**Preface**

**Version History:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Description** | **Author/Contributors** |
| 1.0 | 11Aug2023 | Initial SRS | Name1, Name2, Name3, name4 |

**Introduction**

The Warehouse Management System (WMS) is designed to streamline and optimize inventory management processes for general use. This document outlines the functional and non-functional requirements for the first iteration of the WMS. The system aims to efficiently manage inventory levels, process orders, and provide tools for auditing inventory. It will be a web-based application developed using .NET Core, C#, [MySQL], [Framework-Front End], and [Apache web server], technologies.

**Glossary**

WMS: Warehouse Management System

Inventory: Stock of goods and materials

Order: A request for items or products

Pick: The process of selecting items from inventory for an order

Ship: The process of dispatching orders to customers

**User Requirements Definition**

**Functional Requirements:**

1. Inventory Management:
   1. The system shall maintain accurate records of inventory items, including names, descriptions, quantities, and re-order points.
   2. Users with appropriate permissions shall be able to add, edit, or remove inventory items.
   3. The system shall generate alerts when inventory items fall below their re-order points.
2. Order Processing:
   1. Customers shall be able to submit new orders through the system.
   2. The system shall assign order numbers and statuses to each order (received, processing, and shipped).
   3. Users shall be able to view and manage order details, including items, quantities, and shipping information.
3. Picking and Shipping:
   1. The system shall create pick lists for orders in the "processing" status.
   2. Users shall be able to view and process pick lists, marking items as picked and ready for shipping.
   3. The system shall generate shipping labels and tracking numbers for orders marked as shipped.
4. Inventory Audits:
   1. The system shall provide tools for conducting cyclic inventory audits.
   2. Users shall be able to initiate and complete audits, reconciling physical counts with recorded inventory.

**Non-Functional Requirements:**

1. Performance:
   1. The system shall be capable of handling concurrent user interactions without significant performance degradation.
   2. Response times for critical functions (order processing, inventory updates) shall not exceed [time] seconds
2. Usability:
   1. The user interface shall be intuitive and user-friendly, requiring minimal training for end users.
   2. Error messages shall be clear and informative, aiding users in understanding and resolving issues.
3. Security:
   1. User authentication and authorization mechanisms shall be implemented to ensure data security.
   2. Personally identifiable information (PII) and payment information shall not be stored within the system.

**System Architecture**

The WMS will follow a three-tier architecture, consisting of a presentation layer, business logic layer, and data access layer. The presentation layer will be a web-based user interface developed using HTML5, CSS3, and JavaScript. The business logic layer will be implemented using .NET Core and C#, handling core functionalities such as order processing and inventory management. The data access layer will utilize a relational database to store inventory and order information.

**System Requirements Specification**

**Functional Requirements:**

1. User Authentication:
   1. The system shall provide secure user authentication using username and password credentials.
   2. Users shall have different levels of access based on their roles (admin, warehouse staff, and customer).
2. Order Status Tracking:
   1. The system shall update order statuses automatically based on the order processing workflow.
   2. Users shall be notified of status changes via email notifications.

**Non-Functional Requirements:**

1. Scalability:
   1. The system architecture shall be designed to accommodate potential growth in the number of users and inventory items.
2. Reliability:
   1. The system shall maintain data integrity and consistency during power outages or unexpected shutdowns.
3. Compatibility:
   1. The system shall be compatible with popular web browsers, including Chrome, Firefox, and Edge.

**System Models**

User Case, ect

**System Evolution**

The WMS is designed with flexibility in mind to accommodate future changes and enhancements. Potential hardware evolution, user feedback, and business needs will influence future system updates.

**Appendices**

1. Hardware Requirements
   1. Minimum and optimal hardware configurations for server and client devices.
2. Database Requirements
   1. Logical organization of data tables, relationships, and indexes.

**Index**

1. This is index 1