# Stock Analytics Project Summary

## Summary

Developed a comprehensive stock analytics project that automates the process of fetching, processing, and analyzing stock data from Yahoo Finance. The project includes functionalities for generating detailed reports, sending email notifications, and maintaining a SQLite database for historical stock data. The project follows best practices for database schema management and deployment using Alembic, ORM models, GitHub, and Google Cloud.

## Key Responsibilities

### Data Fetching and Processing

- Utilized yfinance to fetch historical and real-time stock data for S&P 500, NASDAQ, and user-defined watchlist tickers.

- Implemented data enrichment by merging stock data with sector and industry information from a SQLite database.

- Calculated stock returns over various lookback periods for performance analysis.

### Database Management

- Designed and maintained a SQLite database to store historical stock data, sector, and industry information.

- Implemented upsert functionality to efficiently update or insert stock data records.

- Utilized Alembic for database schema migrations, ensuring smooth schema updates and version control.

- Followed best practices by using ORM models to interact with the database, avoiding raw SQL strings.

### Report Generation

- Generated detailed Excel and HTML reports for market scanners, watchlists, and broad market monitoring.

- Created visualizations using Plotly to display sector average PE ratios and top gainers/losers.

- Automated the generation and emailing of daily, weekly, and monthly stock performance reports.

### Error Handling and Logging

- Implemented robust error handling and logging mechanisms to ensure smooth execution and easy debugging.

- Configured email alerts for critical errors to notify the development team promptly.

### Automation and Scheduling

- Automated the entire workflow using Python scripts, enabling scheduled data fetching, processing, and report generation.

- Utilized environment variables for configuration management and secure handling of sensitive information.

### Deployment and Version Control

- Deployed the project on Google Cloud, ensuring scalability and reliability.

- Used GitHub for version control, collaboration, and continuous integration.

## Technologies Used

Python

Yfinance(A Python library to fetch historical and real-time stock data from Yahoo Finance)

SQLAlchemy(An ORM (Object-Relational Mapping) library for Python, used to interact with the SQLite database without writing raw SQL queries

Alembic(A database migration tool for SQLAlchemy, used to manage database schema changes and version control)

SQLite(A lightweight, disk-based database used to store historical stock data, sector, and industry information)

Plotly(A graphing library for creating interactive visualizations, used to generate charts for sector average PE ratios and top gainers/losers.)

Pandas( A data manipulation and analysis library for Python, used for data processing and analysis.)

smtplib (for email notifications)

argparse (for command-line argument parsing)

dotenv (for environment variable management)

Logging (for error handling and logging)

GitHub (for version control)

Google Cloud (for deployment)