**System Analysis Phase – SDLC for Inventory Management System**

Goal: Gather, analyze, and document system requirements to ensure a clear understanding of what needs to be developed.

Outcome: A complete Software Requirements Specification (SRS) and visual representation of system processes.

**1. Conduct Requirement Analysis**

Purpose: Collect detailed information about the system's required features and constraints.

* Identify user needs through observation, interviews, or questionnaires with store owners and cashiers.
* Document functional requirements such as product management, billing, and reporting.
* Document non-functional requirements like performance, security, and usability.
* Specify hardware and software requirements for deployment.
* List all assumptions and dependencies (e.g., internet availability, single-branch system).
* Create a requirement traceability matrix (RTM) to link each requirement with system goals.

Deliverable: Requirement Analysis Report.

**2. Create Use Case Diagrams**

Purpose: Visually represent user interactions with the system.

* Identify all actors (Admin, Cashier, System).
* List key use cases: Login, Add Product, Generate Bill, Update Stock, View Reports, Logout.
* Define relationships between actors and use cases (include, extend).
* Draw use case diagrams using tools like Draw.io, Lucidchart, or StarUML.
* Add short descriptions for each use case in tabular form.

Deliverable: Use Case Diagram and Description Table.

**3. Define User Roles and Permissions**

Purpose: Identify roles and their access levels within the system.

* Define Admin role: Full system control (manage users, products, reports).
* Define Cashier role: Limited access (billing, stock updates, sales records).
* Define Viewer or Owner role: View sales summary and analytics.
* Map role-based access in a permissions table.

Deliverable: User Roles and Permissions Matrix.

**4. Identify System Inputs and Outputs**

Purpose: Define data the system will accept, process, and produce.

* List Inputs: Product details, quantity, sale entries, user credentials.
* List Outputs: Bills, stock alerts, sales reports, low-stock notifications.
* Create input/output flow diagram.
* Document data validation requirements (e.g., numeric input for quantity, unique product codes).

Deliverable: Input/Output Specifications Document.

**5. Create Data Flow Diagrams (DFDs)**

Purpose: Represent data movement within the system at different abstraction levels.

* Design Level 0 DFD (context diagram): Show interaction between external entities (Admin, Cashier) and system.
* Design Level 1 DFD: Show main processes such as 'Manage Inventory', 'Process Billing', 'Generate Reports'.
* Label all data stores and flows clearly.
* Use standard DFD symbols (process, data flow, data store, external entity).
* Review and validate DFDs with stakeholders for accuracy.

Deliverable: Level 0 and Level 1 Data Flow Diagrams.

**6. Write Software Requirements Specification (SRS)**

Purpose: Compile all analyzed data into a formal, structured document.

* Include Introduction: Purpose, Scope, and Definitions.
* Describe Overall Description: Product perspective, user classes, operating environment.
* Add Functional Requirements: Detailed descriptions of system features.
* Add Non-Functional Requirements: Performance, security, reliability, usability, scalability.
* Include External Interface Requirements: Hardware, software, and communication interfaces.
* Attach diagrams: Use Case, DFDs, and I/O specifications.
* Review and approve SRS with stakeholders.

Deliverable: Final SRS Document.

**Final Deliverables of System Analysis Phase**

|  |  |  |
| --- | --- | --- |
| # | Deliverable | Format |
| 1 | Requirement Analysis Report | Word/PDF |
| 2 | Use Case Diagram & Description | Image/Table |
| 3 | User Roles and Permissions Matrix | Table |
| 4 | Input/Output Specification | Word |
| 5 | Data Flow Diagrams (DFD Level 0 & 1) | Diagram File |