

# PhonePe Pulse Data Analysis

SUBMITTED BY

HRITURAJ SAHA



**Data Science with AI/ML Internship**

**Duration - 2 Month**

## 1) INTRODUCTION

The digital payment landscape in India has experienced exponential growth over the last few years, led by Unified Payments Interface (UPI) platforms like PhonePe. To understand this transformation, PhonePe has released an open dataset called PhonePe Pulse, which contains detailed quarterly information on transaction types, volumes, app usage and device distribution.

This report documents the process of extracting, storing and analyzing this dataset using Python, SQLite and Streamlit. It provides insights into user behavior, growth patterns and the adoption of digital payments across India.

## 2) OBJECTIVES

The primary goals of this project are -

- To analyze and visualize trends in digital transactions and user behavior from 2018 to 2023.
- To explore transaction types, volumes, app opens, registered users and device usage patterns.
- To develop an interactive Streamlit dashboard for dynamic insights.
- To present clear and concise findings that can guide product decisions and fintech research.

## 3) DATA SOURCE AND METHODOLOGY

### 1) Data Source -

- Source - PhonePe Pulse Repository (<https://github.com/PhonePe/pulse>).
- Format - JSON files organized by category (transaction, user), year, and quarter.
- Granularity - Aggregated at the country level (India).

### 2) Data Extraction and Storage -

A custom Python script was developed to extract relevant data from the JSON files. The data was cleaned and structured into three main SQLite tables -

Table - *aggregated\_transaction*

Column	Type
state	TEXT
year	INTEGER
quarter	INTEGER
transaction_type	TEXT
count	INTEGER
amount	REAL

Table - *aggregated\_user\_summary*

Column	Type
state	TEXT
year	INTEGER
quarter	INTEGER
registered_users	INTEGER
app_opens	INTEGER

Table - *aggregated\_user\_by\_device*

Column	Type
state	TEXT
year	INTEGER
quarter	INTEGER
brand	TEXT
user_count	INTEGER
percentage	REAL

## 4) TOOLS AND TECHNOLOGIES

- Programming Language - Python.
- Database - SQLite.
- Visualization - Plotly Express.
- Web App - Streamlit.
- Deployment - Google Colab with pyngrok tunneling.
- Version Control - Git & GitHub.

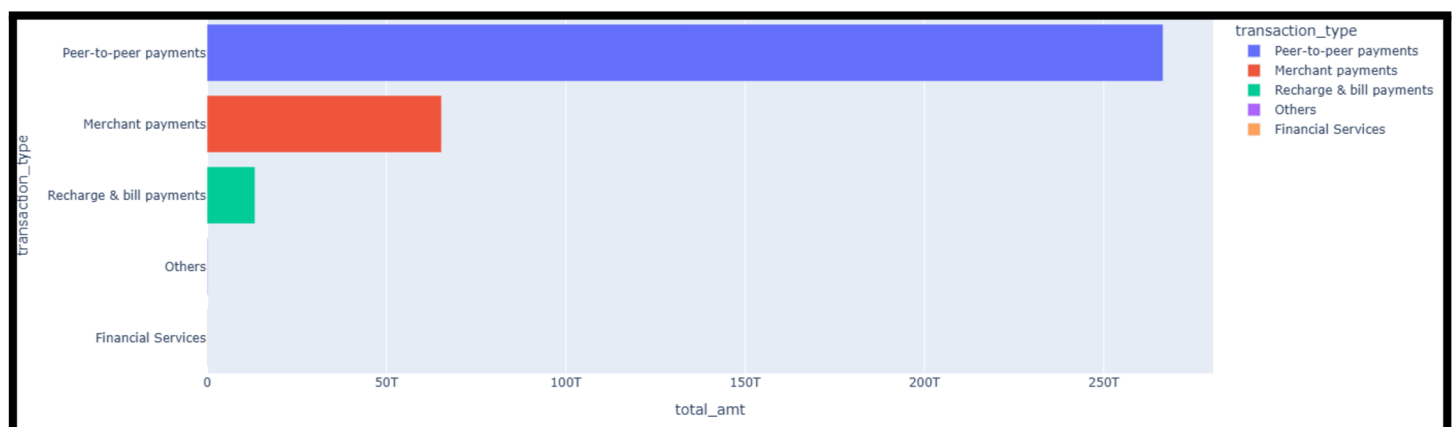
## 5) KEY FINDINGS AND ANALYSIS

### 1) Top Transaction Types -

The five most common transaction types (by total amount) are -

- Peer-to-peer payments
- Recharges and bill payments
- Merchant payments
- Financial services
- Others

