PhonePe Transaction Analysis

By: R. Boneys Dharan

# Abstract

This project focuses on analyzing the PhonePe transaction data to extract insights into user behavior, device usage, insurance adoption, and regional trends. Using data visualization techniques, the project aims to help understand user preferences and market demands for digital payment platforms in India. This study reveals patterns and provides data-driven recommendations for market expansion and user engagement strategies.

# Introduction

PhonePe is one of India’s leading digital payment platforms. With rapid growth in digital transactions, analyzing transaction dynamics, user engagement levels, and insurance-related behavior becomes essential for product innovation and business strategy. This project leverages PhonePe's aggregated data to uncover insights that can shape future initiatives.

# Problem Statement

With the increasing reliance on digital payment systems like PhonePe, understanding the dynamics of transactions, user engagement, and insurance-related data is crucial for improving services and targeting users effectively. This project analyzes and visualizes PhonePe data to help in strategic planning.

# Objectives

The primary objectives of this project are:  
- To analyze device usage patterns and user engagement trends.  
- To evaluate insurance adoption across different states.  
- To study transaction volumes and their growth over time.  
- To identify top-performing regions and stagnating areas.  
- To suggest actionable strategies based on data findings.

# Tools and Technologies Used

Languages & Libraries:  
- Python  
- Pandas, Streamlit, Plotly, JSON, OS, Sklearn  
  
Database:  
- MySQL  
  
Software Tools:  
- Visual Studio Code  
- MySQL Workbench (optional)

**1. Pandas**

Pandas is a fast, flexible, and expressive data analysis library built on top of Python. It provides data structures like Data Frame and Series to manage labelled and relational data efficiently. In this project, Pandas was used to load and preprocess PhonePe’s JSON data files, allowing easy manipulation of large datasets. It enabled tasks such as filtering, grouping, aggregating, and merging data for better analysis. Pandas is ideal for handling structured data due to its SQL-like operations. Its compatibility with other libraries makes it a central tool in data analysis pipelines. The project’s entire data cleaning and preparation phase was managed through Pandas functions. It made working with complex hierarchical PhonePe datasets seamless and efficient.

**2. Streamlit**

Streamlit is an open-source Python library that turns data scripts into interactive web apps quickly. It is designed specifically for data scientists and analysts to deploy dashboards without web development skills. In this project, Streamlit was used to build a dynamic dashboard that allows users to explore PhonePe transaction and engagement data visually. It supports interactive widgets like dropdowns, sliders, and maps, making the data exploration user-friendly. Streamlit automatically updates the UI when the underlying Python code changes. It was crucial for creating a lightweight, browser-based interface to showcase trends and insights. Its simplicity and seamless integration with Pandas and Plotly made it ideal for real-time visualization. The final application is deployable and shareable, requiring no HTML, CSS, or JS knowledge.

**3. Plotly**

Plotly is a Python graphing library that makes interactive and publication-quality visualizations easy to build. It supports a wide range of plots, including line charts, bar charts, pie charts, scatter plots, and choropleth maps. For this project, Plotly was used to visualize transaction trends, user distributions, and regional insights through dynamic charts. It allowed zooming, hovering, and filtering directly on the plots, improving user interaction and data understanding. Plotly’s integration with Pandas helped in building responsive charts with minimal effort. The use of choropleth maps allowed regional comparisons, enhancing the geographical aspect of the analysis. Its elegant visuals improved the presentation quality of the project significantly. Plotly made storytelling through data more engaging and impactful.

**4. JSON (JavaScript Object Notation)**

JSON is a lightweight data-interchange format that is easy for humans to read and write, and easy for machines to parse and generate. It is commonly used for storing and exchanging data between systems. In this project, JSON files were the primary data source, containing state-wise and category-wise PhonePe metrics. Python’s Json module was used to parse and convert these files into dictionaries and DataFrames. JSON helped preserve the hierarchical structure of the raw PhonePe data. It ensured that all transaction types, district-level data, and time-based values could be accessed efficiently. Once converted, this structured data was processed using Pandas. The simplicity and widespread use of JSON made data extraction manageable and consistent.

**5. OS (Operating System Module)**

The OS module in Python provides functions for interacting with the operating system, such as reading file paths and handling directories. It played a vital role in automating data loading from nested folders containing multiple JSON files. In the PhonePe project, it was used to traverse directories and retrieve paths to all relevant data files. This helped in batch processing and reduced manual effort significantly. OS commands like os.listdir(), os.path.join(), and os.walk() streamlined the file handling process. It ensured that data from various categories and states was collected systematically. This automation supported scalability as more data could be added without changing the core logic. Overall, OS enhanced the efficiency and cleanliness of the data pipeline.

**6. Sklearn (Scikit-learn)**

Scikit-learn is one of the most popular machine learning libraries in Python, offering simple and efficient tools for data mining and analysis. In this project, it was used primarily for preprocessing tasks such as normalization, encoding, and possibly clustering or segmentation of users. Sklearn's StandardScaler or LabelEncoder functions helped prepare data for analysis or visual categorization. Though the core focus of the project was on visualization, Sklearn enabled deeper insights by structuring the data for potential prediction or classification tasks. Its wide range of models and utilities makes it an essential tool in any data science project. Integration with Pandas ensures a smooth workflow from raw data to analytical outcomes. Sklearn adds a layer of intelligence to PhonePe’s user and transaction insights.

**7. MySQL**

It is an open-source relational database management system (RDBMS) used to store, manage, and retrieve structured data efficiently. It organizes data into tables and allows the use of SQL (Structured Query Language) to perform operations like inserting, updating, querying, and deleting records. MySQL is widely used due to its speed, reliability, and ease of integration with various applications.

In this project, MySQL was used to store PhonePe’s cleaned data and perform queries for analysis. It supported operations like joins, aggregations, and filtering to extract meaningful insights. MySQL also ensures data consistency and supports indexing for faster data retrieval. Its compatibility with Python (via connectors) made it a key backend component in the data pipeline.

**MySQL Joins**

Joins in MySQL are used to combine records from two or more tables based on a related column, usually a primary-foreign key relationship. The most commonly used is the **INNER JOIN**, which returns only the matching rows from both tables. **LEFT JOIN** returns all rows from the left table and matched rows from the right, while **RIGHT JOIN** does the opposite. Joins are essential when the data is normalized across multiple tables. In this project, joins were used to bring together transaction details, user information, and geographical data into a single view. This allowed comprehensive analysis across multiple dimensions. Proper indexing and join conditions ensured efficient data retrieval.

