**SDLC Tasks for Inventory Management System**

**1. Planning Phase**

Goal: Define purpose, scope, and feasibility.

Tasks:

* Create Project Charter (project name, description, objectives)
* Define system goals (e.g., 'Automate billing and inventory updates')
* Identify stakeholders (store owner, cashier, admin)
* Write Scope Document (what’s included/excluded)
* Prepare Feasibility Report (tech stack, budget, timeline)
* Set up Git Repository & Project Board

Deliverables: Project Charter, Scope & Objectives Document, GitHub repo and project board initialized.

**2. System Analysis Phase**

Goal: Gather and document requirements.

* Conduct requirement analysis (functional + non-functional)
* Create Use Case Diagrams
* Define User Roles (Admin, Cashier, Store Owner)
* Identify System Inputs/Outputs
* Create Data Flow Diagram (DFD) — Level 0 & Level 1
* Write Software Requirements Specification (SRS)

Deliverables: Use Case Diagram, DFDs, SRS Document.

**3. System Design Phase**

Goal: Create architecture and UI plans.

* Design System Architecture Diagram
* Design Database Schema (tables: products, stock, sales, logs, users)
* Prepare ER Diagram
* Design Wireframes/UI mockups (Dashboard, Billing Screen, Stock Page)
* Define API Endpoints & Routes
* Write Design Document (SDD)

Deliverables: ER Diagram, Database Schema, UI Mockups, System Design Document (SDD).

**4. Development Phase**

Goal: Build the system in modules.

Backend Tasks:

* Set up server & folder structure
* Implement authentication (login/logout)
* Implement CRUD for products, inventory, and users
* Implement sales & billing logic
* Implement reports (daily, monthly)

Frontend Tasks:

* Build dashboard layout
* Integrate forms for product entry & stock update
* Create sales page with live calculation
* Display low-stock alerts
* Connect frontend with backend API

Integration: Connect DB (MongoDB/MySQL), setup API routes, test via Postman.

Deliverables: Working prototype, API documentation.

**5. Testing Phase**

Goal: Ensure system quality and fix issues.

* Write test cases (functional + non-functional)
* Unit testing (e.g., product add, update, delete)
* Integration testing (frontend + backend)
* User acceptance testing (UAT)
* Fix bugs and optimize performance

Deliverables: Test Case Document, Bug Report, Tested System.

**6. Deployment & Maintenance**

Goal: Deploy and maintain system.

* Prepare Deployment Plan
* Host backend (e.g., Render, Railway)
* Host frontend (e.g., Netlify, Vercel)
* Configure domain and environment variables
* Prepare User Manual / Training Guide
* Collect feedback and track issues
* Schedule regular maintenance & backups

Deliverables: Deployed System URL, User Manual, Maintenance Log.