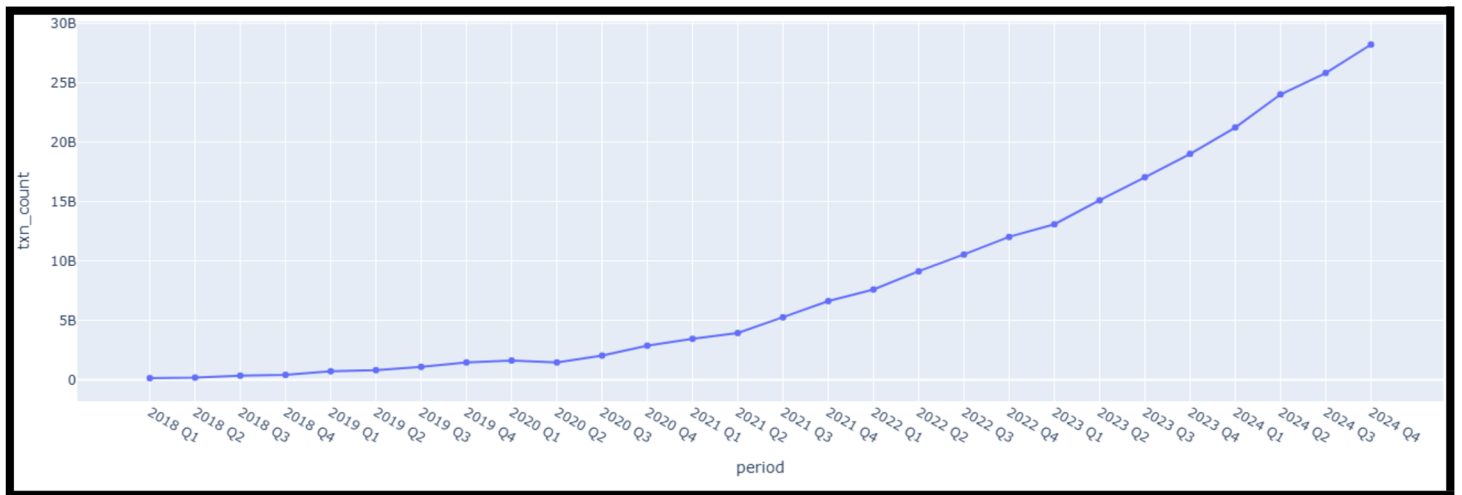
Insight - Peer-to-peer transactions constitute the largest share of the total transaction volume, indicating that UPI is predominantly used for personal transfers.



### 2) Transaction Volume Over Time -

The total number of transactions has shown consistent growth quarter-over-quarter, with a particularly steep rise post-2020.

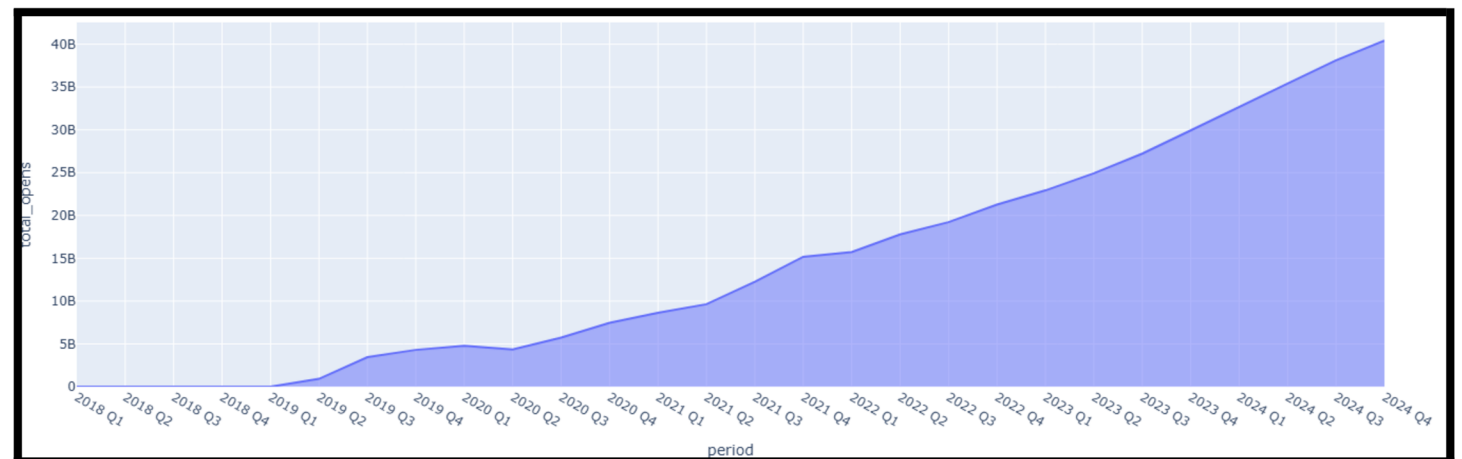
Insight - The COVID-19 pandemic accelerated digital adoption. By Q4 2023, the volume reached its highest level since 2018.