**Aggregate Functions**

Aggregate functions in MySQL are used to perform calculations on sets of data and return a single summary value. Common functions include SUM() for totals, AVG() for averages, COUNT() for counting rows, and MAX()/MIN() for identifying extreme values. These functions are especially useful in reporting and analytics. In this project, aggregate functions were used to compute total transaction amounts, average insurance values, and user counts by state and category. They help convert raw numeric data into meaningful metrics for business decisions. Aggregate functions are typically used alongside GROUP BY to provide breakdowns across categories.

**GROUP BY Clause**

The GROUP BY clause groups rows that have the same values in specified columns into summary rows. It is used with aggregate functions to organize and summarize data. For example, grouping transaction data by state helps calculate state-wise totals or averages. In this project, GROUP BY enabled state-wise, quarter-wise, or category-wise insights for metrics like transaction volume, app usage, or insurance uptake.

It simplifies complex datasets and helps in trend identification across time or regions. Without GROUP BY, aggregate functions would summarize the entire table without segmentation.

**8. Visual Studio Code (VS Code)**

Visual Studio Code is a lightweight, open-source code editor developed by Microsoft. It supports multiple programming languages and provides features like syntax highlighting, IntelliSense (code suggestions), debugging, and integrated terminal. In this project, VS Code was used as the primary development environment for writing Python code, managing project files, and running Streamlit apps. It also supports extensions for Git, Python, Jupyter, and MySQL, making it a versatile tool for full-stack development. Its user-friendly interface and customization options enhance productivity and workflow efficiency.

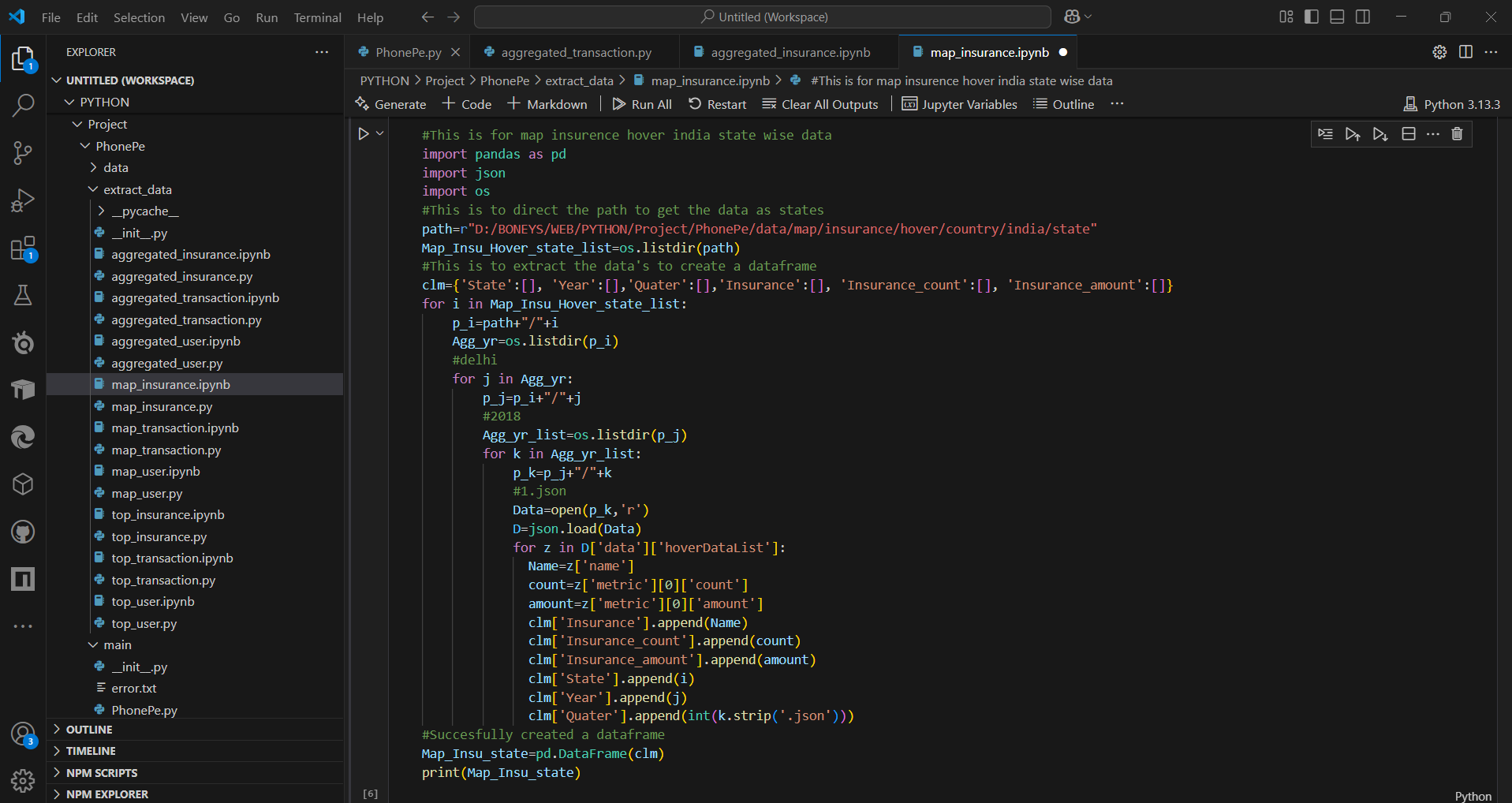
**9. MySQL Workbench (Optional)**

MySQL Workbench is a visual database design and administration tool provided by Oracle. It allows users to create, manage, and query MySQL databases through a graphical interface without writing complex SQL manually. In this project, it was optionally used to view table structures, execute queries, and monitor database performance. It simplifies data modeling and relationship mapping between tables. Although not mandatory, it provides a convenient way to interact with the MySQL database for those who prefer GUI over command-line tools.

# Methodology

**1. Data Collection:**  
 - The data for this project was collected from PhonePe’s publicly available datasets, which include transaction statistics, user registrations, device usage, and insurance data. These datasets are provided in structured JSON format, categorized by state, district, and transaction type. The data represents real, anonymized usage patterns across India. It served as the foundational input for all analysis and visualizations. The collection process involved downloading and organizing files for easy processing.

**2. Data Cleaning & Storage:**  
 - Once collected, the raw JSON data was parsed and cleaned using Python libraries like Pandas. This involved handling missing values, formatting numerical fields, standardizing column names, and converting nested data into tabular formats. After cleaning, the structured data was stored in a MySQL database for efficient querying and scalability. Using MySQL ensured that large volumes of state- and time-based data could be accessed quickly. This step prepared the data for further analysis and visualization.



