDBMS) is a database managed system based around a relational model and are the corner stone's to many software systems including web based systems. RDBMS are one of the most popular data storage methods out in the market and offer many advantages including:

- Fast data extraction using structured query language (SQL).
- Good management of data and security through the management system.
- Good level of data consistency.
- Advanced features including functions and triggers.
- Requirement of a data model to be developed; leading to long termcost effectiveness.

In industry, there are numerous expensive highly functional RDMBSs including Oracle and SQLServer that are very popular and offer technical support. However, there are also numerous open-source solu- tions with many adjudged to be as good or better and are becoming even more popular with small scale software systems. Extensible Markup Language (XML) XML is a markup language that was designed to transport and store data and is another example of a persistent data storage technique. However, it is not a predefined language thus all tags must be defined and due to its hierarchically data structure all 38 elements must be promoted or demoted. XML could be used in two different ways in data storage; storing the XML documents within a database or having the XML documents as the fundamental unit of storage. In both cases the XML can be queried using either XPATH or XQUERY which are query languages for extracting data from XML documents. elements must be promoted or demoted.XML could be used in two different ways in data storage; storing the XML documents within a database or having the XML documents as the fundamental unit of storage. In both cases theXML can be queried using either XPATH or XQUERY which are query languages for extracting data from XML documents. Storage Method Chosen The main difference between XML and RDBMS is that XML is hierarchical and RDBMS is relational . As restaurant data can be best represented in a hierarchical way one would believe that XML would be the best approach but it’s not always that straight forward. SQL is an extremely flexible and robust querying language and for the queries required and the type of software system begin designed, it was concluded that RDBMS would be the best storage