### 3) App Opens Per Quarter -

The number of app opens per quarter increased dramatically from under 5 billion to over 40 billion.

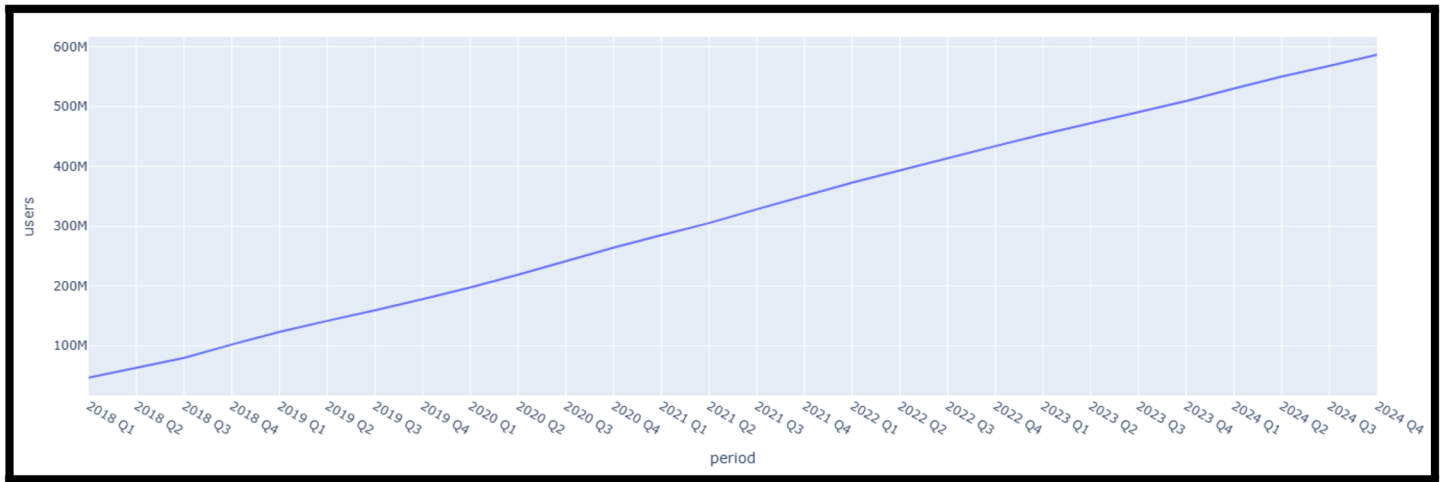
Insight - Increased user engagement correlates with promotional campaigns and user acquisition strategies.



### 4) Registered Users Over Time -

The number of registered users saw massive growth from a few million in early quarters to over 400 million by the end of 2023.

Insight - Aggressive market penetration and mobile-first financial access have driven user adoption, especially in tier-2 and tier-3 cities.



