MoMo Data Analysis Assignment Report

Student Name: Clarence Liberi

Course: Enterprise_Web_Development
Date: 2025-04-17

1. Introduction

This project demonstrates fullstack development skills by building a dashboard to analyze and visualize MTN MoMo SMS transaction data. The workflow includes data cleaning, database design, backend API development, and frontend dashboard implementation.

2. Approach & Architecture

- **Data Extraction:** Parsed the provided XML file containing SMS messages using Python.
- **Data Cleaning & Categorization:** Used regex to extract transaction details and categorize each message by type (e.g., incoming money, payment, transfer, deposit).
- **Database Design:** Designed a normalized SQLite schema to store structured transaction data.
- **Backend API:** Built a FastAPI backend to serve transaction data as JSON, supporting filtering by type, date, and amount.
- **Frontend Dashboard:** Developed an HTML/CSS/JS dashboard using Chart.js for visualizations and interactive filters.

3. Key Design Decisions

- **Separation of Concerns:** Codebase is organized into backend/ and frontend/ directories for clarity and maintainability.
- **Technology Choices:** Used FastAPI for a modern, fast backend; SQLite for simplicity; Chart.js for visualizations.
- **Logging:** All unprocessed or uncategorized messages are logged for review.

4. Challenges & Solutions

- **Parsing Free-Form SMS Text:** SMS messages vary in format. Used regular expressions and robust parsing logic to extract structured data.
- **Frontend-Backend Integration:** Ensured CORS was enabled in FastAPI and recommended serving frontend via HTTP server for cross-origin requests.
- **Data Normalization:** Cleaned and standardized amounts, dates, and names for

reliable analysis.

5. Results & Insights

- The dashboard enables filtering, searching, and visualization of transaction volumes and types.
- Provides a foundation for deeper financial analytics and reporting.

6. Future Improvements

- Add authentication for secure access.
- Support for more advanced analytics (e.g., anomaly detection, trend analysis).
- Deploy the app online (e.g., Heroku, Vercel, Netlify).

7. How to Run

See the README.md for setup and usage instructions.

End of Report