**PROJECT SYNOPSIS (ETCS 460)**

On

# Queue Reducing Application

Submitted for partial fulfilment of award of the degree of

## Bachelor of Technology

In

**Computer Science & Engineering**

Submitted by

**Nikhil – 01618007221**

**Harshit Gupta – 00918007221**

**Arjun Singh – 02318007221**

**Abhishek Jangid– 01518007221**

Under the Guidance of

## **Mr. Aniket Singh**

**(Assistant professor)**



**Department of Computer Science & Engineering**

# DELHI TECHNICAL CAMPUS, GREATER NOIDA

**(Affiliated Guru Gobind Singh Indraprastha University, New Delhi) Session 2023-2024 (8th SEM)**

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S.no** | **Topic** | **Page No.** |
| 1 | Introduction | 4 | |
| 2 | Problem Statements | 5 | |
| 3 | Objectives | 6 | |
| 4 | Feasibility Study | 7 | |
| 5 | Need and Significance | 9 | |
| 6 | Intended User | 10 | |
| 7 | Abbreviations and Acronyms | 11 | |
| 8 | Literature Review | 12 | |
| 9 | Proposed Methodology in brief | 14 | | |
| 10 | Hardware Requirements | 16 | | |
| 11 | Software Requirements | 17 | | |
| 12 | Diagram | 18 | | |
|  | 12.1 E-R Diagram | 18 | | |
|  | 12.2 Data Flow Diagram | 19 | | |
|  | 12.3 Gantt Chart | 21 | | |
| 13 | References | 22 | | |

**Table of Figures**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Name** | **Page No.** |
| Figure 1 | ER Diagram | 18 |
| Figure 2 | Level 0 DFD | 19 |
| Figure 3 | Level 1 DFD | 19 |
| Figure 4 | Level 2 DFD | 20 |
| Figure 5 | Gantt Chart | 21 |

1. **Introduction**

Welcome to the presentation on Optimizing Efficiency: A Queue Deletion Application for Food Service. This application aims to streamline the process of managing food orders and improving customer satisfaction. The purpose of this application is to manage and delete orders from a queue in a food service setting. It provides a simple interface for staff to view queued orders and delete them as necessary once they are completed or no longer required.

1. **Problem Statement**

"Develop a Queue Deletion Application for a food service establishment, enabling staff to manage and remove orders efficiently. The application should allow staff to view queued orders, delete completed or cancelled orders, and search for specific orders by order number or customer name. Real-time updates should reflect changes in the queue, and secure authentication should restrict access to authorized personnel. Deploy the application with robust security measures and consider future enhancements for integration with POS systems and analytics features."

1. **Objectives**

The objective of the Queue Deletion Application for a food service establishment is to streamline order management by providing staff with a user-friendly interface to view, delete, and search for orders in the queue. The application aims to enhance operational efficiency by enabling real-time updates and secure authentication, ensuring only authorized personnel can access and manage the queue. Future objectives include integration with POS systems and implementing analytics features to optimize order processing and improve customer service.

* + **Limitations**

One potential limitation of the Queue Deletion Application is its reliance on internet connectivity for real-time updates and functionality. In cases where internet access is intermittent or unavailable, the application's performance may be impacted, potentially leading to delays in order management and updates. Additionally, the applications effectiveness may be limited if staff members are not adequately trained to use the system or if there are hardware or software compatibility issues with the deployed environment. Furthermore, the application may face scalability challenges if the volume of orders exceeds its capacity to handle concurrent user requests or database transactions.

1. **Feasibility Study**
   * **Preliminary Analysis**

* **Identify the Problem**: Clearly define the problem your app aims to solve, such as reducing wait times and improving customer satisfaction in queues.
* **Propose Solutions**: Outline potential solutions, including the development of a queue deduction app with features like real-time updates, virtual queue management, and notification systems.
  + **Market Analysis**
* **Target Audience**: Identify who will use the app (e.g., businesses, public institutions, event organizers) and who the end consumers will be (e.g., customers, patients, attendees).
* **Competition**: Analyze existing solutions and competitors. Understand their strengths and weaknesses and how your app will differentiate itself.

**Demand Assessment**: Evaluate the demand for such an app. Conduct surveys, focus groups, or interviews with potential users to gauge interest and willingness to adopt your solution.