## 5) Top Devices Used -

The majority of users access PhonePe on -

- Xiaomi
- Samsung
- Vivo
- Oppo
- Realme

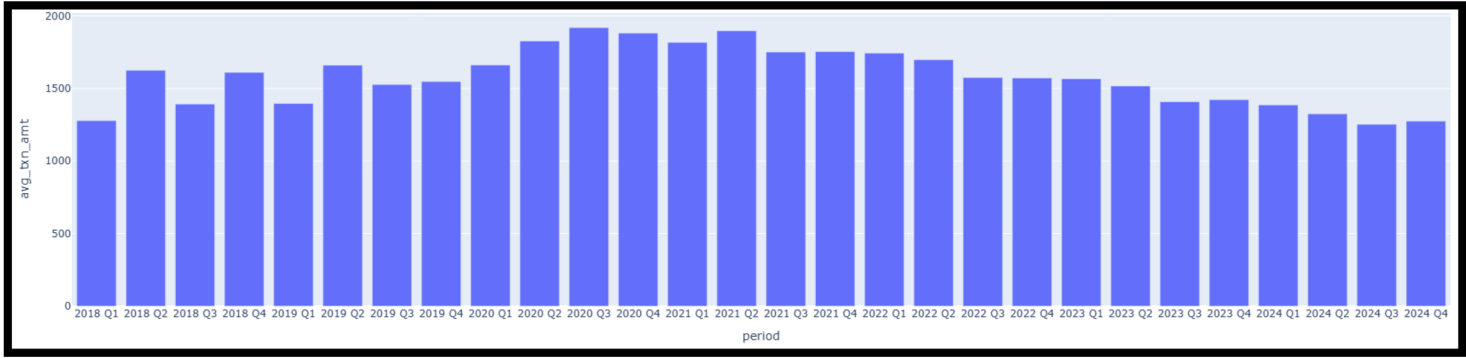
Insight - Budget Android smartphones dominate the digital payment ecosystem, emphasizing affordability and accessibility.



## 6) Average Transaction Amount Over Time -

The average transaction amount per quarter hovers between ₹1,000–₹1,600.

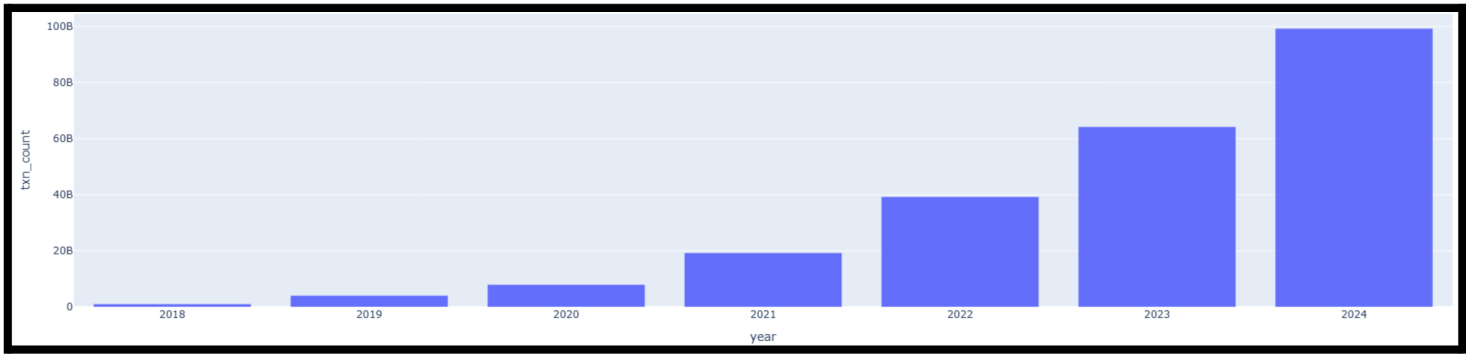
Insight - Indicates consistent use for routine transactions like bill payments, groceries, and peer-to-peer transfers.



**7) Most Active Transaction Years -**

2024 recorded the highest number of digital transactions to date, followed by 2023.