**3. Data Visualization:**  
 - For the visualization phase, Streamlit and Plotly were used to build interactive dashboards. Streamlit allowed the creation of a simple web interface for users to filter, explore, and interact with data in real time. Plotly enabled the development of dynamic charts, maps, and graphs to display trends, comparisons, and patterns effectively. Together, these tools transformed raw data into intuitive visual insights. This made it easier to interpret the performance and user behavior on the PhonePe platform across different regions.

# Business Use Cases

**1. Customer Segmentation**

Customer segmentation involves dividing PhonePe users into distinct groups based on behaviour, transaction volume, device usage, or insurance adoption. In this project, segmentation helps identify high-value users, inactive users, or insurance-focused customers. For example, users frequently transacting via UPI and bill payments can be grouped as "power users" and targeted with premium features or cashback offers. Segmenting based on device types (e.g., Xiaomi vs. Apple users) also aids in optimizing app performance. This allows PhonePe to personalize marketing and service delivery for different user segments.

**2. Geographical Insights**

Geographical insights involve analysing user engagement, transaction trends, and insurance adoption across different states and districts. In this project, mapping and comparing metrics by location helped highlight high-performing regions like Maharashtra or Karnataka and underperforming ones like Assam or Meghalaya. For example, a state with low app opens despite high registrations may require regional language support or UPI education. These insights guide region-specific marketing strategies, partnerships, and product offerings to improve market penetration.

**3. Payment Performance Analysis**

This use case focuses on understanding how different payment modes (e.g., P2P, merchant payments, bill payments) are performing. The project visualized payment volume, frequency, and value trends across regions and time. For instance, a surge in merchant transactions during a festival season can indicate increased platform reliance. Identifying high-usage categories helps PhonePe optimize infrastructure and offer targeted promotions or partnerships with retail chains. It also reveals user preferences and emerging payment trends in different areas.

**4. Insurance Insights**

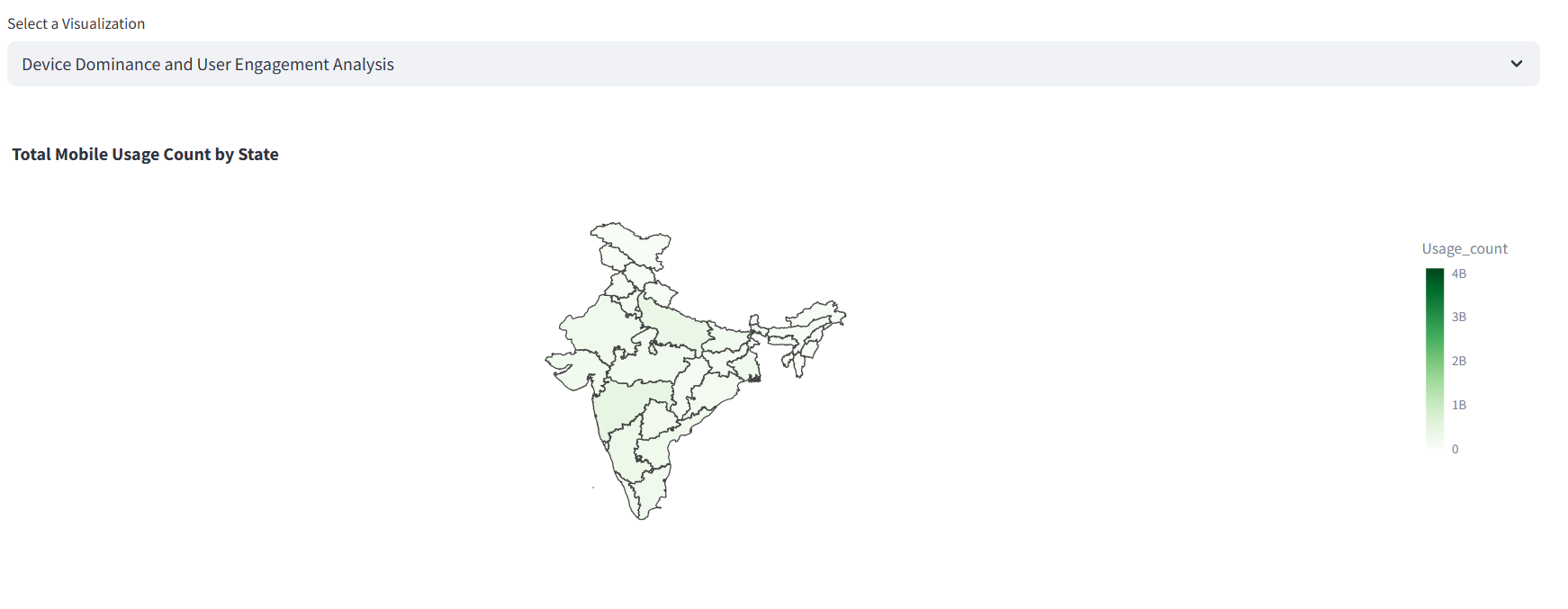
Insurance insights help PhonePe evaluate how users adopt various insurance products offered on the platform. This project analysed state-wise growth in insurance users and investment values. For example, states like Karnataka showed strong insurance engagement, while eastern states lagged behind. Such insights allow PhonePe to design affordable micro-insurance plans or campaigns using local influencers. This is vital for tapping into the underinsured population and driving financial inclusion through digital platforms.

