**Canteen Management System for Retro Campus Canteen**

|  |  |  |
| --- | --- | --- |
| **Futurist IT** | **Retro Campus Canteen** | **Version 1.1** |

**Software Development Project Management**

**January 22, 2025**

**Supervised By: Md. Anwarul Kabir**

**Group: 03**

|  |  |
| --- | --- |
| **Name** | **ID** |
| **Md Noman Islam** | **21-45453-3** |
| **Md Nasimul Islam** | **21-45465-3** |

**Table of Contents**

|  |  |
| --- | --- |
| 1. **Introduction…………………………………………………………………….**    1. **Intention of the Document…..…………………………………………….............**    2. **Target Audience......…………………………………………………………………**    3. **Brief Objectives of the Document.…………………………………………………** | **3**  **3**  **3** |
| 1. **Project Title.…………………………………………………………………….** | **4** |
| 1. **Objectives……………………………………………………………………….**   **3.1 Specific Goals…………………………………………….…………….…………….** | **4**  **4** |
| 1. **Justification……………………………………………………………..……….** | **5** |
| 1. **Systems Overview………………….……………………………………………**    1. **Key Features and Modules…………………………………………………………**    2. **Use Case Diagram Analysis………………………….…………………………….** | **6**  **7**  **7** |
| 1. **Stakeholders Analysis…………………………….…………………………….** | **8** |
| 1. **Feasibility Study………………………………………………………...………**   **7.1 Technical Feasibility …………………………………………………………………**  **7.2 Financial Feasibility………………………………………………………………….** | **9**  **9**  **10** |
| 1. **System Components………………………………………………..……………**   **8.1 Order Management Component…………………….………………………………**  **8.2 Billing and Payment Component……………………………………………………**  **8.3 Inventory Management Component………………………………………………..**  **8.4 Reporting and Analytics Component……………………………………………….**  **8.5 Customer Feedback Component……………………………………………………**  **8.6 Authentication and Authorization Component……………………………………**  **8.7 Notification and Alert Component…………………………………………………**  **8.8 Database Component……………………………………………………………….**  **8.9 Integration Component………………………………………………………..……** | **11**  **11**  **11**  **11**  **11**  **11**  **12**  **12**  **12** |
| 1. **Process Model to Be Followed…………………………………………………**    1. **Selected Process Model…………………………………………………………….**    2. **Steps of the Agile Process Model ……………………………………....................**    3. **Justification for Using Agile……………………………………………………….** | **13**  **13**  **13**  **14** |
| 1. **Effort Estimation…………………………………………………..,………….**    1. **Work Breakdown Structure (WBS)……………………………………..……….**    2. **Role Allocation and Effort ………………………………………………...………**    3. **Team Size Estimation………….……………………………………………...……**    4. **Justification of Efforts …………..……………………………………………..…**    5. **Activity Based WBS …………………………………………………………….…** | **15**  **15**  **16**  **16**  **16**  **16** |
| 1. **Activity Network Diagram…………………………………………….………**   **11.1 Identify Activities and Tasks………………………………………………………**  **11.2 Define Dependencies Between Activities…………………………………………**  **11.3 Arrange Activities in Sequence…………………………………………………...**  **11.4 Assign Durations…………………………………………………………………..**  **11.5 Use Nodes for Activities……………………………………………………………** | **19**  **19**  **19**  **20**  **20**  **20** |
| 1. **Risk Analysis…………………………………………………………………...** | **21** |
| 1. **Estimation of Resources and Budget…………………………………………** | **22** |
| 1. **Conclusion………………………………………………………………………** | **27** |

**1.0 Introduction**

This document serves as an extensive and detailed project plan for the development and implementation of the Canteen Management System for Retro Campus Canteen. It is crafted to guide all stakeholders involved in the project, providing a comprehensive understanding of the project's scope, objectives, and implementation strategies.

**1.1 Intention of the Document**

The primary purpose of this document is to outline the detailed plan for building an efficient, user-friendly, and technology-driven system that streamlines the operations of Retro Campus Canteen. It aims to document all technical, operational, and managerial aspects, ensuring that every phase of the project is executed smoothly and effectively. The document also serves as a reference for all stakeholders to understand their roles and responsibilities throughout the project lifecycle.

**1.2 Target Audience**

The target audience for this document includes:

* **Primary Stakeholders:** Retro Campus Canteen management, who will oversee the project's outcomes.
* **Secondary Stakeholders:** Students and staff, who will use the system for ordering, billing, and providing feedback.
* **Development Team:** Including project managers, developers, designers, and quality assurance testers who will execute the project.
* **Future Maintainers:** IT personnel tasked with system maintenance post-deployment.

**1.3 Brief Objectives of the Document**

1. **Define Project Goals and Scope:** Provide clear insights into the primary objectives of the Canteen Management System, such as enhancing customer experience, reducing operational inefficiencies, and improving inventory management.
2. **Specify Features and Functionalities:** Elaborate on the key modules, including order management, billing, inventory, and reporting systems, to ensure alignment with user needs.
3. **Clarify Resource and Technical Requirements:** Detail the human, physical, and technical resources essential for the project's success.
4. **Establish Development Roadmap:** Present the timeline, cost estimates, and milestones, ensuring all stakeholders have a clear understanding of the project's progression.
5. **Facilitate Communication:** Provide a framework for ongoing collaboration and updates among all stakeholders.
6. **Document Maintenance and Support:** Outline the post-deployment support structure and associated costs, ensuring sustainability.

This introduction sets the stage for a detailed exploration of the project, ensuring all stakeholders have a clear understanding of its importance and their roles in achieving its success.

**2.0 Project Title: Canteen Management System for Retro Campus Canteen**

This project title precisely defines the proposed software system, highlighting its purpose to modernize and streamline the operations of Retro Campus Canteen. The title reflects the system's focus on enhancing efficiency, improving customer satisfaction, and integrating advanced technological solutions into the canteen's daily operations.

