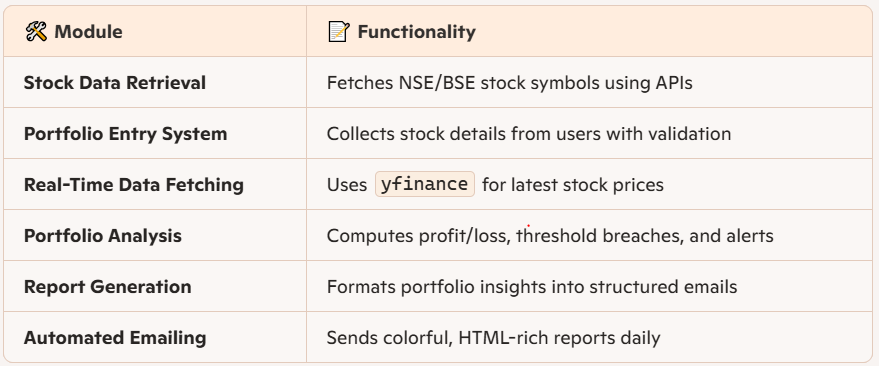
**🎯 Project Specification Document – Portfolio Monitoring System**

🔹 **1. Project Overview**  
🎯 **Objective:**  
Automate a daily email report for portfolio holders, detailing:

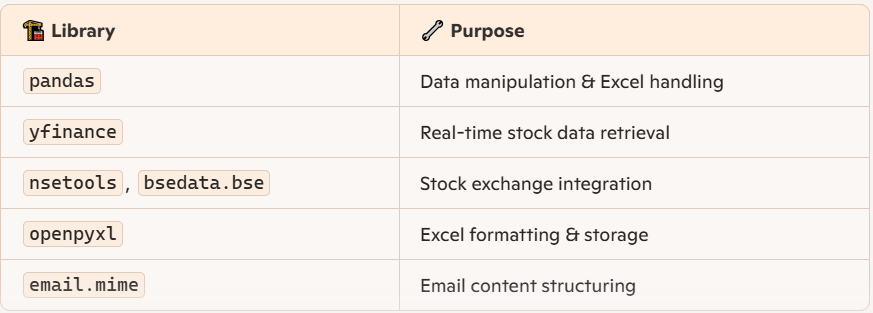
* 📈 Current Profit/Loss per Stock
* 🚨 Threshold Breach Alerts

💡 **Purpose**: Quickly detect significant changes in a user’s portfolio for informed decision-making.

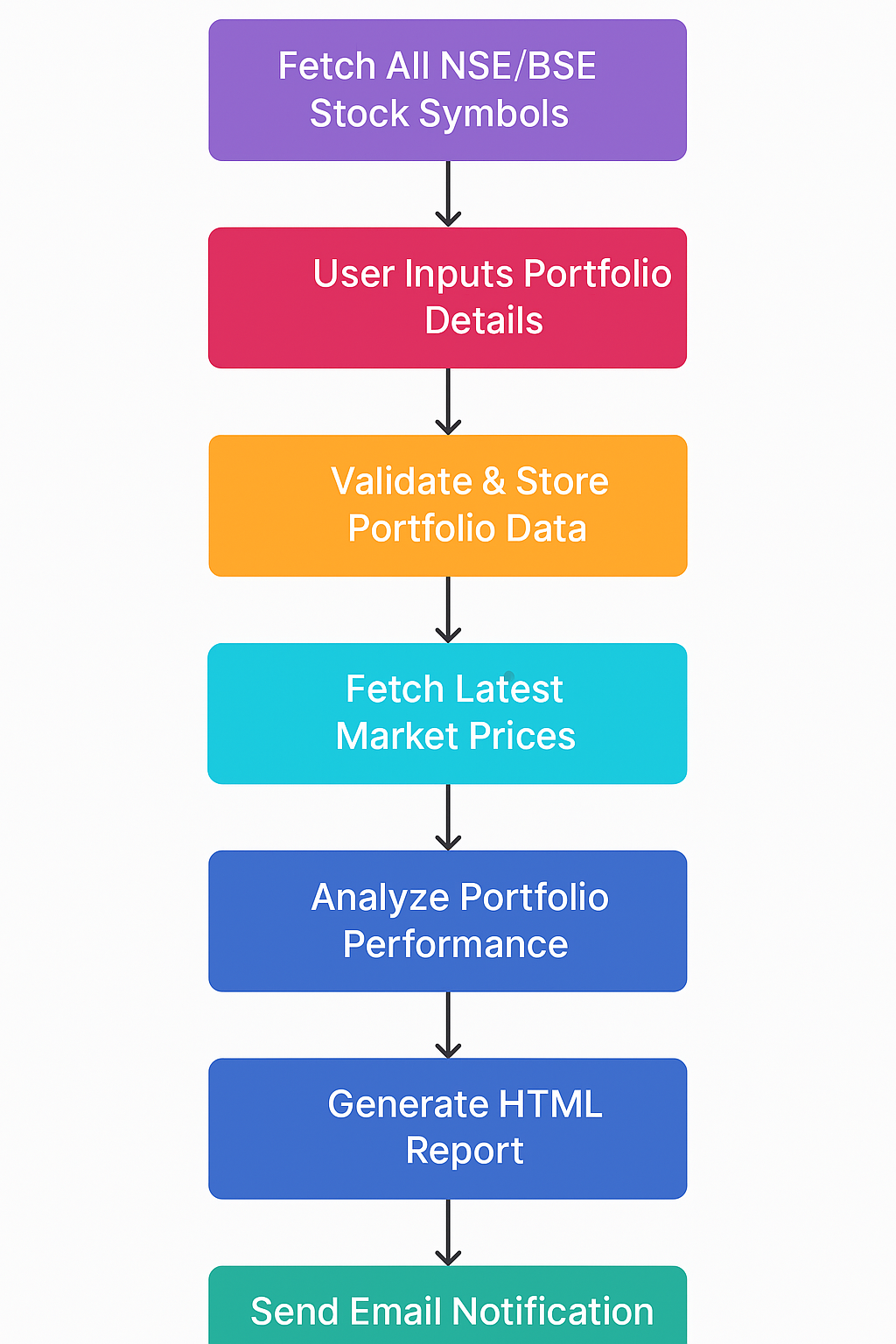
**🔹 2. System Architecture & Features**  
**📌 Key Components**



