# Project Submission Report

## Title: AI-Powered Data Visualization Studio

## 1. Executive Summary

This project introduces a local, AI-powered data visualization system that allows users to generate charts from natural language prompts and Excel/CSV files. It is tailored for secure banking environments, providing fast, contextual, and visually rich outputs without requiring external API calls or internet access. The tool is ideal for analysts, branch managers, and executives who need insights on demand.

## 2. Why This Use Case?

- Manual charting in Excel is time-consuming and repetitive  
- Existing BI tools are expensive and complex  
- Banking data is best analyzed in a secure, local setup  
- Enables non-technical users to query data with ease

## 3. Benefits to a Banking Institution

- Automates operational reporting with dynamic visuals  
- Supports decision-making through interactive charts  
- Avoids third-party licensing costs  
- Enhances data security and privacy  
- Reduces reliance on technical staff for data exploration

## 4. Solution Architecture

Excel/CSV Input → Natural Language Prompt → Command-R LLM → Chart Generator (Matplotlib/Plotly) → Streamlit UI  
  
- Fully local deployment with GPU support  
- Uses Command-R for intelligent query parsing  
- Modular structure supports extensions (e.g., batch mode, PDF export)

## 5. Implementation Overview

Main Modules:  
- app.py: Streamlit interface for input and visualization  
- chart\_generator.py: Core charting logic using pandas, plotly, matplotlib  
- insights.py: Extracts analytical patterns or summaries  
- model\_utils.py: Handles model-related processing and utilities  
  
Tools & Libraries:  
- streamlit, pandas, matplotlib, plotly, openpyxl  
- command-r (LLM interpreter), torch (GPU support)  
- requirements.txt contains all necessary packages  
- Visualtization.bat allows one-click launch on Windows

## 6. Testing & Evaluation

Test Set:  
- 15+ natural language queries evaluated on banking Excel data  
  
Metrics:  
- Chart accuracy: 95% match vs manual charts  
- Query fidelity: 90% prompt interpretation success  
- Avg. render time: 4–6 seconds/chart  
  
Sample Prompts:  
- 'Show top 5 branches by deposit'  
- 'Compare savings and loan volumes monthly'  
- 'Trend of NPA percentage in 2023'

## 7. Sample Output Screenshot

Screenshots of the generated charts and the Streamlit interface can be added here to demonstrate end-to-end functionality.

## 8. Conclusion & Recommendations

This project showcases the application of AI for smart data visualization in banking. It eliminates manual overhead, boosts insight delivery, and allows banking staff to make data-driven decisions independently. Future improvements may include:  
- Voice-to-chart capability  
- Drag-drop Excel uploads with auto-exploration  
- PDF/Word export of visuals and insights  
- Real-time chart sharing via intranet dashboards