**5. Trend and Growth Analysis**

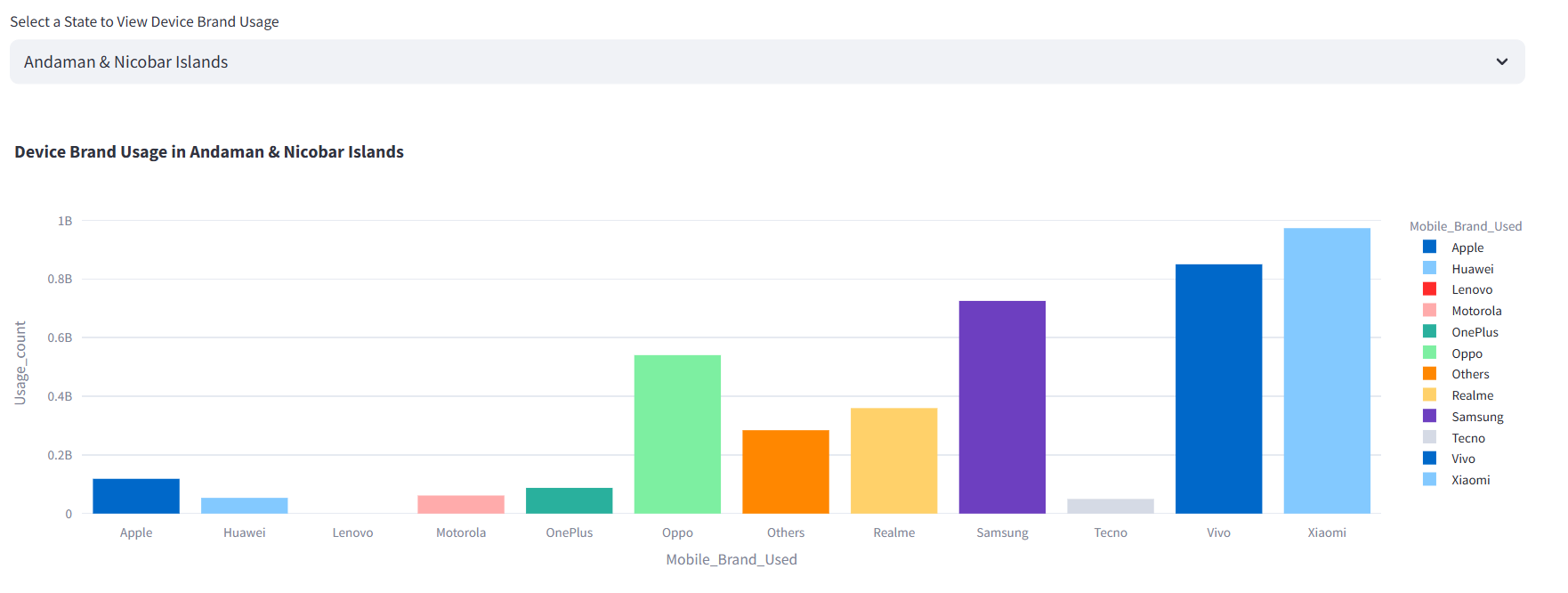
Trend and growth analysis involves tracking how key metrics like transactions, users, and insurance adoption evolve over time. This project compared quarterly data to highlight growth surges or stagnation phases, such as a spike after 2020 Q4. For example, a steady rise in bill payments over a year may suggest increased user reliance on PhonePe for recurring utilities. These trends help forecast future demand, guide business planning, and assess the impact of past strategies. Growth patterns also identify emerging opportunities and areas requiring improvement.

# Detailed Analysis & Insights

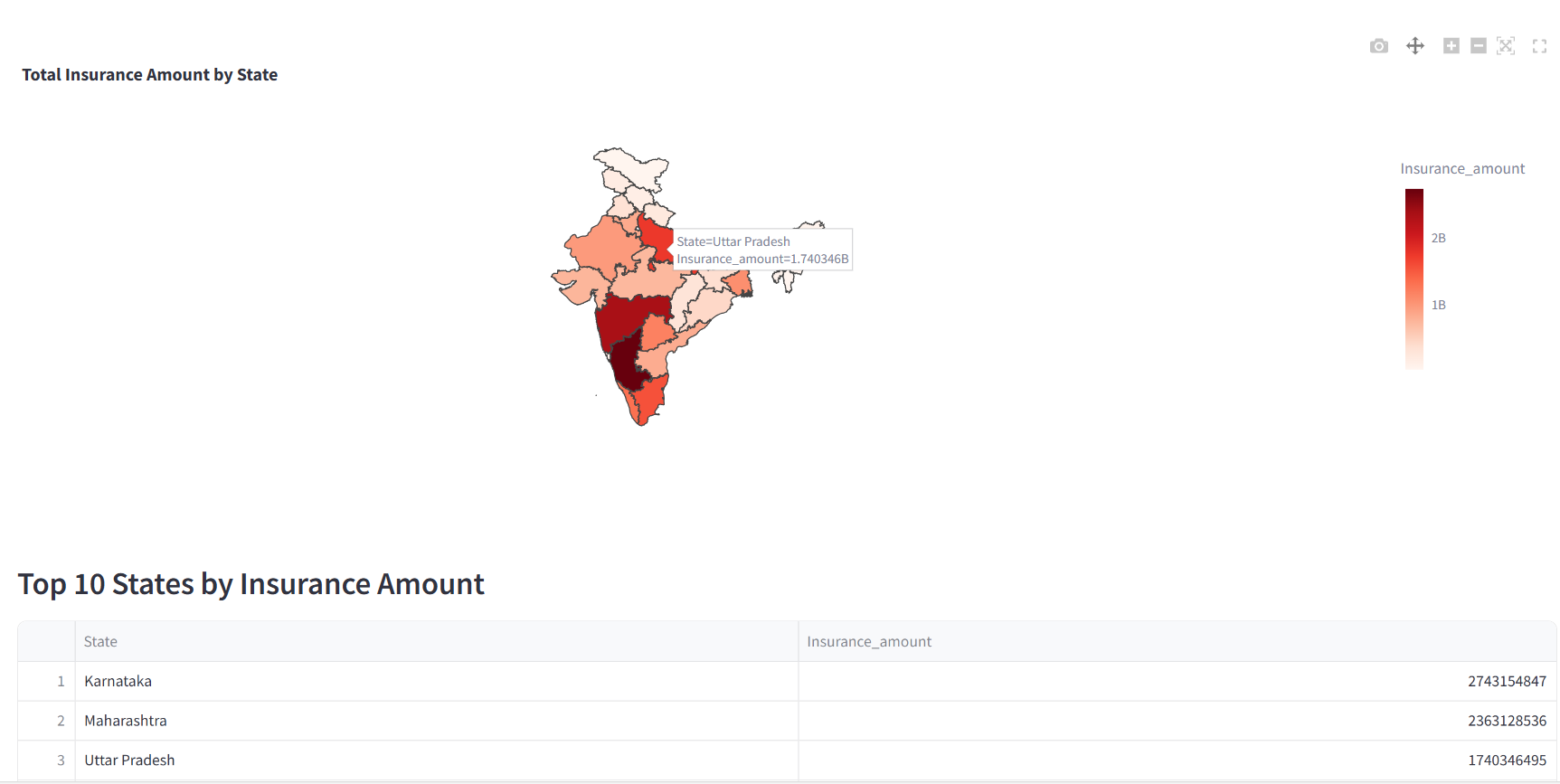
### ****Device Dominance and User Engagement Analysis****



