

PhonePe Transaction Insights

Technical Documentation

Data Science Project

August 20, 2025

Contents

| | | |
|----------|---|----------|
| 1 | Overview | 2 |
| 1.1 | Technology Stack | 2 |
| 1.2 | Key Features | 2 |
| 2 | Setup Instructions | 2 |
| 2.1 | Prerequisites | 2 |
| 2.2 | Installation Process | 2 |
| 2.2.1 | Step 1: Repository Setup | 2 |
| 2.2.2 | Step 2: Dependencies Installation | 2 |
| 2.2.3 | Step 3: Database Configuration | 3 |
| 2.2.4 | Step 4: Data Processing Pipeline | 3 |
| 2.2.5 | Step 5: Launch Application | 3 |
| 3 | Application Architecture | 3 |
| 3.1 | Database Design | 3 |
| 3.2 | Application Components | 3 |
| 4 | Business Case Studies | 3 |
| 4.1 | Transaction Dynamics Analysis | 4 |
| 4.2 | Device Usage & User Engagement | 4 |
| 4.3 | Insurance Market Analysis | 4 |
| 4.4 | Market Expansion Strategy | 4 |
| 4.5 | User Growth Analysis | 5 |
| 5 | Technical Implementation | 5 |
| 5.1 | Data Processing Pipeline | 5 |
| 5.2 | Visualization Framework | 5 |
| 5.3 | Performance Optimization | 5 |
| 6 | Usage Guidelines | 5 |
| 6.1 | Navigation | 5 |
| 6.2 | Dashboard Features | 6 |
| 6.3 | Case Study Analytics | 6 |
| 6.4 | Data Refresh Process | 6 |
| 7 | Conclusion | 6 |

1 Overview

The PhonePe Transaction Insights Dashboard is a comprehensive data analytics application built with Streamlit that analyzes India's digital payment transaction landscape. It provides interactive visualizations and business insights derived from PhonePe's transaction data to support strategic decision-making.

1.1 Technology Stack

- **Frontend:** Streamlit
- **Database:** MySQL
- **Data Processing:** Python, Pandas
- **Visualization:** Plotly
- **Mapping:** GeoJSON for Indian states

1.2 Key Features

- Interactive dashboard with choropleth maps
- Real-time data filtering by year and quarter
- Multiple business case study scenarios
- Transaction pattern analysis across states
- Device usage and user engagement analytics
- Insurance market analysis and growth tracking

2 Setup Instructions

2.1 Prerequisites

Before setting up the application, ensure you have the following installed:

- Python 3.7 or higher
- MySQL Server
- Git for version control

2.2 Installation Process

2.2.1 Step 1: Repository Setup

```
1 git clone <https://github.com/PhonePe/pulse.git>
```

Listing 1: Clone Repository

2.2.2 Step 2: Dependencies Installation

```
1 pip install streamlit pandas plotly sqlalchemy mysql-connector-python
```

Listing 2: Install Required Packages

2.2.3 Step 3: Database Configuration

Create a MySQL database and update connection credentials:

```
1 engine = create_engine("mysql+mysqlconnector://username:password@localhost:3306/phonepe_db")
```

Listing 3: Database Connection Configuration

2.2.4 Step 4: Data Processing Pipeline

1. Execute `phonepe_analysis.ipynb` to extract and process data from PhonePe Pulse repository
2. Run `pysql.ipynb` to create database tables and insert processed data
3. Place `Indian_States.geojson` file in the project root directory

2.2.5 Step 5: Launch Application

```
1 streamlit run app1.py
```

Listing 4: Start Dashboard

3 Application Architecture

3.1 Database Design

The application utilizes a structured database with nine tables organized into three categories:

| Category | Table Name | Purpose |
|------------|------------------------|--|
| Aggregated | aggregated_transaction | Transaction data by state/quarter/type |
| | aggregated_insurance | Insurance transaction aggregates |
| | aggregated_user | User data segmented by device brands |
| Map | map_transaction | District-level transaction mapping |
| | map_insurance | District-level insurance mapping |
| | map_user | District-level user registration data |
| Top | top_transaction | Top performing districts/pincodes |
| | top_insurance | Top insurance performance areas |
| | top_user | Top user registration areas |

