MTN MoMo Analytics Project Report

SMS Data Analysis and Visualization System

1. Project Overview

This project involved developing a full-stack application to process and analyze MTN Mobile Money (MoMo) SMS transaction data. The system processes XML-formatted SMS data, categorizes transactions, stores them in a relational database, and presents insights through an interactive dashboard.

2. Technical Approach

2.1 System Architecture

The project follows a modular, layered architecture:
- Frontend: HTML5, JavaScript, and Tailwind CSS for the dashboard
- Backend: Python Flask for API endpoints and data processing
- Database: MySQL for data persistence
- Data Processing: Custom XML parser for SMS data extraction

2.2 Project Structure

2.3 Key Components

1. XML Parser
- Processes raw SMS data from XML format
- Extracts transaction details using regex patterns
- Categorizes transactions based on message content

2. Database Design
- Transactions table storing:
  - Transaction ID
  - Amount
  - Type
  - Date/Time
  - Phone Numbers
  - Status

3. Dashboard Interface
- Real-time transaction summaries
- Interactive charts and graphs
- Filterable transaction list
- File upload functionality

3. Technical Challenges and Solutions

3.1 Data Processing Challenges

1. Inconsistent SMS Formats
- Challenge: Various SMS formats for different transaction types
- Solution: Implemented robust regex patterns and multiple parsing strategies

2. Data Cleaning
   - Challenge: Missing or malformed data in SMS messages
   - Solution: Added validation and default values for incomplete data

3.2 System Integration Challenges

1. File Upload Processing
   - Challenge: Handling large XML files efficiently
   - Solution: Implemented chunked processing and progress tracking

2. Database Connectivity
   - Challenge: Maintaining stable database connections
   - Solution: Implemented connection pooling and error handling

4. Key Decisions

4.1 Technology Stack Selection

1. Frontend
   - Chose Tailwind CSS for rapid UI development
   - Used Chart.js for responsive visualizations
   - Implemented vanilla JavaScript for better performance

2. Backend
   - Selected Flask for its lightweight nature and easy integration
   - Used MySQL for robust transaction handling
   - Implemented modular structure for maintainability

4.2 Design Decisions

1. Data Processing
   - Separate parser module for better code organization
   - Transaction categorization based on keyword analysis
   - Error logging for failed transactions

2. User Interface
   - Real-time updates for better user experience
   - Responsive design for mobile accessibility
   - Intuitive file upload interface

5. Future Improvements

1. Performance Optimization
   - Implement caching for frequently accessed data
   - Optimize database queries for larger datasets
   - Add batch processing for multiple files

2. Feature Enhancements
   - Add export functionality for reports

- Implement more advanced analytics
- Add user authentication and roles

6. Conclusion

The MTN MoMo Analytics system successfully demonstrates the ability to process, analyze, and visualize transaction data effectively. The modular architecture ensures maintainability and scalability for future enhancement

HOW TO RUN THE SITE:

Git clone : https://github.com/kol-apo/MoMo_Analysis_6.git