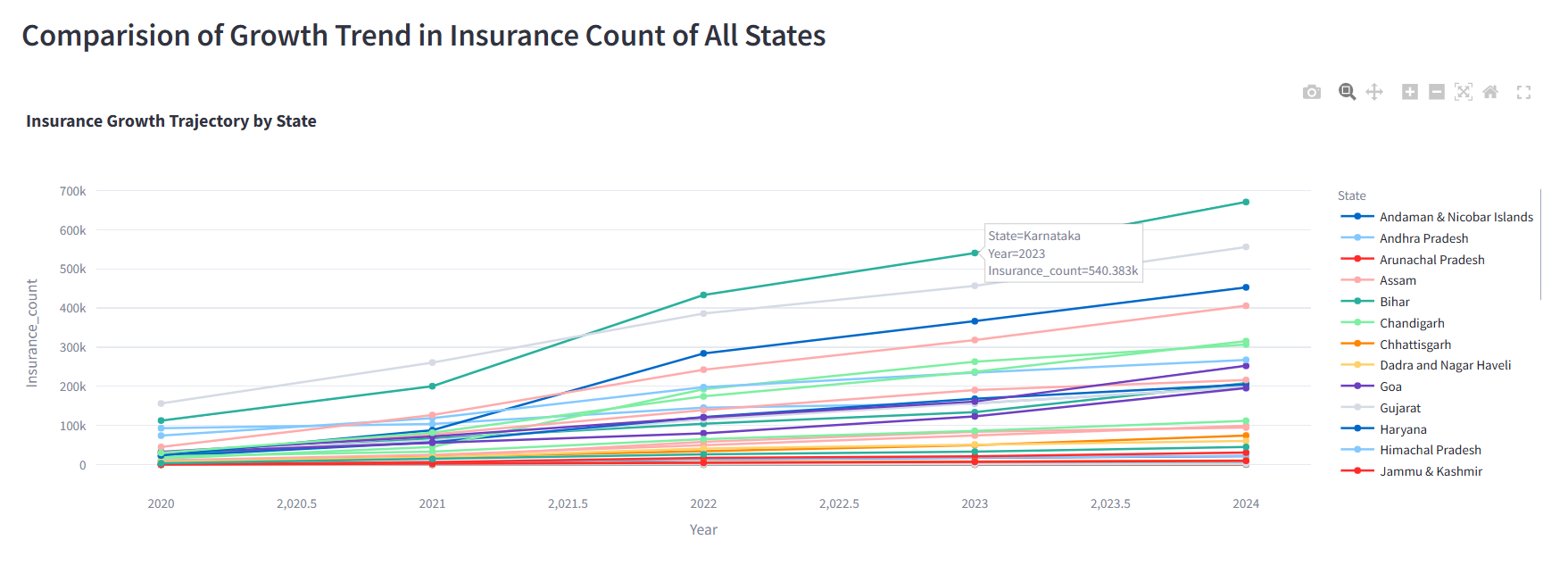
The analysis revealed that Xiaomi leads in device usage among PhonePe users, while Apple and Huawei show high registration but low app engagement. This suggests performance or experience gaps on those devices. Low-engagement states like Assam and Meghalaya may lack awareness or accessibility. To address this, PhonePe should optimize its app for less engaged brands and run localized UPI awareness campaigns. Partnering with OEMs for pre-installation offers can also help boost engagement in these regions.



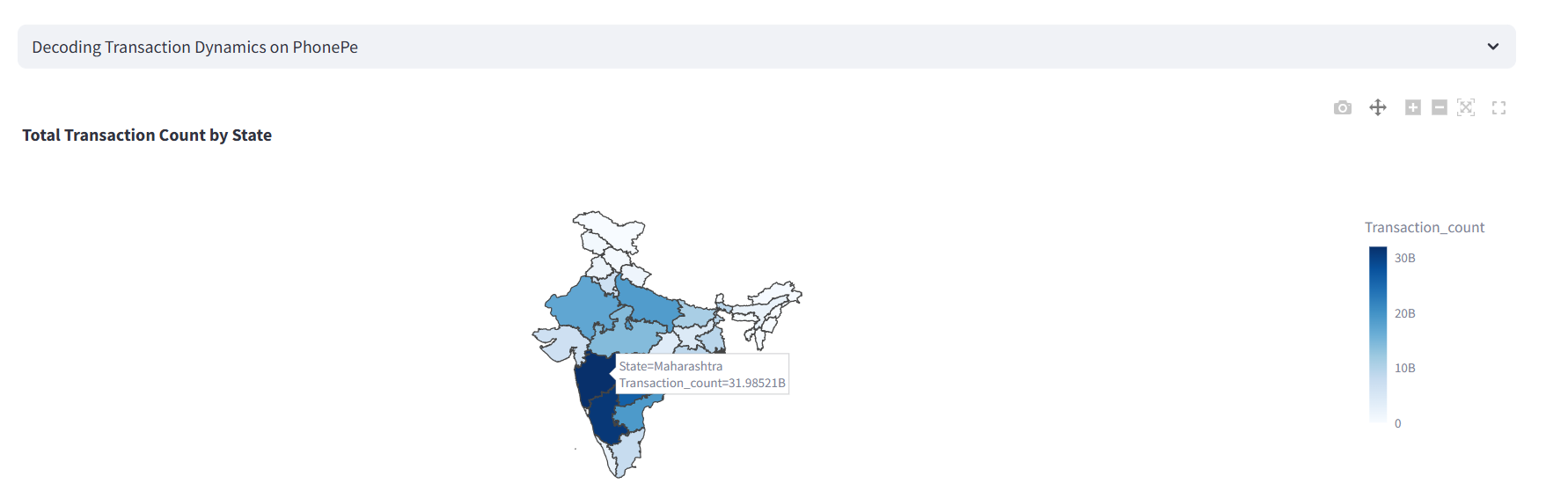
### ****Insurance Penetration and Growth Potential Analysis****



