Project Documentation: PhonePe Transaction Insights

Project Title:

PhonePe Transaction Insights Dashboard

Objective:

To analyze, visualize, and gain strategic insights from PhonePe's digital transaction data across various dimensionsstates, quarters, categories, devices, and insuranceusing Python, PostgreSQL, and Streamlit.

Tech Stack:

- Frontend: Streamlit
- Backend: PostgreSQL (SQL)
- Scripting/Data Handling: Python, Pandas
- Visualization: Plotly, Streamlit Charts
- Deployment: Local / Streamlit Cloud

Data Sources:

Extracted from PhonePe Pulse GitHub repository:

- Aggregated transaction/user/device data
- Top transaction/user/insurance data
- Map data (hover + district)
- JSON files converted to CSVs and loaded into PostgreSQL

Database Schema:

- top_user(state, year, quarter, brand, registeredusers_count, oppopens)
- top_map(state, year, quarter, transaction_type, transaction_count, transaction_amount)
- top_insurance(state, year, quarter, count, amount)
- More: map_user, map_transaction, aggregated_user, etc.

Key Business Cases & Dashboard Pages:

1. Decoding Transaction Dynamics

- Compare state-wise transactions over quarters & types

- SQL aggregation + Streamlit bar chart

2. Device Dominance & User Engagement

- Analyze most-used devices and app opens by brand

- SQL grouping + Streamlit dataframe

3. Insurance Penetration

- Track insurance growth and volume by state/year

- SQL aggregation + Streamlit line chart

4. Market Expansion Potential

- Identify top-performing states by total transaction amount

- SQL ordering + bar chart

5. User Engagement Strategy

- Analyze total app opens and users over time

- SQL by year + Streamlit line chart

6. India-Wide Transaction Map

- Choropleth map of state-wise transaction volume

- Plotly + GeoJSON mapping

Features:

- Interactive filters: select State and Year

- Real-time data loading from PostgreSQL

- Data caching using @st.cache_data

- Indian map visualization using plotly.express.choropleth

How to Run:

1. Install dependencies:

   pip install streamlit pandas sqlalchemy psycopg2 plotly

2. Set up PostgreSQL:

   - Create database: project_phonepe

   - Import tables using CSVs or scripts

3. Run the app:

   streamlit run phonepe.py

File Structure (example):

phonepe_dashboard/

 phonepe.py

 data/

    csv_files/

 schema.sql

 requirements.txt

Outcomes:

- Helped visualize digital transaction growth

- Identified state/device/category-level insights

- Supports PhonePe in targeting growth areas and user retention