**3.0 Objectives**

The overarching objective of the proposed Canteen Management System is to streamline and modernize the operational processes of the Retro Campus Canteen. The system aims to enhance efficiency, improve customer satisfaction, and ensure effective resource utilization by integrating technology-driven solutions into the canteen's daily operations.

**3.1 Specific Goals**

1. **Simplify Order Placement and Billing:** Enable customers to place orders digitally and generate bills automatically, reducing manual effort and minimizing errors.
2. **Real-Time Order Tracking:** Allow both customers and staff to track order statuses in real time, ensuring transparency and reducing wait times.
3. **Efficient Inventory Management:** Implement an inventory tracking system that updates stock levels in real time, generates alerts for low stock, and facilitates procurement planning.
4. **Comprehensive Reporting and Analytics:** Provide detailed reports on sales, customer behavior, and operational performance to aid in data-driven decision-making.
5. **Enhance Customer Engagement:** Introduce a customer feedback module, enabling users to rate services and share reviews, fostering continuous improvement.
6. **Support for Multiple Payment Methods:** Offer a variety of payment options, including digital wallets, credit/debit cards, and cash, to accommodate diverse user preferences.
7. **Mobile and Web Accessibility:** Develop a user-friendly system accessible via both mobile and web platforms to maximize convenience for all stakeholders.
8. **Cost-Effective Operations:** Reduce operational inefficiencies, minimize delays, and optimize resource utilization to ensure long-term cost savings.

These objectives collectively aim to deliver a robust, user-friendly, and efficient system that meets the needs of Retro Campus Canteen and its customers.

**4.0 Justification**

**4.1 Purpose of the System**

The purpose of the proposed Canteen Management System is to revolutionize the operations of Retro Campus Canteen by leveraging advanced technology to enhance efficiency, accuracy, and customer satisfaction. The system will address existing challenges such as manual order management, billing delays, and inventory discrepancies, ensuring a seamless and user-friendly experience for all stakeholders.

**4.2 The key drivers for implementing this system are:**

1. To reduce operational inefficiencies and save time in processing orders and transactions.
2. To enhance the overall customer experience by providing quick and transparent services.
3. To ensure effective inventory control, minimizing wastage and preventing stock shortages.
4. To provide actionable insights through reporting and analytics, aiding in data-driven decision-making.

**4.3 Who Will Benefit?**

1. **Retro Campus Canteen Management:**
   * Gain access to real-time sales and inventory data.
   * Reduce labor costs associated with manual operations.
   * Achieve better control over financial and resource planning.
2. **Students and Staff (End Users):**
   * Enjoy faster order placement and delivery.
   * Access multiple payment options for convenience.
   * Provide feedback easily, ensuring their concerns are addressed.
3. **Canteen Staff:**
   * Streamline order tracking and reduce workload.
   * Focus more on food preparation and customer service.
   * Avoid errors in billing and inventory management.
4. **FuturistIT Development Team:**
   * Showcase their expertise in developing innovative solutions.
   * Build a scalable system that can serve as a blueprint for similar projects in other institutions or sectors.

**4.4 Long-Term Impact**

The implementation of this system will lead to sustained operational improvements, cost reductions, and customer loyalty for Retro Campus Canteen. Furthermore, the system's flexibility and scalability can pave the way for future enhancements and expansions.

**5.0 Systems Overview**

The proposed Canteen Management System for Retro Campus Canteen is designed to optimize operations, improve customer satisfaction, and streamline processes. The system includes features for order management, billing, inventory tracking, reporting, and customer feedback. It will be accessible via both web and mobile platforms, catering to the convenience of users and staff.

**5.1 Key Features and Modules**

Based on the use case diagram provided in the attached document, the system consists of the following primary modules and their interactions:

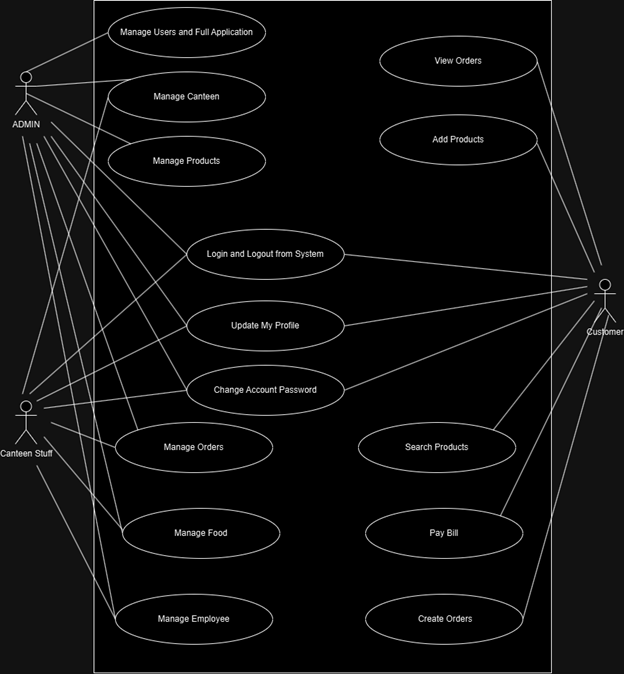
1. **Order Management Module:**
   * Digital Order Placement: Customers can place orders via the application.
   * Real-Time Order Tracking: Staff can monitor order progress, and customers can track the status of their orders.
2. **Billing and Payment Module:**
   * Automatic Bill Generation: Ensures accurate billing for every transaction.
   * Multiple Payment Methods: Supports cash, digital wallets, and card payments.
3. **Inventory Management Module:**
   * Real-Time Stock Updates: Keeps inventory levels current to prevent stockouts.
   * Low Inventory Alerts: Sends alerts to staff when stock levels fall below a set threshold.
4. **Reporting and Analytics Module:**
   * Sales and Performance Reports: Provides detailed insights into daily operations.
   * Customer Feedback Analysis: Aggregates customer reviews and ratings for continuous improvement.
5. **Customer Feedback Module:**
   * Rating and Review System: Allows customers to rate services and provide valuable feedback.