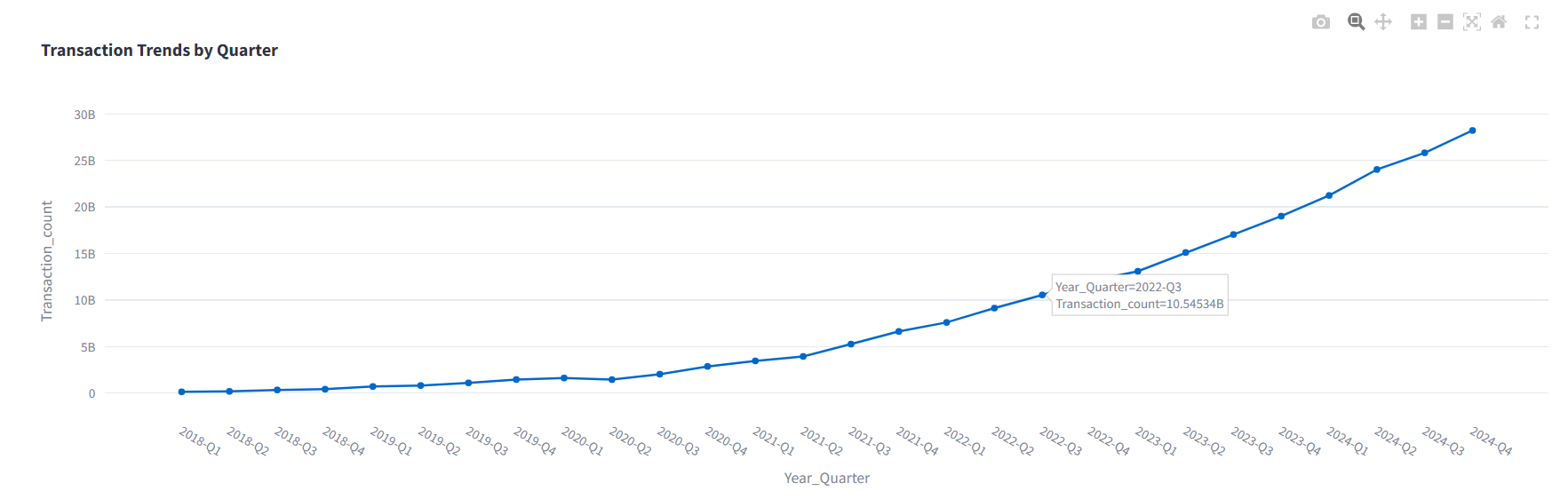
There are noticeable disparities in insurance adoption across Indian states, with western and southern states showing higher uptake. Eastern and northeastern states lag, indicating a need for region-specific strategies. Awareness campaigns driven by local influencers and community outreach can build trust in these markets. Additionally, introducing low-cost, bundled insurance options during routine PhonePe transactions could encourage first-time adoption and enhance financial protection for users.



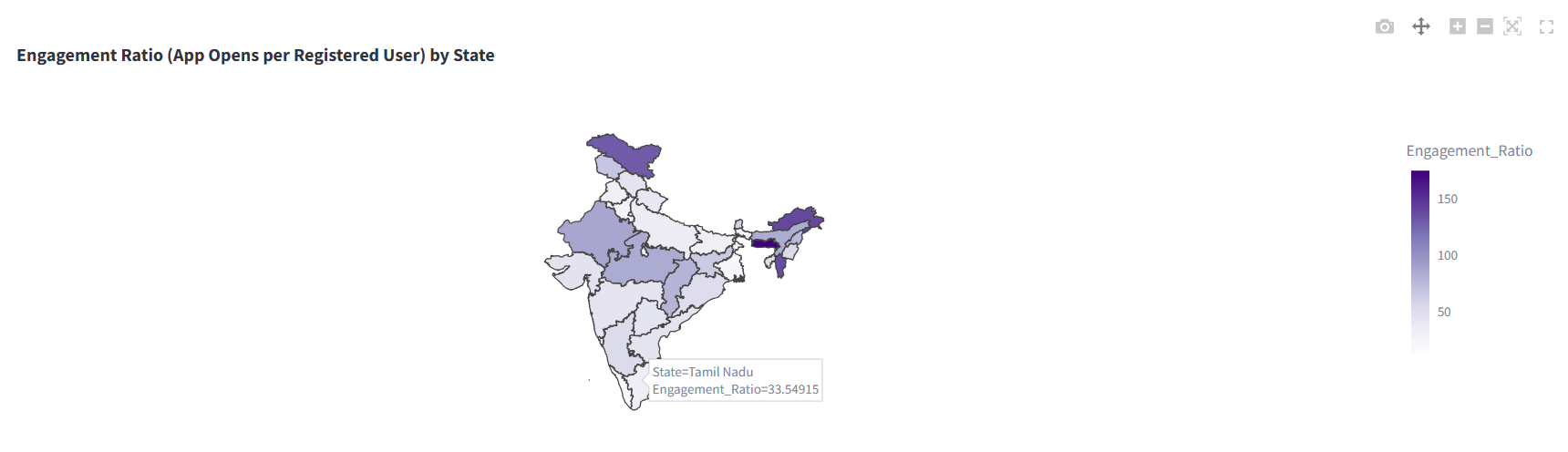
### ****Decoding Transaction Dynamics on PhonePe****



The data highlights that merchant and peer-to-peer (P2P) transactions dominate PhonePe’s ecosystem, showing high user trust and utility. A significant rise in usage is observed after Q4 of 2020, suggesting digital adoption post-COVID. Collaborations with local banks or co-operatives can further strengthen platform trust. Introducing combo offers for recharges and bill payments can enhance user convenience and transaction frequency, supporting growth in both mature and emerging markets.