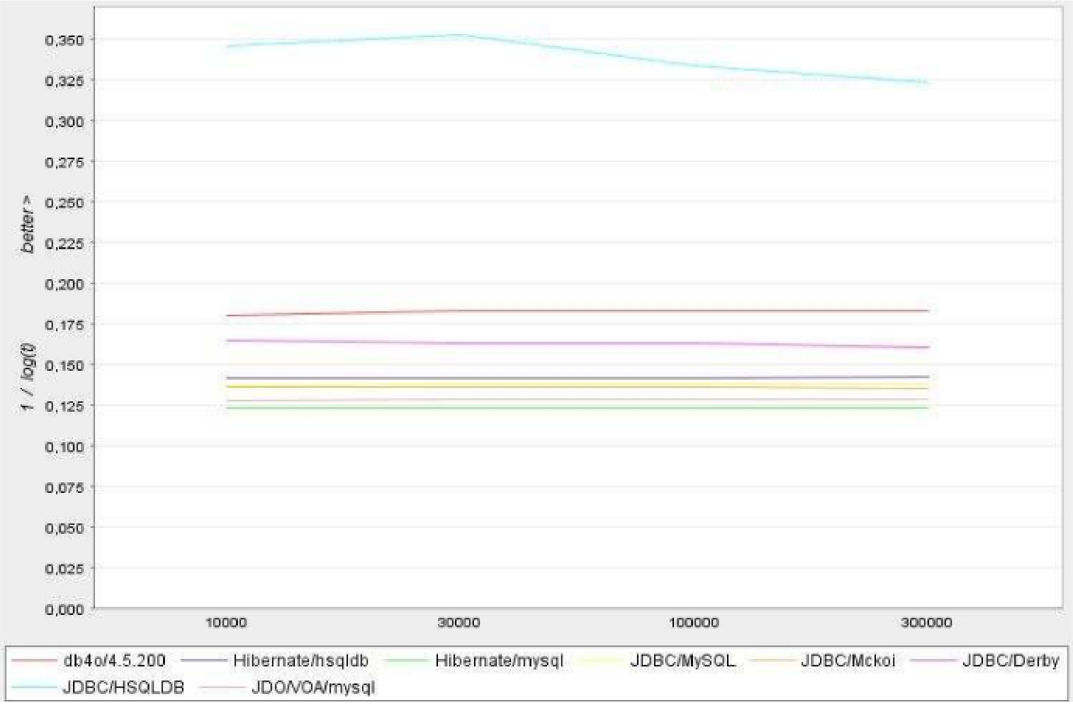


Figure 5.1: Database comparison diagram

elements must be promoted or demoted.XML could be used in two different ways in data storage; storing the XML documents within a database or having the XML documents as the fundamental unit of storage. In both cases theXML can be queried using either XPATH or XQUERY which are query languages for extracting data from XML documents. Storage Method Chosen The main difference between XML and RDBMS is that XML is hierarchical and RDBMS is relational. As restaurant data can be best represented in a hierarchical way one would believe that XML would be the best approach but it’s not always that straight forward. SQL is an extremely flexible and robust querying language and for the queries required and the type of software system begin designed, it was concluded that RDBMS would be the best storage method. The next choice was to decide on the type of RDBMS to use. As discussed there are many open source RDBMSs available for us to choose from and for the main reason of experience, MySQL was the preferred option. it shows just how competitive the performances of different RDBMSs are. The main difference between XML and RDBMS is that XML is hierarchical and RDBMS is relational. As restaurant data can be best represented in a hierarchical way one would believe that XML would be the best approach but it's not always that straight forward. SQL is an extremely flexible and robust querying language and for the queries required and the type of software system begin designed, it was concluded that RDBMS would be the best storage method. The next choice was to decide on the type of RDBMS to use. As discussed there are many open source RDBMSs available for us to choose from and for the main reason of experience, MySQL was the preferred option. it shows just how competitive the performances ofdifferent RDBMSs.

The selection of an appropriate technology stack is crucial for the successful development and implementation of the Resturant Management System. A technology stack comprises the combination of programming languages, frameworks, databases, and tools used to build and run the application. Choosing the right stack ensures that the system is efficient, secure, scalable, and easy to maintain. For this project, a blend of widely used and reliable technologies has been selected to cover both the frontend user interface and the backend server operations, along with the database management and development environment, providing a comprehensive and cohesive solution. A technology stack consists of a combination of programming languages, frameworks, databases, servers, and development tools that work together to deliver a seamless user experience. In this project, the technology stack is carefully chosen to balance ease of development, cost-effectiveness, and robust functionality. The frontend technologies ensure an intuitive and responsive interface for users, while the backend technologies manage the complex business logic, data storage, and server-side operations. Additionally, reliable database management systems and development tools enhance the overall efficiency and maintainability of the system. Together, these technologies provide a strong foundation to build a secure, scalable, and user-friendly Resturant management system tailored to meet the specific needs of managing girls' hostel operations.

5.3 Frontend languages

The frontend of the Resturant Menu Management System is developed using essential web technologies such as HTML, CSS, and JavaScript. HTML (HyperText Markup Language) forms the basic structure of the web pages, defining elements like headings, forms, tables, and buttons. CSS (Cascading Style Sheets) is used to style these elements, ensuring the website is visually appealing and responsive across different devices and screen sizes. JavaScript adds interactivity to the user interface by enabling dynamic content updates, form validation, and enhanced user experience features. Additionally, frameworks like Bootstrap are often incorporated to simplify responsive design and provide ready-made components. Together, these frontend technologies create a user-friendly and visually engaging interface that allows students, wardens, and administrators to easily interact with the hostel management system.

5.3.1 HTML

HTML (HyperText Markup Languag**e)** is the foundational language used to create the structure and content of the Resturant Management System's web pages. It organizes the information by defining various elements such as headings, paragraphs, forms, tables, buttons, and links, which are essential for displaying student details, room allocation, fee payment forms, and more. HTML provides a semantic framework that allows browsers to render content correctly and ensures accessibility for all users. By using HTML5, the latest version, the project benefits from enhanced features like multimedia support, improved form controls, and better page structuring, which contribute to a modern, efficient, and user-friendly interface for managing hostel operations.