**5.2 Use Case Diagram Analysis**

The use case diagram in the provided document illustrates the interactions between the system's actors and its primary functionalities. Below is an analysis:

1. **Actors:**
   * Customer: Engages with the system to place orders, track orders, make payments, and provide feedback.
   * Staff: Manages order preparation, inventory updates, and bill processing.
   * Admin/Manager: Monitors overall operations, analyzes reports, and manages inventory.
2. **Use Cases:**
   * Place Order: Customers submit orders through the system.
   * Track Order: Both customers and staff can monitor the progress of orders in real time.
   * Make Payment: Customers finalize transactions using various payment options.
   * Provide Feedback: Customers share their reviews and ratings.
   * Update Inventory: Staff ensures inventory data reflects the current stock levels.

The diagram visually represents these interactions, highlighting the seamless integration of the system's features with its user base. It ensures that all actors, from customers to administrators, interact efficiently to achieve their respective objectives.



**6.0 Stakeholders Analysis**

Stakeholder analysis is a critical part of the project as it identifies the individuals, groups, and organizations involved or impacted by the Canteen Management System for Retro Campus Canteen. Understanding their roles, expectations, and contributions is essential to ensure the project's success.

**Primary Stakeholders**

1. **Retro Campus Canteen Management:**
   * **Role:** The project sponsor and primary decision-maker for the system's implementation.
   * **Responsibilities:** Approving budgets, providing operational insights, and monitoring progress.
   * **Expectations:** A reliable, cost-effective system that streamlines operations and enhances service quality.
2. **Customers (Students and Staff):**
   * **Role:** End users of the system who interact with features like order placement, payment, and feedback.
   * **Responsibilities:** Provide feedback to improve the system and use it for daily transactions.
   * **Expectations:** User-friendly interface, quick service, and multiple payment options.

**Secondary Stakeholders**

1. **FuturistIT Development Team:**
   * **Role:** System developers responsible for designing, developing, and deploying the software.
   * **Responsibilities:** Gathering requirements, coding, testing, and providing post-deployment support.
   * **Expectations:** Clear requirements and timely approvals from the canteen management.
2. **Canteen Staff:**
   * **Role:** System operators who use the backend for order tracking and inventory management.
   * **Responsibilities:** Manage daily operations, update inventory, and process orders.
   * **Expectations:** Intuitive backend functionalities to reduce workload and minimize errors.
3. **IT Support Team (Post-Deployment):**
   * **Role:** Responsible for maintaining the system after deployment.
   * **Responsibilities:** Fixing bugs, implementing updates, and ensuring system uptime.
   * **Expectations:** Clear documentation and smooth transition from the development team.

**7.0 Feasibility Study**

A feasibility study evaluates whether the development and deployment of the Canteen Management System for Retro Campus Canteen are viable. This involves analyzing both technical and financial feasibility to ensure the project can meet its goals within constraints.

**7.1 Technical Feasibility**

**Assessment of Technologies and Resources:**

1. **Operating System Compatibility:**
   * The system will support Windows, Linux, and macOS for servers and Android/iOS for mobile applications, ensuring cross-platform usability.
2. **Development Frameworks:**
   * Backend: Laravel, a reliable PHP framework known for its scalability and robustness.
   * Frontend: React for web applications and Flutter for mobile applications, enabling responsive and visually appealing user interfaces.
3. **Database Management:**
   * MySQL or PostgreSQL will be used for managing data securely and efficiently, providing robust storage for user orders, inventory, and reports.
4. **Hardware Requirements:**
   * Server with at least 8 GB RAM, 1 TB storage, and a dual-core processor to ensure smooth operations.
   * Devices for development and testing, such as desktop computers and smartphones.
5. **Technical Challenges and Solutions:**
   * **Challenge:** Ensuring real-time inventory updates and order tracking.
     + **Solution:** Use WebSocket or similar technologies for real-time communication.
   * **Challenge:** Integration of multiple payment gateways.
     + **Solution:** Use APIs provided by payment gateway providers for seamless transactions.
6. **Availability of Skilled Personnel:**
   * A skilled development team, including project managers, developers, designers, and QA testers, is available to execute the project.

**Conclusion:**  
The proposed system is technically feasible, given the availability of modern frameworks, database technologies, and skilled resources. The use of scalable and tested technologies reduces the risk of technical failure.

**7.2 Financial Feasibility**

**7.2.1 Development Cost Breakdown:**

1. Requirement Gathering and Analysis: BDT 30,000
2. UI/UX Design: BDT 42,500
3. Backend Development: BDT 100,500
4. Frontend Development (Web and Mobile): BDT 99,000
5. Database Design and Integration: BDT 44,500
6. Testing and Quality Assurance: BDT 33,000
7. Deployment and Initial Support: BDT 16,500

**Total Development Cost: BDT 365,000**

**7.2.2 Maintenance Cost Breakdown (Per Year):**

1. Bug Fixes and Updates: BDT 24,750
2. Hosting and Server Costs: BDT 8,250
3. Technical Support: BDT 16,500

**Total Maintenance Cost: BDT 49,500**

**7.2.3 Financial Benefits:**

1. Increased Efficiency: Faster order processing and inventory management will reduce operational delays and labor costs.
2. Customer Retention: Improved customer satisfaction through streamlined processes will boost repeat business.
3. Data-Driven Decisions: Reporting and analytics will help management identify cost-saving opportunities.