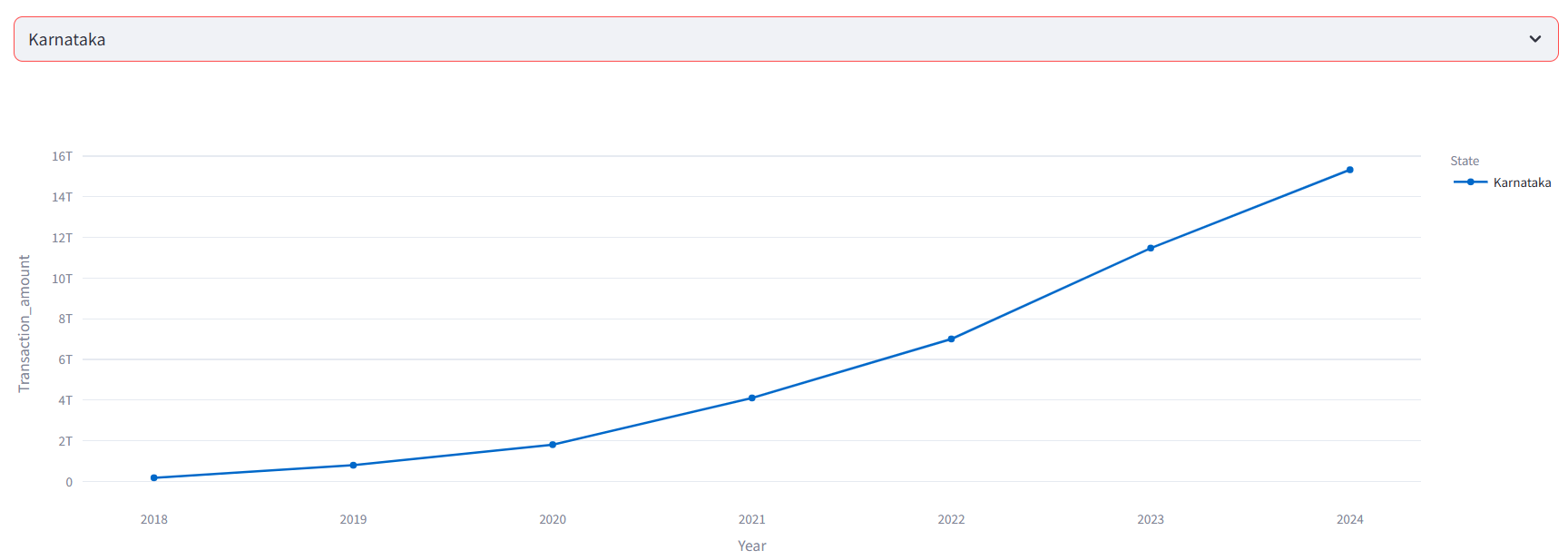


### ****User Engagement and Growth Strategy****

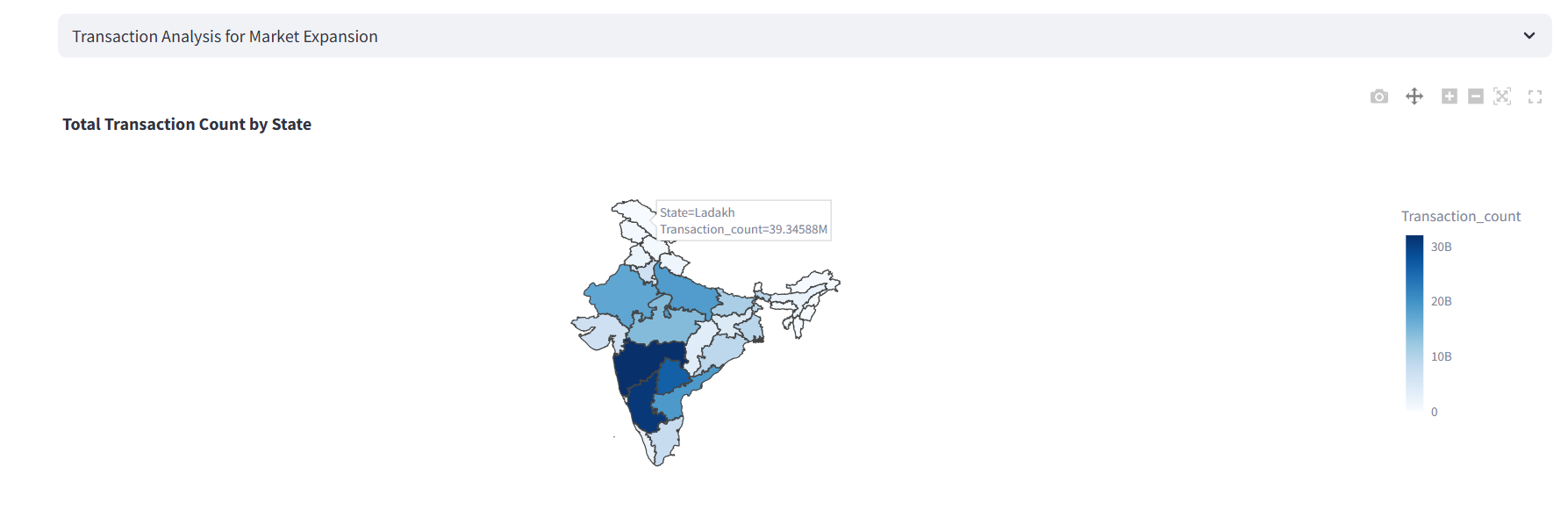


User engagement varied widely across states, with some regions showing high app usage per user and others underperforming. Low-engagement states often need regional language support, better network optimization, and personalized offers to drive activity. High-engagement states offer opportunities for deeper marketing and partnerships. Improving app performance and targeted communication can help PhonePe convert registrations into active usage, expanding its footprint more effectively.

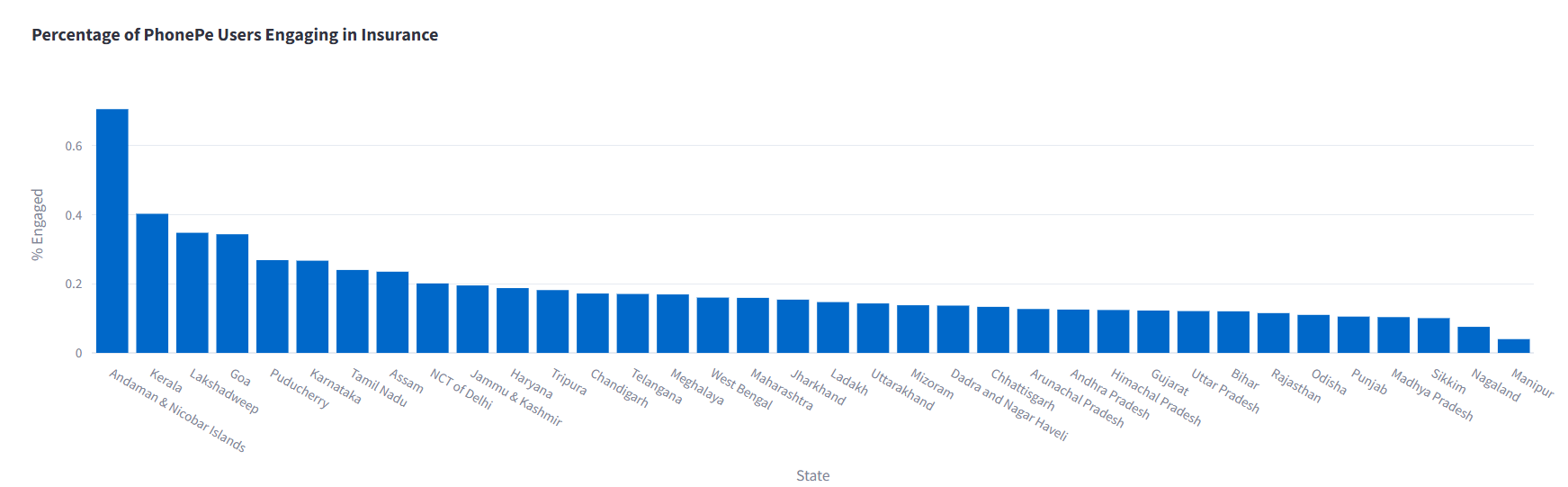
### ****Transaction Analysis for Market Expansion****



PhonePe’s transaction volume has shown a consistent rise since late 2020, particularly in digitally active states. However, dormant users remain untapped. Targeted push notifications, seasonal offers, and loyalty rewards can reactivate these users. Expanding into services like school fee payments, tax payments, and government utilities can further increase daily app usage. Promoting non-UPI services such as gold investment and credit tracking can diversify PhonePe’s product offering and appeal.