HTML5, the latest standard of HyperText Markup Language, plays a crucial role in the development of the Girls' Hostel Management System by providing a rich set of semantic elements that improve the structure, accessibility, and SEO of the web application. Unlike earlier versions, HTML5 introduces new elements such as <section>, <article>, <header>, <footer>, and <nav>, which help organize content in a meaningful way, making it easier for browsers and assistive technologies to interpret the page layout. This semantic structuring enhances user experience, especially for students and administrators who rely on clear navigation and well-organized information. HTML5 also supports multimedia elements like <audio> and <video> without requiring external plugins, enabling integration of announcements or tutorials within the hostel portal. Additionally, HTML5 forms offer advanced input types (e.g., date, email, number) and validation attributes, which improve the accuracy and usability of data entry tasks such as student registration and fee submission. Combined with CSS3 and JavaScript, HTML5 provides a robust foundation for building responsive, interactive, and accessible web applications tailored to the needs of hostel management.

5.3.2 CSS

CSS (Cascading Style Sheets) is a cornerstone technology used in the Girls' Hostel Management System to control the presentation, layout, and visual aesthetics of the web pages. CSS3, the latest version, introduces advanced styling features such as animations, transitions, gradients, and flexible box layouts (Flexbox), which help create a modern, responsive, and visually appealing user interface. By separating content (HTML) from design (CSS), the project ensures maintainability and scalability, allowing developers to easily update the look and feel without affecting the underlying structure. Media queries in CSS3 enable the system to adapt seamlessly to different screen sizes and devices, providing an optimal user experience for students and staff accessing the portal from desktops, tablets, or smartphones. Additionally, CSS frameworks like Bootstrap can be integrated to accelerate development by offering pre-designed components and grid systems, ensuring consistency and responsiveness across the application. Overall, CSS empowers the Resturant Management System to deliver an engaging and user-friendly interface that enhances usability and accessibility.

CSS3 also supports animations, transitions, and media queries, allowing developers to add dynamic effects and ensure the interface remains user-friendly on desktops, tablets, and smartphones. By separating content (HTML) from style (CSS), the system's design becomes easier to maintain and update. Furthermore, CSS supports the use of variables (custom properties) and pseudo-classes/elements which help write cleaner, reusable, and more modular code, enhancing overall efficiency. Integrating CSS frameworks such as Bootstrap can further speed up development by providing pre-built responsive components and consistent styling.

General CSS Syntax

```
/* Style for all paragraphs */
p {
  color: #333333;
  font-size: 16px;
  line-height: 1.5;
}
```

5.3.3 JavaScript

JavaScript is a versatile, client-side scripting language used extensively in the Resturant Management System to enhance user interaction and provide dynamic functionality. It enables the system to respond instantly to user actions without needing to reload the entire web page, creating a smoother and more efficient user experience. JavaScript powers features such as form validation, interactive menus, real-time notifications, and dynamic content updates like room availability or fee status. Modern JavaScript (ES6 and beyond) offers powerful constructs including arrow functions, promises, async/await, and modules, which facilitate writing clean, modular, and maintainable code. Additionally, JavaScript can manipulate the Document Object Model (DOM) to dynamically update the webpage based on user input or server responses. When combined with AJAX (Asynchronous JavaScript and XML), it allows for seamless communication with the backend without page refreshes, essential for real-time data handling in hostel management tasks. Libraries and frameworks such as jQuery or React can further simplify complex interactions, although this project primarily focuses on vanilla JavaScript for core functionalities.
General JavaScript Syntax

```
// Function to validate a form field
function validateName() {
 let name = document.getElementById('studentName').value;
if (name === '') {    alert('Please enter your name.');
 return false;
  }
  return true;
}
// Event listener for form submission

document.getElementById('registrationForm').addEventListener('submit',
function(event) {

  if (!validateName()) {

   event.preventDefault(); // Prevent form submission if validation fails
```