**Conclusion:**  
The project is financially feasible with manageable development and maintenance costs. The expected benefits, such as improved efficiency and customer satisfaction, justify the investment.

Both technical and financial analyses indicate that the project is feasible. The chosen technologies ensure scalability and reliability, while the financial projections demonstrate that the system can achieve a favorable return on investment within a reasonable timeframe.

**8.0 System Components**

The Canteen Management System for Retro Campus Canteen is composed of multiple interdependent components, each designed to handle specific functionalities. These components work together to deliver a seamless experience for users, efficient management for staff, and valuable insights for administrators.

**8.1 Order Management Component**

* Handles customer order placement and staff order tracking.
* Key Features:
  + Digital order placement through web and mobile platforms.
  + Real-time order status updates for customers and staff.

**8.2 Billing and Payment Component**

* Manages billing and payment processing for all transactions.
* Key Features:
  + Automatic bill generation based on orders.
  + Integration with multiple payment gateways (cash, card, digital wallets).

**8.3 Inventory Management Component**

* Tracks stock levels and updates inventory in real time.
* Key Features:
  + Alerts for low inventory levels.
  + Stock usage tracking for better procurement planning.

**8.4 Reporting and Analytics Component**

* Provides insights into sales, inventory, and operational performance.
* Key Features:
  + Sales and revenue reports.
  + Inventory usage and waste analysis.
  + Customer feedback trends and operational efficiency metrics.

**8.5 Customer Feedback Component**

* Collects and manages customer reviews and ratings.
* Key Features:
  + Rating system for service and food quality.
  + Feedback collection for identifying areas of improvement.

**8.6 Authentication and Authorization Component**

* Ensures secure access to the system for different user roles.
* Key Features:
  + Role-based access control for customers, staff, and administrators.
  + Secure login via encryption protocols.

**8.7 Notification and Alert Component**

* Sends notifications and alerts to users and staff.
* Key Features:
  + Order status updates for customers.
  + Inventory alerts for staff.

**8.8 Database Component**

* Central repository for storing and managing all system data.
* Key Features:
  + Stores order details, inventory data, user information, and feedback.
  + Ensures data integrity and availability with MySQL or PostgreSQL.

**8.9 Integration Component**

* Facilitates interaction between system modules and third-party services.
* Key Features:
  + Payment gateway integration.
  + API connections for future enhancements (e.g., external inventory systems).

**Summary of Components**

The Canteen Management System is modular, ensuring scalability, ease of maintenance, and efficient operations. Each component plays a distinct role, and together they create a cohesive and reliable system for the Retro Campus Canteen.

**9.0 Process Model to Be Followed**

**9.1 Selected Process Model:** Agile Development Model

The Agile Development Model will be used for the development of the Canteen Management System. This model is ideal for projects like this due to its flexibility, iterative nature, and ability to incorporate feedback from stakeholders throughout the development lifecycle.

**9.2 Steps of the Agile Process Model**

**1. Concept and Inception**

* Activities:
  + Identify project goals, objectives, and constraints.
  + Develop an initial project plan and resource allocation.
* Output:
  + Project charter and high-level requirements.

**2. Iteration Planning**

* Activities:
  + Break down requirements into smaller, manageable tasks (user stories).
  + Prioritize tasks based on business value and technical feasibility.
  + Create a sprint backlog for the first iteration.
* Output:
  + Sprint backlog with clear deliverables.

**3. Iterative Development (Multiple Sprints)**

* Activities in Each Sprint:
  1. Sprint Planning: Set goals and assign tasks.
  2. Development: Code and implement features.
  3. Testing: Perform unit, integration, and functional testing.
  4. Review and Feedback: Demonstrate the sprint output to stakeholders and collect feedback.
  5. Retrospective: Identify what went well and what needs improvement for the next sprint.
* Duration: Each sprint lasts 2–4 weeks.
* Output (Per Sprint):
  1. Functional and tested features (e.g., order management, payment processing).

**4. Integration and Final Testing**

* Activities:
  + Integrate all sprint deliverables into a complete system.
  + Conduct end-to-end testing to ensure all components work seamlessly.
  + Fix any identified issues or bugs.
* Output:
  + Fully functional and integrated system ready for deployment.

**5. Deployment**

* Activities:
  + Deploy the system on production servers.
  + Ensure the system is accessible to all intended users (web and mobile platforms).
* Output:
  + Live system available for Retro Campus Canteen.

**6. Post-Deployment Support and Maintenance**

* Activities:
  + Provide technical support and resolve any operational issues.
  + Implement updates and new features based on feedback.
* Output:
  + Stable and up-to-date system.

**9.3 Justification for Using Agile**

1. Flexibility: Agile allows changes to be incorporated based on stakeholder feedback, ensuring the system meets evolving requirements.
2. Stakeholder Collaboration: Regular sprint reviews keep stakeholders involved and informed throughout development.
3. Iterative Delivery: Enables delivery of functional features early in the process, allowing stakeholders to see progress and provide feedback.
4. Risk Mitigation: Continuous testing and feedback reduce the risk of major flaws in the final product.
5. Efficiency: Agile promotes efficient use of resources by focusing on high-priority tasks first.

