Team: Shubham Potadar, Sri devi Naik, Sudhesh, Supriya S Chavan

Stock Maintenance System

Problem Statement

The problem statement for a Stock Maintenance System involves designing and implementing a software solution that can manage and track inventory levels of a business or organization. The system should be able to keep track of the stock levels, monitor stock movements (incoming and outgoing), generate reports, and provide alerts for low stock levels.

Some specific requirements of a Stock Maintenance System could include:

Accurately tracking the inventory levels of different products or items in real-time

Allowing users to update inventory levels as new stock is received or sold

Providing alerts when the stock level of a particular item falls below a specified threshold

Generating reports on inventory levels, stock movements, and other relevant data

Allowing multiple users to access and use the system simultaneously with appropriate levels of access control and security

Integrating with other systems such as point of sale (POS) systems, ordering systems, and accounting software to ensure that stock levels are accurately reflected across all systems

Providing a user-friendly interface that is easy to use and navigate

The overall objective of a Stock Maintenance System is to improve the efficiency and accuracy of inventory management and reduce the risk of stockouts or overstocking, which can lead to lost sales or increased costs. The system can be implemented in various industries such as retail, manufacturing, and distribution, where inventory management is critical to business operations.

Software Requirement Specification(SRS)

1 Introduction

1.1 Purpose of this document

The purpose of this document is to define the software requirement specification for the development of a Stock Maintenance System application that will be used to automate the day-to-day transactions in the stock market. This document will outline the functional and non-functional requirements, design constraints, and user interface specifications for the system. This document serves as the reference for the development team to ensure that the system meets the requirements of the clients and as well as for the stakeholders.

1.2 Scope of this document

The scope of this document for Stock Maintenance System is to provide a comprehensive overview of the software solution's objectives, requirements, and design. The document defines the boundaries of the project, outlining what is included and excluded from the scope of the system.

The scope of the Stock Maintenance System includes:

Inventory Management: The system should be able to manage and track inventory levels of different products or items in real-time.

Stock Monitoring: The system should allow users to update inventory levels as new stock is received or sold, providing alerts when the stock level of a particular item falls below a specified threshold.

Reporting: The system should generate reports on inventory levels, stock movements, and other relevant data.

Security and Access Control: The system should allow multiple users to access and use the system simultaneously with appropriate levels of access control and security.

Integration with other systems: The system should integrate with other systems such as point of sale (POS) systems, ordering systems, and accounting software to ensure that stock levels are accurately reflected across all systems.

1.3 Overview

Stock Maintenance System is a software solution designed to manage and track inventory levels of a business or organization. The system provides real-time information about stock levels, stock movements, and generates reports for analysis and decision-making. The system's main objective is to ensure that the right quantity of products or items is available at the right time, while minimizing excess inventory levels and the associated costs.

2 General Description

The Stock Maintenance System allows users to monitor stock levels, update inventory as new stock is received or sold, and set alerts for low stock levels. It also provides a user-friendly interface that allows multiple users to access and use the system simultaneously with appropriate levels of access control and security.

In addition to these features, a Stock Maintenance System can also integrate with other systems such as point of sale (POS) systems, ordering systems, and accounting software to ensure that inventory levels are accurately reflected across all systems. This integration ensures that the business can maintain a centralized view of inventory levels and make informed decisions based on accurate data.

The system generates reports on inventory levels, stock movements, and other relevant data, providing insights to users on stock performance and helping them to make informed decisions. The system also helps to prevent stockouts, which can lead to lost sales, and minimizes excess inventory levels, which can result in increased storage costs and wastage.

Overall, the Stock Maintenance System is an essential tool for businesses to manage their inventory levels more efficiently, improve operational efficiency, and reduce costs. The system provides real-time information, generates reports, and ensures accurate inventory levels, enabling businesses to make informed decisions and stay competitive in their respective industries

3 Functional Requirements:

- Inventory Management: The system should allow users to manage inventory levels and track stock movements in real-time. This includes updating stock levels as new stock is received or sold, setting alerts for low stock levels, and generating reports on stock levels and movements.

- Product Management: The system should allow users to manage products, including adding, editing, and deleting products, as well as assigning product codes and descriptions.
- Order Management: The system should allow users to manage orders, including tracking orders, generating invoices, and updating stock levels based on orders received.
- Supplier Management: The system should allow users to manage suppliers, including adding, editing, and deleting suppliers, as well as tracking supplier information such as contact details and pricing.
- User Management: The system should allow users to manage user accounts and access levels, including assigning appropriate levels of access control and security.
- Reporting: The system should generate reports on inventory levels, stock movements, sales, and other relevant data to help users make informed decisions.
- Integration: The system should integrate with other systems such as POS systems, ordering systems, and accounting software to ensure that inventory levels are accurately reflected across all systems.
- Alerts: The system should send alerts when stock levels fall below a certain threshold to prevent stockouts and ensure timely reordering.
- Barcode Scanning: The system should have the ability to scan barcodes to quickly and accurately update inventory levels and track stock movements.
- Forecasting: The system should have the ability to forecast future inventory levels based on historical data and trends to help businesses make informed decisions about inventory management.

