Data to Decisions

Project Abstract:

The project aims to design and implement a data analytics system capable of extracting sales, product, and customer information from multiple sources such as spreadsheets and APIs. The system will continuously monitor for updates, process the data, and store it in a database. The system will leverage AI technologies to suggest reports, charts, and statistics based on data availability. Additionally, it will generate detailed descriptive reports with statistical insights and send automated reports to users via email daily, weekly, and monthly.

Project Objectives:

- Data Integration:
    - Gather data from spreadsheets and APIs.
    - Process and store sales, product, and customer data in a database.
    - Ensure real-time or periodic updates in the database based on changes in the source data.
- AI-based Reporting:
    - Analyze the data and use AI to suggest possible charts, tables, and statistics.
    - Automatically generate tables and charts for user-defined parameters or based on available data.
- Descriptive Reporting and Automation:
    - Generate descriptive reports with statistical insights and suggestions for improvement.
    - Automate the sending of reports via email, with customizable frequencies such as daily, weekly, and monthly.

Project Requirements

Technologies:

- Programming Language: Python
- Data Analytics Libraries: Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn
- Database: PostgreSQL or MySQL
- API Interaction: Requests, Flask or FastAPI
- Data Processing: Pandas, Openpyxl for spreadsheets, Requests for API
- AI Models for Reporting: Scikit-learn, TensorFlow, or PyTorch (for suggesting stats and charts)
- Email Automation: SMTP, Python libraries like smtplib or using external services like SendGrid or AWS SES
- Data Visualization: Matplotlib, Seaborn, Plotly
- Report Generation: Python-docx, Jinja2 for templates

Suggested Learning Path for Students:

- Python Programming: Focus on understanding data structures, functions, and OOP concepts.
- Data Analysis Tools: Learn how to use Pandas and NumPy for data processing, Scikit-learn for statistical models, and Matplotlib/Seaborn for data visualization.
- Database Management: Basic SQL and working with databases like PostgreSQL or MySQL, and understanding how to integrate databases with Python.
- API Development: Use Flask or FastAPI for building APIs and learn how to interact with external APIs using Python's Requests module.
- AI Technologies: Learn Scikit-learn for basic ML models to suggest statistics and charts.
- Email Automation: Implement email services using Python libraries and APIs like SendGrid or AWS SES.

Milestones:

Milestone 1: Data Collection and Storage (Month 1-2)

- Objective: Develop a system that collects data from spreadsheets and APIs, processes it, and stores it in the database.
- Tasks:
- Create a data ingestion module that accepts spreadsheets (Excel files) and extracts product, sales, and customer information.
- Implement an API integration to fetch sales, product, and customer information (based on provided documentation).
- Store the processed data in a relational database such as PostgreSQL or MySQL.
- Implement a system that regularly checks for data updates and synchronizes the database.

Deliverables:

- Data ingestion scripts for spreadsheets and APIs.
- Database schema design.
- Basic application interface to display ingested data.

Milestone 2: AI-based Data Analysis and Reporting (Month 3-4)

- Objective: Develop AI-powered reporting that suggests charts, tables, and statistical insights based on the available data.
- Tasks:
- Implement AI models that analyze the ingested data to suggest relevant statistics, charts, and tables.
- Build a dashboard that dynamically generates visualizations (charts and tables) based on data availability.
- Develop an interactive interface for users to customize the reports.

Deliverables:

- AI-based report generation module.
- Interactive dashboard displaying dynamic tables, charts, and insights.
- Documentation of AI models and how they make report suggestions.

Milestone 3: Descriptive Reports and Email Automation (Month 5-6)

- Objective: Generate detailed descriptive reports based on available data and automate sending reports via email.
- Tasks:
- Create a module that generates descriptive reports, including statistical summaries and business suggestions.
- Implement a reporting scheduler that sends daily, weekly, and monthly reports to users via email.
- Add user customization options to adjust report parameters and frequency.

Deliverables:

- Automated report generation and email system.
- Customizable user interface for report scheduling.
- Final project report with descriptive analytics.