**📌 Technology Stack**



**🔹 3. Workflow & Execution Flowchart**  
🖼 Flowchart illustrating the complete process from stock retrieval to reporting has been included!



**🔹 4. Code Overview & Functionality Breakdown**

1️⃣ **Stock Data Retrieval (NSEBSE Names Integrated.py)**  
**Purpose:** Fetches stock symbols from NSE and BSE using APIs (nsetools, bsedata.bse, yfinance).  
**Process:** Cleans and standardizes stock names, stores them in NSEBSE\_stocks\_name.xlsx.  
**Outcome:** Provides an extensive list of tradable stock symbols with their reference names.

2️⃣ **Portfolio Entry & Storage (Portfolio Puller.py)**  
**Purpose:** Collects portfolio details from users (Holder’s name, Stock, Exchange, Threshold %, Shares).  
**Process:** Validates stock names from NSEBSE\_stocks\_name.xlsx, structures input, and stores in Portfolio\_details.xlsx.  
**Outcome:** Creates a structured, validated dataset for further analysis.

3️⃣ **Fetching Live Market Prices (API Builder Portfolio Specific.py)**  
**Purpose:** Uses yfinance API to retrieve real-time stock prices for tracked portfolio stocks.  
**Process:** Fetches open, close, high, low, and previous close values, stores results in API\_stock\_prices.xlsx.  
**Outcome:** Provides latest stock market data for analytical computation.

4️⃣ **Portfolio Performance Analysis (Portfolio Analyser.py)**  
**Purpose:** Computes profit/loss, threshold breaches, and percentage changes using live market data.  
**Process:** Merges Portfolio\_details.xlsx and API\_stock\_prices.xlsx, calculates high-low variations, stores results in Portfolio\_Analyser.xlsx.  
**Outcome:** Flags stocks that cross predefined thresholds and evaluates portfolio performance.

5️⃣ **Automated Report Generation (HTML Email Report)**  
**Purpose:** Generates dynamic, color-coded reports summarizing portfolio holder’s stock movements.  
**Process:** Computes total portfolio value, profit/loss, gainers & losers, formats email as HTML-rich content.  
**Outcome:** Sends daily email alerts with personalized portfolio insights.

6️⃣ **Final Optimized Execution (Colourful Assembler Optimised.py)**  
**Purpose:** Unifies all modules into a single streamlined execution process.  
**Process:** Runs data fetching, portfolio validation, price updates, analysis, and reporting seamlessly.  
**Outcome:** Fully automated monitoring system with real-time alerts.

**🔹 5. Portfolio Report Format**