**10.0 Effort Estimation**

Effort estimation for developing the Canteen Management System is derived from the financial breakdown. This involves analyzing the cost of individual tasks and mapping them to the required workforce, considering the timeline and complexity of each task. Here is a Work Breakdown Structure (WBS):

**10.1 Work Breakdown Structure (WBS)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Cost (BDT)** | **Effort (Hours)** | **Resources Needed** |
| 1. Requirement Gathering and Analysis | 30,000 | 175 | 1 Business Analyst, 1 PM |
| 2. UI/UX Design | 42,500 | 245 | 1 UI/UX Designer |
| 3. Backend Development | 100,500 | 365 | 2 Backend Developers |
| 4. Frontend Development (Web & Mobile) | 99,000 | 285 | 2 Frontend Developers |
| 5. Database Design and Integration | 44,500 | 160 | 1 Database Engineer |
| 6. Testing and Quality Assurance | 33,000 | 245 | 1 QA Tester |
| 7. Deployment and Initial Support | 16,500 | 40 | 1 DevOps Engineer |

**10.2 Role Allocation and Effort**

1. **Project Manager (PM):**
   * Effort Allocation: 70 hours
   * Role: Oversees the project, coordinates team efforts, and ensures project goals are met.
2. **Business Analyst:**
   * Effort Allocation: 105 hours
   * Role: Gathers and documents requirements, acting as a liaison between stakeholders and the development team.
3. **UI/UX Designer:**
   * Effort Allocation: 245 hours
   * Role: Designs intuitive interfaces for web and mobile platforms.
4. **Backend Developers (2):**
   * Effort Allocation: 365 hours (shared)
   * Role: Develop server-side logic, APIs, and integrations.
5. **Frontend Developers (2):**
   * Effort Allocation: 285 hours (shared)
   * Role: Build web and mobile user interfaces.
6. **Database Engineer:**
   * Effort Allocation: 160 hours
   * Role: Designs and integrates the database structure.
7. **QA Tester:**
   * Effort Allocation: 245 hours
   * Role: Conducts system testing to identify and resolve issues.
8. **DevOps Engineer:**
   * Effort Allocation: 40 hours
   * Role: Deploys the system and ensures smooth operation in the production environment.

**10.3 Team Size Estimation**

Based on the tasks and their effort estimates, the following team composition is suggested:

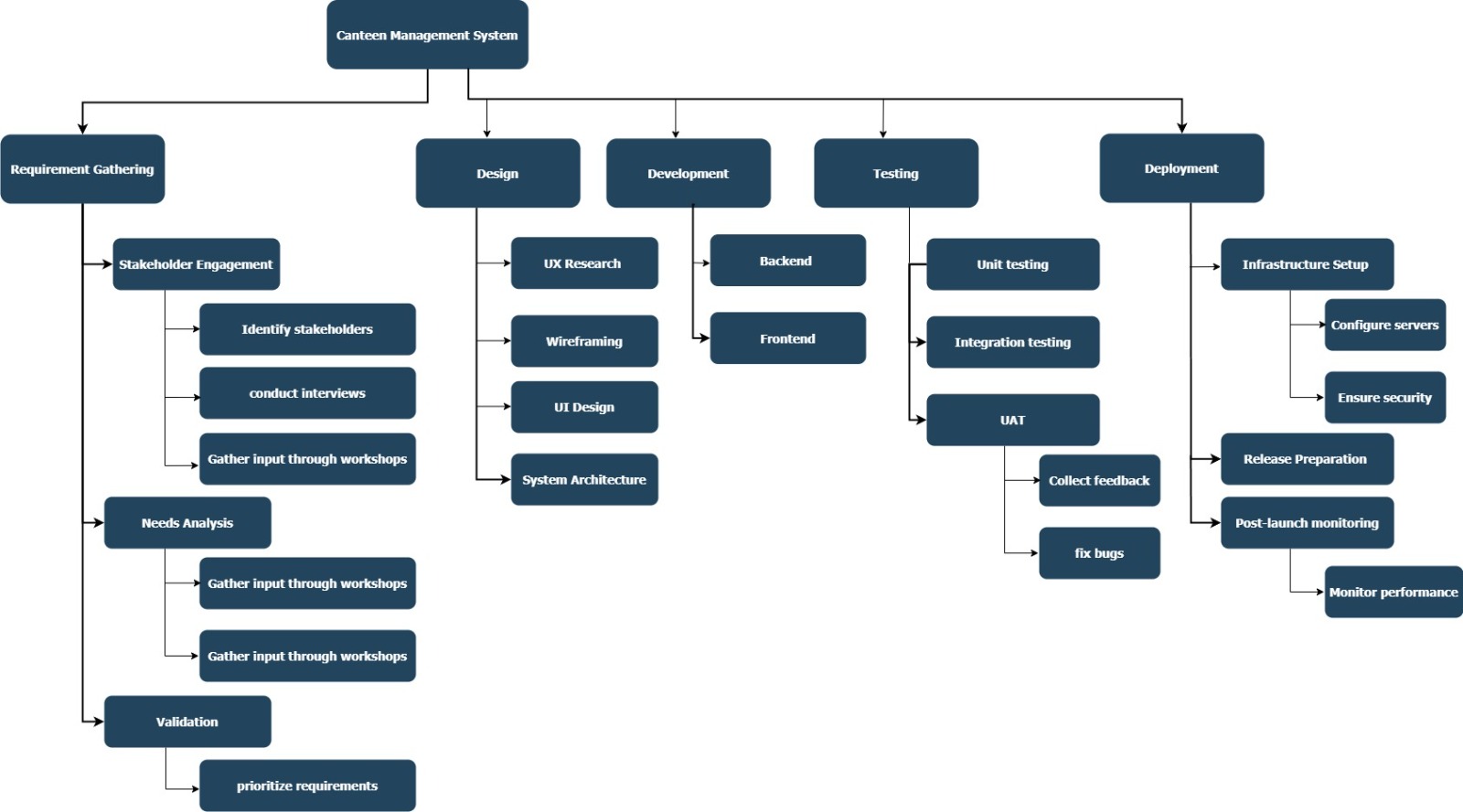
* 1 Project Manager
* 1 Business Analyst
* 1 UI/UX Designer
* 2 Backend Developers
* 2 Frontend Developers
* 1 Database Engineer
* 1 QA Tester
* 1 DevOps Engineer

**Total Team Members: 10**

**10.4 Justification of Efforts**

1. Aligned with Financial Feasibility: The total cost aligns with the development budget of BDT 365,000.
2. Balanced Resource Allocation: Efforts are distributed evenly across team roles, ensuring timely delivery without overburdening any single resource.
3. Realistic Timeline: With a 40-hour workweek, the project can be completed in approximately 12–14 weeks, including development, testing, and deployment.