}

JavaScript’s event-driven programming model is extensively used to handle user iputs, such as validating registration forms in real-time, dynamically updating room availability, and providing immediate feedback to users without waiting for server responses. The use of the Fetch API and Promises allows asynchronous communication with the backend PHP scripts, enabling the system to retrieve or send data in the background without full page reloads—improving performance and user experience significantly.JavaScript also leverages the Document Object Model (DOM) API to manipulate webpage elements dynamically. This includes showing or hiding sections, updating content based on user roles (students, wardens, admins), and creating interactive UI avaScript’s event-driven programming model plays a critical role in enhancing the interactivity and responsiveness of modern web applications. It is widely used to handle user inputs in real time—such as validating form fields during user registration, checking for empty inputs, invalid email formats, or mismatched passwords—all without requiring a full page refresh. This instant feedback helps prevent user errors and ensures data integrity. Advanced JavaScript features such as event listeners**,** callbacks**,** and dynamic rendering allow for a truly interactive experience where the user interface updates in response to their actions without delay. Additionally, modern libraries like jQuery or frameworks such as Vue.js or React (if included in future development) can further simplify and enhancthe implementation of these dynamic features.

JavaScript’s event-driven programming model is extensively used to handle user inputs, such as validating registration forms in real-time, dynamically updating room availability, and providing enhancthe implementation of these dynamic features.JavaScript’s event-driven programming model is extensively used to handle user inputs, such as validating registration forms in real-time, dynamically updating room availability, and providing enhancthe implementation of these dynamic features.JavaScript’s event-driven programming model is extensively used to handle user inputs, such as validating registration forms in real-time, dynamically updating room availability, and providing enhancthe implementation of these dynamic features.JavaScript’s event-driven programming model is extensively used to handle user inputs, such as validating registration forms in real-time, dynamically updating room availability, and providing immediate feedback to users without waiting for server responses. The use of the Fetch API and Promises allows asynchronous communication with the backend PHP scripts, enabling the system to retrieve or send data in the background without full page reloads—improving performance and user experience significantly.JavaScript also leverages the Document Object Model (hiding sections, updating content based on user roles (students, wardens, ad andcreating interactive UI avaScript’s event-driven programming model plays

Fig. 5.2 Frontend Language

a critical role in enhancing the interactivity and responsiveness of modern web applications. It is widely used to handle user inputs in real time—such as validating form fields during user registration, checking for empty inputs, invalid email formats, or mismatched passwords—all without requiring a full page refresh. This instant feedback helps prevent user errors and ensures data integrity. Advanced JavaScript features such as event listeners**,** callbacks**,** and dynamic rendering allow for a truly interactive experience where the user interface updates in response to their actions without delay. Additionally, modern libraries like jQuery or frameworks such as Vue.js or React (if included in future development) can further simplify and enhance the implementation of these dynamic features.

The backend of the Resturant Management System is responsible for handling data storage, processing user requests, managing authentication, and enabling smooth communication between the frontend and the database. Below are the backend languages and technologies used for this project. The backend of the Girls' Hostel Management System is developed using PHP, a widely-used server-side scripting language known for its efficiency in web development and seamless integration with databases. PHP handles all major functionalities such as student registration, room and bed allocation, staff management, and fee collection. The system uses a MySQL database to store and manage structured data including student profiles, room availability, payment records, and login credentials. The backend communicates with the frontend through form submissions and HTTP requests, allowing real-time data processing and updates. PHP sessions are used to manage user authentication and access control, ensuring that sensitive operations are restricted to authorized users like wardens or administrators. Proper validation and sanitization techniques are applied to secure the system against common vulnerabilities like SQL injection and cross-site scripting (XSS). This backend setup offers a stable, secure, and cost-effective solution for managing day-to-day hostel operations efficient.

5.4 PHP

PHP (Hypertext Preprocessor) is used as the primary backend programming language. PHP is a popular open-source server-side scripting language that is especially well-suited for web development. It allows dynamic content generation, form handling, and seamless communication with databases. In this project, PHP is used to handle essential operations such as student registration and login, room and bed allocation, staff and warden management, and hostel fee tracking. PHP interacts with a MySQL database to store and retrieve data securely and efficiently. Built-in PHP functions are used for session management, data validation, and server-side processing, ensuring secure and smooth functioning of the system. Its ease of use, wide community support, and compatibility with various web servers make PHP an ideal choice for building a reliable and scalable Recturent Management System**.** PHP works seamlessly with HTML and can be embedded directly into HTML code, making it easy to create interactive web applications. It supports various databases, with MySQL being the most commonly used. For this project, PHP communicates with a MySQL database to store and retrieve information about students, rooms, payments, and staff. PHP also includes built-in features for session handling, form validation, file uploads, and security measures such as input sanitization and password hashing.