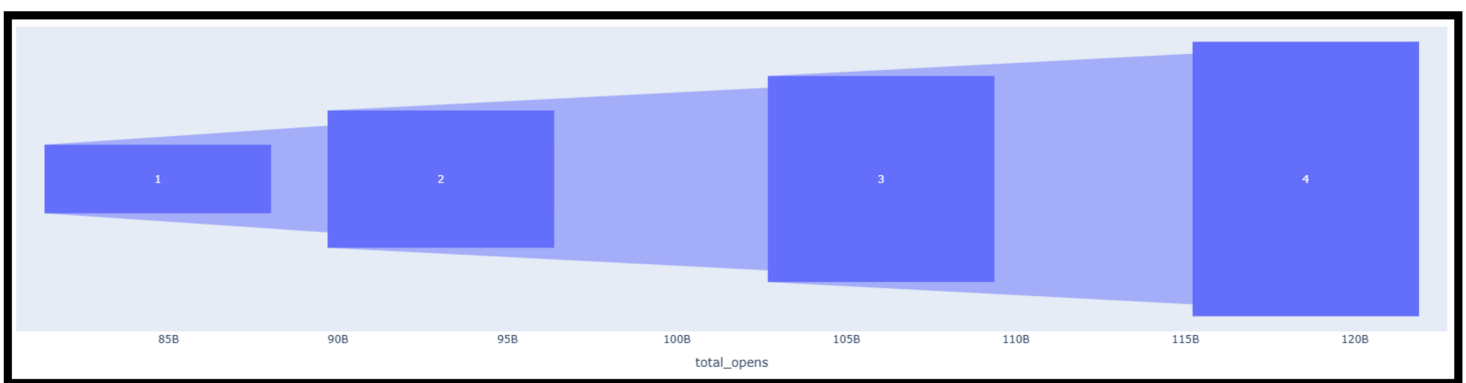
Insight - The digital payment economy is scaling consistently, with no signs of decline, reflecting macroeconomic digitalization efforts.



**8) Most Popular Quarters for App Opens -**

Q4 (October to December) consistently has the highest app open rates across years.

Insight - Festive seasons like Diwali and end-of-year bonuses drive consumer spending and app activity.



# 6) CODE IMPLEMENTATION

## 1) Visualization -

```
PhonePe.ipynb
File Edit View Insert Runtime Tools Help
Q Commands + Code + Text Run all
# Step 4 - Generate Visualizations (8 Key Findings)
import pandas as pd
import plotly.express as px
import sqlite3
conn = sqlite3.connect("phonepe.db")

# 1. Top Transaction Types
df1 = pd.read_sql("""
    SELECT transaction_type, SUM(amount) as total_amt
    FROM aggregated_transaction GROUP BY transaction_type
    ORDER BY total_amt DESC LIMIT 5
""", conn)
fig1 = px.bar(df1, x="total_amt", y="transaction_type", orientation="h",
              color="transaction_type", title="Top Transaction Types")
fig1.show()

# 2. Transaction Volume Over Time
df2 = pd.read_sql("""
    SELECT year, quarter, SUM(count) as txn_count
    FROM aggregated_transaction GROUP BY year, quarter
    ORDER BY year, quarter
""", conn)
df2['period'] = df2['year'].astype(str) + ' Q' + df2['quarter'].astype(str)
fig2 = px.line(df2, x="period", y="txn_count", markers=True,
              title="Transaction Volume Over Time")
fig2.show()

# 3. App Opens Per Quarter
df3 = pd.read_sql("""
    SELECT year, quarter, SUM(app_opens) as total_opens
    FROM aggregated_user_summary GROUP BY year, quarter
    ORDER BY year, quarter
""", conn)
df3['period'] = df3['year'].astype(str) + ' Q' + df3['quarter'].astype(str)
fig3 = px.area(df3, x="period", y="total_opens", title="App Opens Per Quarter")
fig3.show()

# 4. Registered Users Over Time
df4 = pd.read_sql("""
    SELECT year, quarter, SUM(registered_users) as users
    FROM aggregated_user_summary GROUP BY year, quarter
    ORDER BY year, quarter
""", conn)
df4['period'] = df4['year'].astype(str) + ' Q' + df4['quarter'].astype(str)
fig4 = px.line(df4, x="period", y="users", title="Registered Users Over Time")
fig4.show()

# 5. Top Devices Used
df5 = pd.read_sql("""
    SELECT brand, SUM(user_count) as total_users
    FROM aggregated_user_by_device GROUP BY brand
    ORDER BY total_users DESC LIMIT 5
""", conn)
fig5 = px.pie(df5, names="brand", values="total_users", title="Top 5 Devices Used")
fig5.show()
```