4 Interface Requirements

- User-Friendly Interface: The system should have a user-friendly interface that is easy to navigate and use.
- Customization: The system should allow users to customize the interface to meet their specific needs.
- Accessibility: The system should be accessible from multiple devices and platforms, including desktops, laptops, tablets, and smartphones.

- Security: The system should have appropriate levels of access control and security to prevent unauthorized access and ensure data privacy.
- Search Functionality: The system should have a search functionality that allows users to quickly search for products, orders, and other relevant information.
- Notifications: The system should send notifications to users when stock levels fall below a certain threshold or when orders are received.
- Dashboards: The system should have a dashboard that provides an overview of inventory levels, sales, and other relevant data.
- Reports: The system should generate reports on inventory levels, stock movements, sales, and other relevant data in various formats, including PDF, Excel, and CSV.
- Barcode Scanning: The system should have a barcode scanning functionality to quickly and accurately update inventory levels and track stock movements.
- Integration: The system should integrate with other systems such as POS systems, ordering systems, and accounting software to ensure that inventory levels are accurately reflected across all systems.

5 Performance Requirements

- The system should have a response time of less than 2 seconds for most functions.
- The system should have a high level of availability and reliability, with a minimum uptime of 99.9%.
- The system should be scalable and able to handle increases in traffic and demand.
- The system should have adequate security measures to protect customer data and prevent unauthorized access.
- The system should have a backup and recovery mechanism to ensure business continuity in case of system failure or disaster.
- The system should comply with relevant industry standards and regulations for data privacy, security, and accessibility.
- The system should be able to handle a large volume of requests and transactions without performance degradation.
- The system must be able to respond quickly to user requests, especially those related to stock updates and inventory queries.
- The system should be available for use as much as possible, with minimal downtime or outages
- The system should be designed to use resources efficiently, including CPU, memory, and network bandwidth.

6 Design Constraints

- The system should have a user-friendly and intuitive interface.
- The system should follow industry-standard design principles and guidelines.
- The system should be responsive and adaptable to different screen sizes and resolutions.
- The system should follow a three-tier architecture with a presentation layer, business logic layer, and data access layer.
- The system should be modular and extensible to allow for future enhancements and modifications.
- The system should be designed to handle high traffic loads and large amounts of data.
- The system should use caching mechanisms to improve performance.
- The system should have a secure architecture that prevents unauthorized access and data breaches.
- The system should use encryption mechanisms to protect sensitive data.
- The system should have a backup and recovery plan in place to prevent data loss in case of security breaches.

7 Non-Functional Attributes

- **Reliability:** The system should be reliable, meaning it can consistently perform its functions without failure or error, and can recover from any failures that do occur in a timely manner.
- **Maintainability:** The system should be designed for ease of maintenance, such as through modular code, well-defined interfaces, and clear documentation, to reduce the time and effort required to maintain and enhance the system.
- **Scalability:** The system should be scalable, meaning it can accommodate growth in stock volume or changes in the number of users or transactions, without requiring significant changes or performance degradation.
- **Availability:** The system should be available, meaning it can be accessed by users when needed, with minimal downtime or interruptions, and can recover from any disruptions that do occur.
- **Performance:** The system should have acceptable performance, meaning it can respond to user requests in a timely manner, without significant delays or slowdowns.
- **Security:** The system should be secure, meaning it can protect sensitive stock data and transactions from unauthorized access, manipulation, or disclosure, and comply with relevant security standards.

- **Usability:** The system should be usable, meaning it can be used easily and effectively by users with different levels of experience or technical proficiency, with a clear and consistent interface that enables efficient and error-free stock maintenance tasks.
- **Compatibility:** The system should be compatible with other systems, tools, or platforms that may be used by the organization, to ensure seamless integration and interoperability across different systems.

8 Preliminary Schedule and Budget

The development of the Stock Maintenance System is estimated to take eight months. The project will include design, development, testing, and deployment phases.
Assuming a team of 5 developers, a project duration of 6 months, and an hourly rate of Rs. 500 per developer, the cost for the development team would be Rs.3,450,000.
Testing and QA costs: Assuming a testing budget of 10% of the total development cost, this would be an additional Rs. 3,450,00.

Project management and documentation costs: Assuming a project management and documentation budget of 15% of the total development cost, this would be an additional Rs.517,500.

Adding all these costs together, the total preliminary budget for the Stock Maintenance System would be approximately Rs. 4,312,500.