### ****Insurance Engagement Analysis****



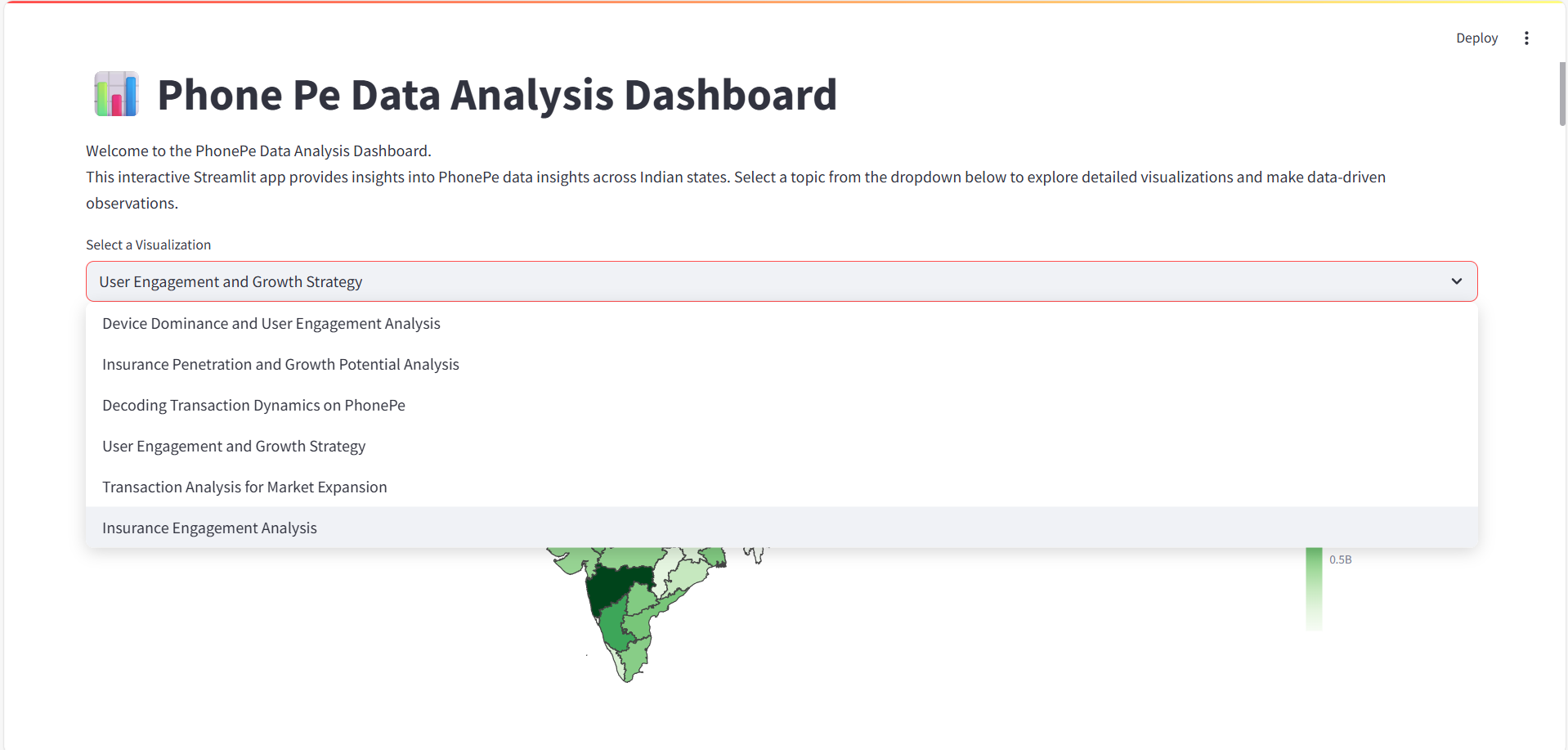
States like Maharashtra and Karnataka dominate high-value insurance transactions, while populous states like Uttar Pradesh and Bihar show low engagement despite large user bases. This gap highlights the need for simplified insurance onboarding, especially for first-time users. PhonePe can boost adoption by offering cashback, vouchers, or limited-period discounts on insurance products. Campaigns focused on short-term, low-premium policies may help reach broader, price-sensitive audiences in low-engagement regions.

# Results

The visualizations and data analysis show:  
- Diverse trends in device usage and engagement.  
- Sharp contrasts in insurance adoption and digital payment maturity across states.  
- Transaction growth even during the pandemic, indicating strong user trust.

# Actions & Recommendations

- Boost regional adoption with targeted offers.  
- Collaborate with OEMs, local influencers, and governments.  
- Expand services to untapped regions and demographics.  
- Ensure device optimization and local language support.



# Conclusion

This project provided a data-driven exploration of PhonePe’s performance across various dimensions such as user engagement, device usage, transaction patterns, and insurance adoption. By leveraging Python, MySQL, Streamlit, and data visualization tools, we converted large-scale raw data into structured, insightful views. The state-wise and district-wise analyses uncovered significant variations in digital payment adoption, highlighting how user behaviour, device preference, and regional dynamics influence engagement. The findings demonstrate that while states like Maharashtra and Karnataka are mature markets, others like Assam and Bihar show untapped potential that can be unlocked through targeted efforts.

Insights from the analysis point toward specific opportunities for PhonePe to strengthen its market position. Device analysis revealed that despite high registrations, brands like Apple and Huawei see lower app usage—indicating the need for technical optimization and better UX. Insurance trends showed growth potential in underserved states, suggesting a need for affordable policies and strong awareness campaigns. Additionally, understanding transaction dynamics showed a dominance of merchant and peer-to-peer payments, with a post-2020 surge emphasizing the increasing digital trust. Identifying low-engagement regions and dormant users opens the door for personalized marketing and deeper user reactivation strategies.

The project lays the groundwork for further strategic expansion by emphasizing data-backed decision-making. Future work could include integrating real-time analytics, predictive modeling (e.g., churn prediction), and clustering users for hyper-personalized services. PhonePe can benefit from local partnerships, pre-installation collaborations with OEMs, and multilingual support to boost inclusivity. Overall, the analysis highlights the platform’s strong foundation and the importance of tailoring strategies to user behavior, regional characteristics, and product demand. With ongoing refinement and targeted outreach, PhonePe has a clear path to further solidify its leadership in India’s digital payment and financial services space.

# Future Scope

- Incorporate real-time data analytics.  
- Develop AI models for user behavior prediction.  
- Expand to pan-India services like e-governance, education payments, and microloans.  
- Study competitive platforms for comparative insights.