**10.5 Activity Based WBS**



**10.6 Product Based WBS**

**1. Requirement Specification WBS**

A diagram of a company

Description automatically generated

**2. System Design WBS**

A diagram of a diagram

Description automatically generated

**3. Development WBS**

A diagram of a company

Description automatically generated

**4. Testing WBS**

A diagram of a company

Description automatically generated

**5. Deployment WBS**

A diagram of a company

Description automatically generated

**11.0 Activity Network Diagram**

**11.1. Identify Activities and Tasks**

For the Canteen Management System, here’s an example list of activities:

1. Requirement Gathering
2. System Design
3. UI/UX Prototyping
4. Database Design
5. Backend Development
6. Frontend Development
7. Integration Testing
8. System Testing
9. User Acceptance Testing (UAT)
10. Deployment

**11.2. Define Dependencies Between Activities**

Determine which activities depend on the completion of others. Here are the dependencies:

* Requirement Gathering → Precedes System Design
* System Design → Precedes UI/UX Prototyping, Database Design
* UI/UX Prototyping → Precedes Frontend Development
* Database Design → Precedes Backend Development
* Frontend Development and Backend Development → Precede Integration Testing
* Integration Testing → Precedes System Testing
* System Testing → Precedes UAT
* UAT → Precedes Deployment
* Deployment → Finish

**11.3. Arrange Activities in Sequence**

Using the dependencies, organize activities in chronological order.

Example:

1. Start with Requirement Gathering (no predecessor).
2. Draw an arrow to System Design, showing dependency.
3. From System Design, branch out arrows to UI/UX Prototyping and Database Design.
4. Similarly, connect all dependent activities.

**11.4. Assign Durations**

Estimate the time required to complete each activity. Example:

* Requirement Gathering: 5 weeks
* System Design: 8 weeks
* UI/UX Prototyping: 7 weeks
* Database Design: 6 weeks
* Backend Development: 15 weeks
* Frontend Development: 12 weeks
* Integration Testing: 5 weeks
* System Testing: 7 weeks
* UAT: 4 weeks
* Deployment: 3 weeks

**11.5. Use Nodes for Activities and Arrows for Dependencies**

* Each node represents an activity (labeled with its name and duration).
* Each arrow represents a dependency (connect the nodes according to their precedence).

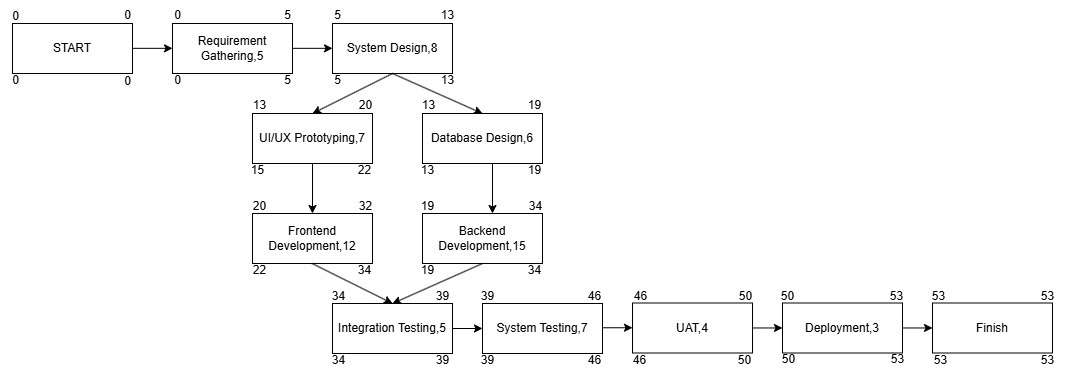


Fig: Activity Network Diagram for Retro Canteen Management System

**12.0 Risk Analysis**

The following table presents a comprehensive list of possible risks in the project planning phase of the Canteen Management System, categorized under different risk management strategies. Each risk is assessed based on its title, potential impact, likelihood, and mitigation strategy.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risk Title** | **Category** | **Impact** | **Likelihood** | **Mitigation Strategy** |
| System Downtime for Maintenance | Risk Acceptance | Low | Low | Schedule maintenance during off-peak hours to minimize user disruption. |
| Complex Feature Implementation | Risk Avoidance | High | Medium | Simplify the initial scope and focus on delivering core functionalities. |
| Inadequate User Training | Risk Reduction | Medium | High | Provide user manuals, conduct training sessions, and create a FAQ section. |
| Testing Failures | Risk Reduction | High | Medium | Conduct regular unit, integration, and system testing during the development phase. |
| Payment Gateway Integration Issues | Risk Transfer | High | Low | Partner with a reliable third-party payment gateway provider with clear SLAs. |
| Data Breach Risks | Risk Mitigation | High | Medium | Implement encryption, secure authentication mechanisms, and regular security audits. |
| Scope Creep | Risk Mitigation | High | High | Establish a clear scope and implement a change management process. |
| Delayed System Deployment | Risk Mitigation | High | Medium | Create a realistic project schedule with buffer time for unexpected delays. |
| Lack of Stakeholder Communication | Risk Reduction | High | High | Conduct regular meetings and progress updates with stakeholders. |
| Resource Unavailability | Risk Mitigation | Medium | Medium | Maintain a backup plan for resources and allocate them effectively. |
| Poor System Performance | Risk Reduction | High | Low | Optimize code and conduct thorough performance testing before deployment. |
| Insufficient Budget | Risk Mitigation | High | Medium | Monitor expenditures and prioritize features to avoid overspending. |
| Integration Challenges | Risk Mitigation | High | Medium | Develop an integration plan and test external system connections early. |
| Unclear Requirements | Risk Avoidance | High | High | Conduct detailed requirements elicitation and validation early in the project. |
| Resistance to Change by Users | Risk Reduction | Medium | High | Engage users early in the process, gather feedback, and emphasize system benefits. |
| Legal or Compliance Issues | Risk Transfer | High | Low | Consult with legal and compliance experts to ensure adherence to all regulations. |