## 2) Streamlit -

```
PhonePe.ipynb
File Edit View Insert Runtime Tools Help
Q Commands + Code + Text Run all
# Step - 5 Streamlit App Creation
app_code = """
import streamlit as st
import pandas as pd
import sqlite3
import plotly.express as px
conn = sqlite3.connect("phonepe.db")
st.set_page_config(page_title="PhonePe Insights", layout="wide")
st.title("PhonePe Transaction Insights Dashboard")
st.subheader("8 Key Findings with Interactive Charts")

# 1. Top Transaction Types
df1 = pd.read_sql("SELECT transaction_type, SUM(amount) as total_amt FROM aggregated_transaction GROUP BY transaction_type ORDER BY total_amt DESC LIMIT 5", conn)
fig1 = px.bar(df1, x="transaction_type", y="total_amt", title="Top Transaction Types", color="transaction_type")
st.plotly_chart(fig1, use_container_width=True)

# 2. Transaction Volume Over Time
df2 = pd.read_sql("SELECT year, quarter, SUM(count) as txn_count FROM aggregated_transaction GROUP BY year, quarter ORDER BY year, quarter", conn)
df2['period'] = df2['year'].astype(str) + ' Q' + df2['quarter'].astype(str)
fig2 = px.line(df2, x="period", y="txn_count", title="Transaction Volume Over Time", markers=True)
st.plotly_chart(fig2, use_container_width=True)

# 3. App Opens Per Quarter
df3 = pd.read_sql("SELECT year, quarter, SUM(app_opens) as total_opens FROM aggregated_user_summary GROUP BY year, quarter ORDER BY year, quarter", conn)
df3['period'] = df3['year'].astype(str) + ' Q' + df3['quarter'].astype(str)
fig3 = px.area(df3, x="period", y="total_opens", title="App Opens Per Quarter")
st.plotly_chart(fig3, use_container_width=True)

# 4. Registered Users Over Time
df4 = pd.read_sql("SELECT year, quarter, SUM(registered_users) as users FROM aggregated_user_summary GROUP BY year, quarter ORDER BY year, quarter", conn)
df4['period'] = df4['year'].astype(str) + ' Q' + df4['quarter'].astype(str)
fig4 = px.line(df4, x="period", y="users", title="Registered Users Over Time")
st.plotly_chart(fig4, use_container_width=True)

# 5. Top Devices Used
df5 = pd.read_sql("SELECT brand, SUM(user_count) as total_users FROM aggregated_user_by_device GROUP BY brand ORDER BY total_users DESC LIMIT 5", conn)
fig5 = px.pie(df5, names="brand", values="total_users", title="Top 5 Devices Used")
st.plotly_chart(fig5, use_container_width=True)

# 6. Average Transaction Amount Over Time
df6 = pd.read_sql("SELECT year, quarter, SUM(amount)*1.0/SUM(count) as avg_amt FROM aggregated_transaction GROUP BY year, quarter ORDER BY year, quarter", conn)
df6['period'] = df6['year'].astype(str) + ' Q' + df6['quarter'].astype(str)
fig6 = px.bar(df6, x="period", y="avg_amt", title="Avg. Transaction Amount Over Time")
st.plotly_chart(fig6, use_container_width=True)

# 7. Most Active Transaction Years
df7 = pd.read_sql("SELECT year, SUM(count) as txn_count FROM aggregated_transaction GROUP BY year ORDER BY txn_count DESC", conn)
fig7 = px.bar(df7, x="year", y="txn_count", title="Most Active Transaction Years")
st.plotly_chart(fig7, use_container_width=True)

# 8. Most Popular Quarters for App Opens
df8 = pd.read_sql("SELECT quarter, SUM(app_opens) as total_opens FROM aggregated_user_summary GROUP BY quarter ORDER BY total_opens DESC", conn)
fig8 = px.bar(df8, x="quarter", y="total_opens", title="Most Popular Quarters for App Opens")
st.plotly_chart(fig8, use_container_width=True)
st.success("Dashboard loaded Successfully!")
...
with open("app.py", "w") as f:
    f.write(app_code)
"""
```



## 7) FUTURE WORK

🔧 Potential enhancements -

- 📍 Add state-wise or city-level analysis.
- 🔄 Add category-wise merchant insights.
- 🧠 Train ML models for transaction forecasting.
- 📊 Add filters (year, type) in Streamlit for dynamic interaction.

## 8) CONCLUSION

This analysis demonstrates the transformative impact of UPI platforms like PhonePe on India's digital economy. The growth in user base, steady transaction volumes, and regional device penetration tell a compelling story of digital inclusion.

Key Takeaways -

- UPI is India's default mode for both peer and commercial transactions.
- User engagement has grown 10x since 2019.
- Budget smartphones play a critical role in digital payment adoption.