These features are essential for ensuring that the web application is secure and functions correctly across different use cases.

Why PHP Is Suitable for This Project

- Easy to learn and implement
- Supports integration with HTML and databases
- Provides built-in security features
- Fast performance for small to medium-sized applications
- Widely supported on most web hosting platforms

5.5 Tools and IDEs

Visual Studio Code (VS Code) was the primary code editor due to its lightweight design, user-friendly interface, and support for extensions that aid in PHP, HTML, CSS, and JavaScript development Web browsers like Google Chrome and Mozilla Firefox were used for testing and debugging the frontend, with the help of built-in developer tools. These tools collectively ensured efficient development and testing of the project, improving both code quality and development speed. Visual Studio Code (VS Code) is the main code editor used in the development of the Resturant Web Project. It is a free, open-source, and lightweight IDE developed by Microsoft, widely popular among web developers for its speed, flexibility, and wide range of features. VS Code supports multiple programming languages including PHP, HTML, CSS, JavaScript, and more, making it ideal for full-stack web development. It includes built-in support for syntax highlighting, code formatting, and error detection. Additionally, its extension marketplace allows developers to install helpful tools such as PHP IntelliSense, Live Server for real-time browser preview, and Git integration for version control. With its user-friendly interface and powerful features, VS Code helped streamline the coding, testing, and debugging processes during the development of the project. One of the key advantages of VS Code is its extensive extension marketplace, which allows developers to enhance its functionality by installing plugins such as PHP IntelliSense (for code suggestions and auto-completion), Live Server (to view real-time changes in the browser), Prettier (for consistent code formatting), and Debugger for PHP (for step-by-step error checking). It also has built-in Git integration, making it easy to manage version control and collaborate on code. The editor includes features like syntax highlighting, bracket matching, code snippets, multi-cursor editing, and a terminal window, all in a user-friendly and lightweight interface. These tools significantly improved productivity, helped catch errors early, and made the development of the Resturant Managment management system faster and more efficient.

CONCLUSION

The development of the Restaurant Menu Page represents a significant step toward modernizing and streamlining the customer experience within the restaurant industry. By leveraging a combination of intuitive design and practical functionality, this system ensures that users can easily browse, search, and select food items in a convenient and visually engaging way. Key features such as real-time item availability, category-based organization, and responsive layout across devices not only enhance the user's browsing experience but also contribute to faster and more accurate order placement.

The intuitive design plays a key role in ensuring that customers—regardless of their age or familiarity with technology—can navigate the menu with ease. Clear categories, high-quality food images, simple navigation, and appealing layout design help users browse and discover items effortlessly. Whether they are looking for vegetarian dishes, beverages, or desserts, everything is just a few clicks away. Practical functionalities such as real-time item availability provide transparency to customers. For example, if a particular dish is out of stock, the system will immediately reflect this, helping avoid disappointment and reducing order errors. The category-based organization groups menu items logically (e.g., Starters, Main Course, Beverages), which speeds up the search process and enhances the customer’s decision-making.

REFRENCES

1. J. Duckett, *HTML and CSS: Design and Build Websites*, 1st ed., Wiley, Indianapolis, IN: 2011, pp. 1–490. ISBN: 978-1118008188.
2. E. Freeman, E. Robson, *Head First JavaScript Programming*, 1st ed., O'Reilly Media, Sebastopol, CA: 2014, pp. 1–700. ISBN: 978-1449340131.
3. Twitter Inc., *Bootstrap 5 by Example*, 2nd ed., Packt Publishing, Birmingham, UK: 2022, pp. 1–300. ISBN: 978-1803248052.
4. L. Ullman, *PHP and MySQL for Dynamic Web Sites*, 5th ed., Peachpit Press, San Francisco, CA: 2017, pp. 1–672. ISBN: 978-0134301846.
5. Apache Friends, "XAMPP Documentation," Version 8.2.0, 2023. [Online]. Available: https://www.apachefriends.org/index.html