

Smart Stock Inventory For Retail Stores

Day1:

overview of the project

Received an introduction to the internship structure, rules, and expectations. Understood the overall flow of tasks and learning outcomes.

We discussed about the main problems in retail inventory management such as overstock and out of stock situations.

DAY2:

We discussed and finalized the technical components required for building the Smart Stock Inventory Optimization System.

Frontend: HTML, CSS, JavaScript for creating the user interface and dashboard.

Backend: Python for implementing the core logic, stock calculations, alerts, and prediction models.

Database: MySQL / SQL / MongoDB for storing product details, sales records, supplier info, and stock level.

NumPy: A Python library used in the backend to perform numerical calculations, mathematical operations, and forecasting formulas efficiently.

Pandas: A data handling library used for loading, cleaning, analyzing, and processing datasets such as product data and sales records.

Python Format Strings: Formatting feature in Python used to print structured, clean, and readable outputs in the backend system.

API Basics: Understanding how different systems communicate APIs such as fast API and flask APIs. used if the project needs frontend-backend data exchange.

DAY3:

Flask API: Flask is a lightweight Python web framework used to create simple APIs. It allows the backend to send and receive data easily between the user interface and the server. Flask is easy to learn and suitable for small-scale projects.

Fast API: FastAPI is a modern and very fast Python framework for building APIs. It supports high performance, automatic documentation, and faster request handling. FastAPI is suitable for large-scale and real-time applications.

Difference Between Flask and Fast API :

Flask: Simple, lightweight, beginner-friendly.

FastAPI: Faster, modern, supports automatic documentation

Postman : Postman is a tool used to test APIs by sending requests and checking responses. Developers use it to debug and verify API endpoints.

DAY4:

GitHub Overview: GitHub is an online platform that stores Git repositories. It allows sharing, collaboration, and remote code storage .

Repository (Repo): A repository is a folder where your project files and their full history are stored.

Local Repository: A Git repository stored on your computer.

Cloning : Copying a GitHub repository to your local system.

Committing : Saving your code changes in Git with a message describing the update.

Pushing : Sending your local commits to the GitHub remote repository.

Pulling: Getting the latest changes from GitHub to your local system.

Merge: Combining changes from one branch into another.

Fork: Creating your own copy of another person's repository on GitHub.

Pull Request (PR):A request to merge your code changes into the main project branch.

Git Commands :

git init: Initializes a new Git repository in your project folder.

git clone : Copies a GitHub repo to your system. **git status:** Shows the current changes in your project.

git add . : Stages all changes to be committed.

git commit -m "message" : Saves the staged changes with a message.

git push origin main : Uploads your commits to the remote GitHub repo.

git pull : Downloads latest updates from GitHub.

git branch : Shows all branches in your repo.

Day5:

Introduction to SQL

SQL : Structured Query Language, used to interact with databases. Can create, read, update, and delete data (CRUD operations)

Create (INSERT): Add new records to a table

Read (SELECT): Retrieve data from a table.

Update: Modify existing records in a table

Delete: Remove records from a table.

Database: Collection of tables.

Table: Collection of rows (records) and columns (fields).

Primary Key: Unique identifier for a row.

Foreign Key: Links tables together.

Artificial Intelligence: AI is the simulation of human intelligence by machines to perform tasks like learning, reasoning, and problem-solving. Example: ChatGPT, self-driving cars.

Machine learning: ML is a subset of AI where machines learn from data and improve performance without being explicitly programmed. Example: Predicting sales, spam email detection.

Types of ML

supervised Learning – Learning from labeled data.

Example: Predicting product demand.

Unsupervised Learning – Finding patterns in unlabeled data.

Example: Customer segmentation.

Reinforcement Learning – Learning by trial and error using rewards.

Example: Game-playing AI, robotic movement.

Day6:

What is normalization

Normalization is the process of scaling or adjusting data so that it fits within a certain range or standard format. This is often done in databases or data preprocessing for machine learning.

There are two main contexts where normalization is used:

Database normalization:

In databases, normalization is about structuring a database to reduce redundancy and improve data integrity.

Purpose: Avoid duplicate data, ensure consistent updates, and make queries efficient.

How: Organize data into tables with relationship

Common Forms of Database Normalization:

1. **1NF (First Normal Form)**: Ensure each column has atomic values (no lists in a cell).
2. **2NF (Second Normal Form)**: Meet 1NF and remove partial dependencies (non-key columns depend on the whole primary key).
3. **3NF (Third Normal Form)**: Meet 2NF and remove transitive dependencies (non-key columns depend only on the primary key).
4. **BCNF (Boyce-Codd Normal Form)**: Stronger version of 3NF to remove anomalies.

Day7:

GitHub repository:

- A centralized place to store and manage version-controlled project code.

Project Review :

- Evaluated current project progress and clarified next development steps