Table 1: Database Table Structure

3.2 Application Components

The dashboard consists of two main sections:

- **Dashboard:** Overview with key metrics and visualizations
- **Case Studies:** Five detailed business analysis scenarios

4 Business Case Studies

The application implements five comprehensive business case studies based on PhonePe's strategic requirements:

4.1 Transaction Dynamics Analysis

Objective: Analyze transaction patterns across states, quarters, and payment types for strategic decision making.

Key Features:

- State-wise transaction heatmaps
- Top 10 states by transaction amount
- Payment type distribution analysis
- Quarterly performance tracking

4.2 Device Usage & User Engagement

Objective: Understand user device preferences and engagement patterns to optimize app performance.

Analytics Include:

- Device brand distribution analysis
- Top districts by app opens
- User engagement metrics
- Regional device preference patterns

4.3 Insurance Market Analysis

Objective: Analyze insurance transaction growth and identify market expansion opportunities.

Insights Provided:

- Insurance coverage heatmaps
- Quarterly growth trend analysis
- State-wise insurance penetration
- Market opportunity identification

4.4 Market Expansion Strategy

Objective: Identify high-potential regions and growth opportunities for market expansion.

Strategic Tools:

- Market penetration heatmaps
- Growth opportunity scoring
- Transaction density analysis
- Expansion priority ranking

4.5 User Growth Analysis

Objective: Analyze user registration patterns and engagement metrics for growth strategy.

Metrics Include:

- User distribution heatmaps
- Engagement rate calculations
- Registration pattern analysis
- State-wise user growth tracking

5 Technical Implementation

5.1 Data Processing Pipeline

The data processing follows a structured ETL approach:

1. **Extract:** Raw JSON data from PhonePe Pulse GitHub repository
2. **Transform:** Data normalization, state name standardization, and structure conversion
3. **Load:** Processed data insertion into MySQL database tables

5.2 Visualization Framework

The application employs Plotly for interactive visualizations:

- **Choropleth Maps:** Geographic representation of metrics across Indian states
- **Bar Charts:** Comparative analysis and rankings
- **Pie Charts:** Distribution analysis
- **Line Charts:** Temporal trend analysis

5.3 Performance Optimization

- Database query optimization for rapid dashboard loading
- Streamlit caching implementation for data loading functions
- Efficient data aggregation techniques
- Error handling for robust operation

6 Usage Guidelines

6.1 Navigation

The dashboard provides intuitive navigation through:

- Sidebar radio buttons for section selection
- Interactive filters for year and quarter selection
- Dynamic visualizations that update based on user selections

6.2 Dashboard Features

- **Quick Statistics:** Key performance indicators display
- **Transaction Heatmap:** Interactive state-wise visualization
- **Trend Analysis:** Time series transaction patterns

6.3 Case Study Analytics

Each case study provides:

- Time period selection controls
- Multiple visualization perspectives
- Interactive charts and maps
- Detailed metric analysis

6.4 Data Refresh Process

To update with new PhonePe data:

1. Run `phonepe_analysis.ipynb` with updated data files
2. Execute `pysql.ipynb` to refresh database tables
3. Restart the Streamlit application
4. Verify data integrity and completeness

7 Conclusion

The PhonePe Transaction Insights Dashboard represents a comprehensive solution for analyzing India's digital payment ecosystem. By combining robust data processing capabilities with interactive visualizations, it enables stakeholders to make informed strategic decisions based on empirical evidence.

The modular architecture ensures scalability and maintainability, while the user-friendly interface makes complex data analysis accessible to business users. The five implemented business case studies address key strategic questions and provide actionable insights for market expansion, user engagement optimization, and product development.

This documentation serves as a complete guide for setup, usage, and maintenance of the dashboard, ensuring successful implementation and operation of the analytics platform.