**12.1 Key Points for Risk Management**

1. Risk Acceptance: Accept risks with low likelihood and low impact that do not require active mitigation.
2. Risk Avoidance: Eliminate risks entirely by changing the project approach or avoiding certain activities.
3. Risk Reduction: Minimize the likelihood or impact of risks through proactive measures like regular testing and communication.
4. Risk Transfer: Shift responsibility for certain risks to a third party, such as outsourcing or using external services.
5. Risk Mitigation: Create contingency plans and strategies to handle risks if they occur.

This risk analysis helps ensure proper project planning, minimizes potential threats, and improves the chances of successful project execution.

**12.2. Risk Assessment using Risk Reduction Leverage (RRL)**

Risk Reduction Leverage (RRL) is calculated as follows:

**RRL= ​**

Where:

* Risk Exposure (RE) = Impact × Likelihood
* Impact is the potential severity of the risk.
* Likelihood is the probability of the risk occurring.

**Risks with RRL:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Risk Title** | **Impact (I)** | **Likelihood (L)** | **RE Before Mitigation (I × L)** | **RE After Mitigation** | **Mitigation Cost** | **RRL** | **Worthwhile?** |
| System Downtime | High (5) | Medium (0.3) | 1.5 | 0.6 | 0.2 | 4.5 | Yes |
| Data Breach | High (5) | Medium (0.4) | 2.0 | 0.8 | 0.5 | 2.4 | Yes |
| Scope Creep | Medium (3) | High (0.6) | 1.8 | 0.6 | 0.4 | 3.0 | Yes |
| Delayed Deployment | High (4) | Medium (0.3) | 1.2 | 0.5 | 0.3 | 2.3 | Yes |

All risks with a positive RRL (>1) demonstrate that mitigation is worthwhile, making the project feasible and worth pursuing.

**12.3. PERT Analysis to Evaluate Effects of Uncertainty**

The Program Evaluation and Review Technique (PERT) uses three time estimates to evaluate project uncertainty:

* Optimistic Time (O): The best-case scenario.
* Most Likely Time (M): The most realistic estimate.
* Pessimistic Time (P): The worst-case scenario.

**Formula for Expected Time (TE):**

**TE =**

**Steps for PERT Analysis:**

1. List Project Activities: Identify key tasks and their dependencies.
2. Estimate Time for Each Activity:
   * Example: Task A (Requirement Analysis)
     + Optimistic Time (O): 3 days
     + Most Likely Time (M): 5 days
     + Pessimistic Time (P): 7 days
3. Calculate TE for Each Activity:
   * TE = = 5.33
4. Compute the Variance for Each Activity:
   * Variance for Task A = = = = 0.444
5. Determine Critical Path:
   * Sum up the TE of critical activities to determine the total expected project completion time.
   * Compute total variance of the critical path to assess project uncertainty.
6. Calculate Probability of Meeting Deadlines: Use the standard deviation of the project duration to calculate probabilities:
   * Z =
   * Use a Z-score table to find the probability.

**PERT Table for Key Activities:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activity** | **O** | **M** | **P** | **TE** | **Variance** |
| Requirement Analysis | 3 | 5 | 7 | 5.33 | 0.444 |
| Design | 5 | 8 | 11 | 8.00 | 1.000 |
| Development | 10 | 15 | 20 | 15.00 | 2.778 |
| Testing | 3 | 4 | 6 | 4.17 | 0.250 |
| Deployment | 2 | 3 | 5 | 3.17 | 0.250 |

Using RRL demonstrates the worthiness of risk mitigation in the project. Meanwhile, PERT provides a structured method for evaluating the effects of uncertainty, helping in better planning and realistic deadline setting for the Canteen Management System project.

**12.4. Risk Due to Product Size**

* **Description:** Large product size introduces complexity in design, development, and testing. Managing dependencies between modules and ensuring integration becomes challenging.
* **Key Risks:** 
  + Increased development time and cost due to managing large-scale features.
  + Greater likelihood of defects due to the complexity of the system.
* **Mitigation Strategies:** 
  + Conduct regular code reviews and integration testing to identify issues early.
  + Adopt Agile methodologies for incremental development.

**12.5. Risk Due to Business Impact**

* **Description:** The system's failure or underperformance may have significant consequences for stakeholders, such as financial loss, reputational damage in canteen operations.
* **Key Risks:** 
  + Failure to meet user expectations, leading to reduced adoption.
  + Financial losses due to operational downtime or inefficiency.
* **Mitigation Strategies:** 
  + Conduct a thorough requirements elicitation to align with business goals.
  + Perform rigorous testing to ensure system reliability and robustness.
  + Create fallback procedures to minimize operational downtime.

**12.6. Risks Due to the Customer**

* **Description:** Challenges arise from unclear or evolving customer requirements, delays in feedback, or lack of stakeholder engagement.
* **Key Risks:** 
  + Misaligned expectations due to unclear requirements.
  + Delays in project delivery are due to insufficient feedback.
* **Mitigation Strategies:** 
  + Establish clear and detailed requirements documentation.
  + Maintain regular communication with stakeholders through meetings.

**12.7. Technology Risks**

* **Description:** The use of new or unfamiliar technologies, tools, or platforms can lead to technical challenges and delays.
* **Key Risks:** 
  + Compatibility issues between different technologies used in the system.
  + Lack of expertise in the chosen technology stack.
  + High cost or obsolescence of technology during the project lifecycle.
* **Mitigation Strategies:** 
  + Select a technology stack that aligns with the team's expertise and project requirements.
  + Provide training to the development team on new technologies.