* + **Technical Feasibility**
* **Technology Requirements**: Determine the technology stack needed to develop the app, including software, hardware, and network requirements.
* **Development Team**: Assess if you have the right team in place or if you need to hire developers with specific skills.
* **Development Timeline**: Estimate how long it will take to develop the MVP (Minimum Viable Product) and future iterations.
  + **Economic Feasibility**
* **Cost Analysis**: Estimate the total cost of developing, launching, and maintaining the app. Include costs for the development team, marketing, technology, and any other overheads.
* **Revenue Model**: Outline how the app will generate revenue. This could be through direct sales, subscription models, advertising, or in-app purchases.
* **Break-even Analysis**: Calculate how long it will take for the app to become profitable.
  + **Legal Feasibility**
* **Regulatory Requirements**: Identify any legal or regulatory requirements affecting your app, such as data protection laws (e.g., GDPR in Europe), digital accessibility standards, and industry-specific regulations.
* **Intellectual Property**: Ensure that your app idea, name, and any unique features can be legally protected.
  + **Operational Feasibility**
* **User Adoption**: Assess how you will encourage adoption among your target audience. Consider partnerships, marketing strategies, and incentives.
* **Support and Maintenance**: Plan for ongoing support and maintenance of the app, including updates, bug fixes, and customer service.
  + **Scheduling Feasibility**
* **Timeline**: Create a detailed project timeline, including key milestones for development, testing, launch, and post-launch evaluations.
* **Risk Management**: Identify potential risks to your timeline, such as technical challenges, team availability, or market changes, and how you will mitigate these risks.

1. **Need and Significance**

The need for the Queue Deletion Application in a food service setting arises from the necessity to streamline order management processes. It significantly reduces manual efforts by providing a digital platform for staff to efficiently view, delete, and search for orders in the queue. This not only improves operational efficiency but also enhances customer satisfaction by ensuring timely order processing and minimizing errors.

The significance of the application lies in its ability to optimize workflow, reduce wait times, and improve overall service quality. By enabling real-time updates and secure authentication, it enhances transparency and accountability in order management. Additionally, the application's future enhancements, such as integration with POS systems and analytics features, further enhance its significance by providing valuable insights for decision-making and improving business performance. Overall, the Queue Deletion Application plays a crucial role in enhancing operational efficiency, customer satisfaction, and business productivity in the food service industry.

1. **Intended User**

The intended users of the Queue Deletion Application are staff members working in food service establishments, including restaurants, cafes, fast-food chains, and other similar businesses. These users typically include waitstaff, kitchen staff, managers, and other personnel involved in order processing and management.

Specifically, the application is designed to cater to staff members responsible for managing the order queue, including:

1. Waitstaff: Responsible for taking orders from customers and inputting them into the system.
2. Kitchen Staff: Responsible for preparing and fulfilling orders based on the queue displayed in the application.
3. Managers: Responsible for overseeing order management processes, monitoring queue status, and making decisions to optimize workflow.
4. Support Staff: Responsible for handling customer inquiries, order modifications, and resolving any issues related to order processing.

The application is user-friendly and intuitive, catering to users with varying levels of technical expertise. It offers features such as real-time updates, search functionality, and secure authentication to enhance user experience and ensure efficient order management. Overall, the Queue Deletion Application is tailored to meet the needs of staff members involved in order processing and management within food service establishments.

1. **Abbrevation & Arconyms**

* QDA: Queue Deletion Application
* FS: Food Service
* POS: Point of Sale
* CRUD: Create, Read, Update, Delete (basic functions of persistent storage)
* UI: User Interface
* UX: User Experience
* API: Application Programming Interface
* DB: Database
* HTTP(S): Hypertext Transfer Protocol (Secure)
* JWT: JSON Web Token (for authentication)
* MVC: Model-View-Controller (software design pattern)
* ORM: Object-Relational Mapping
* CSS: Cascading Style Sheets
* HTML: Hypertext Markup Language
* JS: JavaScript
* CLI: Command Line Interface
* QA: Quality Assurance
* MVP: Minimum Viable Product
* CDN: Content Delivery Network
* SaaS: Software as a Service

These abbreviations and acronyms help to streamline communication and documentation within the development and deployment process of the application.

