Project Topic Proposal

Project Title: *SIMPLE INVENTORY MANAGEMENT SYSTEM FOR SMALL BUSINESSES*

Overview
The goal of this project is to create a lightweight inventory management tool for small businesses to track stock levels, monitor sales, and alert when restocking is needed. It will consist of a simple user interface and basic backend functionality.

Problem Solved
Small businesses often struggle with expensive or complex inventory tools. This system offers an easy-to-use, affordable alternative.

Task Breakdown, Assigned Members, and Timelines

Task	Description	Assigned Member	Time Needed	Deadline	Dependencies
Research and Planning	Finalize system features and architecture (UI, Backend, Database).		2 hours	Sunday Night	None
GitHub Setup	Create repository, add team members, and set up project board.		1 hour	Sunday Night	None
User Interface Design	Create wireframes for pages like Add Item, Stock Levels, and Alerts.		3 hours	Sunday Night	Planning
Backend Development	Create APIs for adding, updating, and checking inventory data.		5 hours	Sunday Night	Planning, GitHub Repo
Frontend Development	Build the UI for the app and connect it to the backend using APIs.		5 hours	Sunday Night	Backend Development
Testing and Bug Fixing	Test the system, fix bugs, and ensure it works smoothly.	Entire Team	3 hours	Sunday Night	Backend and Frontend Complete
Report Writing	Compile all sections of the report: Introduction, Tasks, Screenshots, etc.		3 hours	Sunday Night	All other tasks
Presentation Prep	Create acknowledgement and practice delivering it for submission.	Entire Team	2 hours	Monday night	Report and Testing Complete

Steps to Complete and Solve as a Team

1. **Kickoff Meeting (Sunday):**
   o Confirm the project topic and system features.
   o Review assigned tasks and clarify doubts.
   o Ensure everyone has access to GitHub.
2. **Daily Updates:**
   o Use GitHub's project board to mark tasks as *In Progress* or *Done*.
3. **Midweek Check-In (Wednesday):**
   o Review progress on Backend and UI.
   o Reassign tasks if any delays occur.
4. **Final Check (Saturday):**
   o Ensure the entire system works (testing phase).

Compile all deliverables (screenshots, report, code)

Overview to Share with the Team

What is the project?
We are building a **Simple Inventory Management System** that helps small business owners:

1. Track their inventory (how much stock they have).
2. Monitor sales and updates in real-time.
3. Get alerts when stock is low.

Why is this important?
Small businesses often don't have expensive systems to track inventory. Our tool is easy to use and helps them avoid running out of stock or over-ordering items.

What must we deliver?

1. A simple, functional system that has:
   o A user-friendly interface (UI) for adding/viewing stock.
   o A backend to store and process inventory data.
   o Notifications when stock is low.
2. A **project report** with explanations, screenshots, and how the system works.

How to Fix and Solve the Problem

Here's how we'll solve the problem in steps:

1. **Understand User Needs**

- Small businesses need a way to:
  - Add inventory details (item name, quantity, price).
  - View all stock at once.
  - Get alerts when something needs restocking.
2. **Plan the System Architecture**
   - **UI (Frontend):** The screens the user interacts with (Add Stock, View Inventory).
   - **Backend:** The system that processes and stores the data (using a database).
   - **Database:** Stores all inventory details securely.
3. **Build the System**
   - Create basic screens (UI).
   - Write backend code to process data.
   - Connect the UI to the backend.
   - Test it all to ensure it works as planned.
4. **Write the Report**
   - Include what we built, how it solves the problem, and screenshots.

Team Member Responsibilities and Timeline

Each member gets a specific task to finish in **2 days**. This ensures the system is ready for testing and the report is nearly complete.

Task	Description	Assigned Member	Deadline	Dependencies
Research Features	List all the features and workflows of the system.	Member 1 (Planner)	Day 1 (End)	None
UI Design	Create simple mockups for screens: Add Stock, View Stock, Restock Alerts.	Member 2 (Designer)	Day 1 (End)	Research Features
Backend Code	Build backend APIs to process inventory (e.g., Add Item, View All Items, Check Alerts).	Member 3 (Backend)	Day 2 (End)	UI Design Complete
Database Setup	Create tables for storing inventory (Item Name, Quantity, Price, Alert Threshold).	Member 4 (DB Admin)	Day 1 (End)	Research Features
Frontend Development	Build the screens (UI) and connect them to the backend.	Member 5 (Frontend)	Day 2 (End)	Backend/Database Complete
System Testing	Test the system: add items, check stock, and ensure alerts work.	Member 6 (Tester)	Day 2 (End)	Frontend/Backend Complete

Task	Description	Assigned Member	Deadline	Dependencies
Report Writing	Write the project report, including screenshots and explanations.	Member 7 (Writer)	Day 2 (End)	Testing Complete

Detailed Steps for Each Task

1. Research Features (Member 1)

- Think about what a small business owner would need:
  - Add Inventory: Add item name, price, and quantity.
  - View Inventory: See all items and their stock levels.
  - Restock Alerts: Show items that are below a certain quantity.
- Write these down and share with the group by **Day 1 evening**.

2. UI Design (Member 2)

- Use simple tools like Canva, Figma, or draw on paper.
- Design three screens:
  1. Add Inventory: Fields for item name, quantity, and price.
  2. View Stock: A list of all inventory items.
  3. Restock Alerts: A list of items with low stock.
- Share designs with the group by **Day 1 evening** for approval.

3. Backend Code (Member 3)

- Write functions for:
  1. Adding inventory to the database.
  2. Fetching all items from the database.
  3. Checking stock levels and returning alerts.
- Test these APIs using a tool like Postman and share results by **Day 2 evening**.

4. Database Setup (Member 4)

- Create a simple table in a database (e.g., MySQL or SQLite) with columns:
  - Item ID, Item Name, Quantity, Price, Alert Threshold.
- Share the database structure and sample data with the group by **Day 1 evening**.

5. Frontend Development (Member 5)

- Use React.js or another tool to create:
    - A form for adding inventory.
    - A page for viewing inventory.
    - A page for restock alerts.
- Connect the frontend to the backend by **Day 2 evening**.

---

6. System Testing (Member 6)

- Test the entire system:
    1. Add sample inventory.
    2. Check if stock levels display correctly.
    3. Verify that restock alerts appear for low stock.
- Fix any bugs and share results with the team by **Day 2 evening**.

---

7. Report Writing (Member 7)

- Write a clear and simple report:
    - Introduction: What is the project?
    - Features: What does it do?
    - Architecture: Explain the UI, Backend, and Database.
    - Screenshots: Add pictures of the UI and database.
- Share a draft by **Day 2 evening** for feedback.

Daily Timeline

Day	Tasks to Complete	Deadline
Day 1	Research Features, Database Setup, UI Design	End of Day 1
Day 2	Backend Development, Frontend Development, Testing	End of Day 2
Day 2	Complete Report Writing	End of Day 2