**13.0 Estimation of Resources and Budget**

**13.1. Resource Requirements for the Project**

The project requires various resources, including human, technical, and infrastructure, for successful implementation. Below is the detailed estimation:

|  |  |  |
| --- | --- | --- |
| **Resource Type** | **Description** | **Quantity** |
| **Human Resources** | | |
| Project Manager | Responsible for project planning, scheduling, and execution. | 1 |
| Business Analyst | Gathers and documents requirements from stakeholders. | 1 |
| Software Developers | Develop and implement the canteen management system. | 3 |
| UI/UX Designer | Designs user-friendly interfaces for the system. | 1 |
| Quality Assurance (QA) | Test the system for bugs and quality standards. | 2 |
| DevOps Engineer | Handles deployment and system maintenance. | 1 |
| **Technical Resources** | | |
| Development Tools | IDEs, version control systems, etc. | VS Code, GitHub, Docker |
| Server Infrastructure | Hosting and data storage. | Cloud-based (AWS/Azure) |
| Software Licenses | Libraries, frameworks, and platforms. | As required |
| Testing Tools | Automated testing and bug tracking tools. | Selenium |
| **Physical Resources** | | |
| Office Space | For team collaboration and work. | Shared workspace |
| Hardware | Computers, testing devices, etc. | Laptops (6), Test Devices |

**13.2. Resource Requirements for the Clients**

For clients (staff, managers, and customers) to use the system, the following resources are required:

|  |  |  |
| --- | --- | --- |
| **Resource Type** | **Description** | **Quantity** |
| **Hardware** | | |
| Point-of-Sale (POS) Devices | Devices for billing and order processing. | 3–5 devices per canteen |
| Tablets/Smartphones | Devices for staff to manage orders. | 1–2 devices per canteen |
| Printers | For printing receipts or reports. | 1 per canteen |
| **Software** | | |
| System Access | User accounts and roles for different users. | Client requirements |
| Internet Connection | Required for accessing cloud-based system | Stable connection |
| **Training** | | |
| User Training | Training for staff to operate the system. | Training sessions |

**13.3. Budget for the Project**

The project budget includes human resources, technical resources, and operational expenses, optimized for profitability.

|  |  |  |
| --- | --- | --- |
| **Category** | **Description** | **Estimated Cost** |
| **Human Resources** | | |
| Project Manager | BDT 3,000/week × 5 weeks | 15,000 |
| Business Analyst | BDT 2,500/week × 5 weeks | 12,500 |
| Software Developers | BDT 4,000/week × 2 devs × 15 weeks | 120,000 |
| UI/UX Designer | BDT 3,500/week × 7 weeks | 24,500 |
| QA Engineers | BDT 3,000/week × 1 engineer × 7 weeks | 21,000 |
| DevOps Engineer | BDT 4,000/week × 3 weeks | 12,000 |
| **Technical Resources** | | |
| Development Tools | Subscriptions and licenses. | 10,000 |
| Server Infrastructure | Cloud hosting and storage. | 6,000 |
| Testing Tools | Subscriptions for testing frameworks | 8,000 |
| **Physical Resources** | | |
| Office Space | Rent for workspace during the project (5 months). | 45,000 |
| **Training Costs** | | |
| Client Training | Sessions for system users. | 5,000 |
| Contingency | 10% of the total cost for unforeseen expenses. | 26,000 |
| **Total Estimated Budget** |  | **295,000** |

**13.4 Company Maintenance Cost Breakdown (Per Year)**

|  |  |  |
| --- | --- | --- |
| **Category** | **Description** | **Estimated Cost** |
| Bug Fixes and Updates | Ongoing bug fixes and regular updates | 10,000 |
| Hosting and Server Costs | Monthly cloud hosting, storage, and related services | 5,000 |
| Technical Support | Maintenance support (calls, chat, etc.) | 6,000 |
| Other Operational Costs | Miscellaneous maintenance costs | 4,000 |
| Total Maintenance Cost | Total Maintenance Cost (Per Year) | 25,000 |

**13.5 Conclusion:**

1. **Client Budget**
   * Development Cost: BDT 365,000
   * Maintenance Cost (Per Year): BDT 49,500
   * Total Client Budget: BDT 414,500
2. **Company Actual Cost**
   * Development Cost: BDT 295,000
   * Maintenance Cost (Per Year): BDT 25,000
   * Total Company Cost: BDT 320,000

**14.0 Conclusion**

The Canteen Management System is a comprehensive solution aimed at modernizing and streamlining the operations of canteens. This document has meticulously outlined the project's scope, objectives, feasibility, risks, resource requirements, and implementation strategies to ensure the project's success. The system is designed to address the operational inefficiencies faced by canteen management, enhance customer experience, and facilitate seamless integration of processes such as order placement, billing, inventory management, and reporting. The use of modern technologies, such as cloud-based architecture, ensures scalability, reliability, and ease of access for both the staff and customers.

By following a structured development approach, as highlighted in the process model and work breakdown structures, the project team will efficiently allocate resources and meet the project goals within the estimated budget. Additionally, detailed risk analysis and mitigation strategies will help ensure the project's resilience and adaptability to unforeseen challenges. The completion of this system is expected to significantly benefit canteen staff, management, and customers by reducing manual workload, enhancing accuracy, and providing a convenient platform for operations. The project's feasibility in terms of both technical and financial aspects, combined with the robust planning and estimations provided, demonstrates that this is a viable and worthwhile endeavor.

In conclusion, the Canteen Management System promises to be a transformational tool for canteen operations, delivering value to all stakeholders and setting a benchmark for efficient management systems in the food service industry.