1. **Literature Review**
2. **Technological Trends and Innovations**:
   * Explore scholarly articles, journals, and conference proceedings focusing on emerging technologies relevant to queue management systems in food service.
   * Investigate advancements in real-time communication protocols, database management systems, and frontend frameworks that enhance the performance and functionality of queue deletion applications.
   * Examine how technologies such as IoT (Internet of Things), cloud computing, and mobile applications are being integrated to optimize order management processes.
3. **User Interface Design and User Experience**:
   * Review literature on user interface design principles, usability guidelines, and human-computer interaction (HCI) theories applicable to queue deletion applications.
   * Analyse case studies or research papers that evaluate different UI/UX design patterns and their impact on user satisfaction and efficiency in food service environments.
   * Explore methodologies such as user-centered design and iterative prototyping to understand how user feedback is incorporated into the development of queue deletion applications.
4. **Best Practices in Order Management**:
   * Identify studies or industry reports that highlight best practices and strategies for efficient order management in food service establishments.
   * Examine the role of queue deletion applications in optimizing order processing workflows, reducing wait times, and improving overall service quality.
   * Investigate how data analytics and business intelligence tools are used to analyse order data and identify trends that inform decision-making in food service operations.
5. **Security and Privacy Considerations**:
   * Review literature addressing security challenges and privacy concerns associated with queue deletion applications, especially in the context of handling sensitive customer information and transaction data.
   * Explore encryption techniques, access control mechanisms, and compliance standards (e.g., GDPR, PCI DSS) relevant to securing data transmitted and stored within queue deletion applications.
6. **Case Studies and Implementation Strategies**:
   * Examine case studies or real-world examples of queue deletion applications deployed in food service establishments, highlighting successful implementation strategies, challenges encountered, and lessons learned.
   * Evaluate the impact of queue deletion applications on operational efficiency, customer satisfaction, and business performance through quantitative and qualitative analysis.
7. **Proposed Methodologies**
   * **Requirement Analysis and Planning:**
   * Collaborate with cafeteria staff and users to gather detailed requirements.
   * Define user stories and prioritize features.
   * Plan sprints and iterations for development.
   * **UI/UX Design:**
   * Design intuitive interfaces using ReactJS.
   * Conduct usability testing for efficient navigation.
   * Iterate based on feedback for various devices.
   * **Backend Development with Express:**
   * Set up Express server for handling HTTP requests.
   * Implement RESTful APIs for user authentication, menu retrieval, and order processing.
   * Incorporate middleware for security measures.
   * **Database Design and Integration:**
   * Design PostgreSQL schema for storing menu items, users, and orders.
   * Implement CRUD operations using Sequelize or another ORM.
   * Ensure data integrity and indexing for performance.
   * **Testing, Deployment, and Maintenance:**
   * Write unit tests for frontend and backend components.
   * Conduct integration tests for interactions between layers.
   * Deploy using Heroku or AWS.
   * Monitor performance and apply CI/CD practices.
   * Provide ongoing support and enhancements.
8. **Hardware Requirement**
   * **Servers**

* Processor: Multi-core processor (e.g., Intel Xeon, AMD Ryzen) to handle concurrent requests and database operations efficiently.
* RAM: Sufficient RAM capacity (e.g., 8GB or more) to support the application's memory requirements and ensure smooth performance.
* Storage: SSD (Solid State Drive) or HDD (Hard Disk Drive) with ample storage capacity for storing application files, databases, and log files.
  + **Network Infrastructure**
* High-speed internet connection with adequate bandwidth to support real-time communication between the client and server components of the application.
* Router or switch to facilitate network connectivity and ensure reliable data transmission between devices.
  + **Client Devices**
* Desktop Computers: Standard desktop computers or workstations with modern web browsers (e.g., Chrome, Firefox, Safari) to access the web-based application.
* Mobile Devices: Smartphones or tablets with internet connectivity to access the application via a mobile-responsive web interface.
  + **Peripheral Devices**
* Printers: Optional printers for generating order receipts, invoices, or other documents directly from the application.
* POS Devices: Integration with POS (Point of Sale) systems or hardware peripherals for processing payments and synchronizing order data.
  + **Backup and Redundancy**
* Backup Storage: External backup storage devices or cloud backup solutions to regularly backup application data and ensure data recovery in case of hardware failures or data loss.
* Redundant Components: Redundant power supplies, RAID configurations, and failover mechanisms to minimize downtime and ensure continuous operation of critical hardware components.
  + **Security Hardware**
* Firewalls: Hardware firewalls to protect the server and network infrastructure from unauthorized access and cyber threats.
* Intrusion Detection/Prevention Systems (IDS/IPS): Hardware-based IDS/IPS appliances to monitor network traffic and detect/prevent security breaches.

1. **Software Requirements**
   * **Operating System**

Window-7 and above

* + **Chrome Version**

Chrome 1.0

Chrome 2.0

Chrome 3.0

Chrome 4.0

* + **Programing Language**

- Reactjs (Version – 17)

- AXIOS (Version –0.21.4)

- NodeJs (Version –16.13.1)

- ExpressJs (Version –4.17.1)

- PostgreSQL (Version – 14.02)

* **Code Editor**V.S. Code -1.63.2