

## BAMBASWAP KPI ANALYSIS

Period: October 2024

# Introduction

---

This report provides an in-depth analysis of Bambaswap's key performance indicators (KPIs) from January to October 2024, focusing on Customer Acquisition Rate (CAR), Monthly Recurring Revenue (MRR), Retention Rate, and Customer Lifetime Value (CLV).

By examining these KPIs, we aim to understand customer growth trends, revenue sustainability, retention challenges, and overall business health. A particular focus is placed on October 2024, where significant shifts in these metrics highlight critical business challenges.

This analysis is done using the PostgreSQL and you can find all the queries here: [Bambaswap October Analysis 2024](#)

## Tools Used

---

For my deep dive into the SwapFintech data analysis and KPI's, I harnessed the power of several key tools:

**SQL:** The backbone of my analysis, allowing me to query the database and unearth critical insights.

**PostgreSQL:** The chosen database management system, ideal for handling the SwapFintech data.

**Visual Studio Code:** My go-to for database management and executing SQL queries.

**Power BI:** Essential for visuals utilized during the analysis.

**Git & GitHub:** Essential for version control and sharing my SQL scripts and analysis, ensuring collaborations and project tracking.

## The Analysis

---

Each query for this project aimed at investigating specific aspect of the SwapFintech KPI's. KPIs identified for the Swap Fintech to track progress and drive business growth include:-

- **Customer Acquisition Rate (C.A.R)**
- **Cusomer Conversion Rate (C.C.R)**
- **Monthly Recurring Revenue (M.R.R)**
- **Monthly Retention Rate (R.R)**
- **Churn Rate (C.R)**
- **Customer Lifetime Value (C.L.V)**

### 1. Customer Acquisition Rate (C.A.R)

**Customer Acquisition Rate (CAR)** The **Customer Acquisition Rate** measure how many new customers a BambaSwap has gained over a period relative to its existing customer base. In our case, we calculate **CAR**

month by month from January to September as historical data and focusing on October in this report.

To calculate the **CAR** we utilized the formula:-

$$\text{CAR}_{\{m\}} = \frac{\text{Returning Customers Before Month } m}{\text{New Customers in Month } m} \times 100$$

Where;-

- **New Customers in Month m** : Customer who appeared in month m (**October**) but was not a returning customer.

Identified by checking if the phone\_number exists in October but (do not exist) in earlier data. (Jan to Sep 2024)

- **Returning Customers** : BambaSwap customers who have made transactions at least 3 times before October (January to September).

This ensures that we only consider customers who have shown consistent engagement.

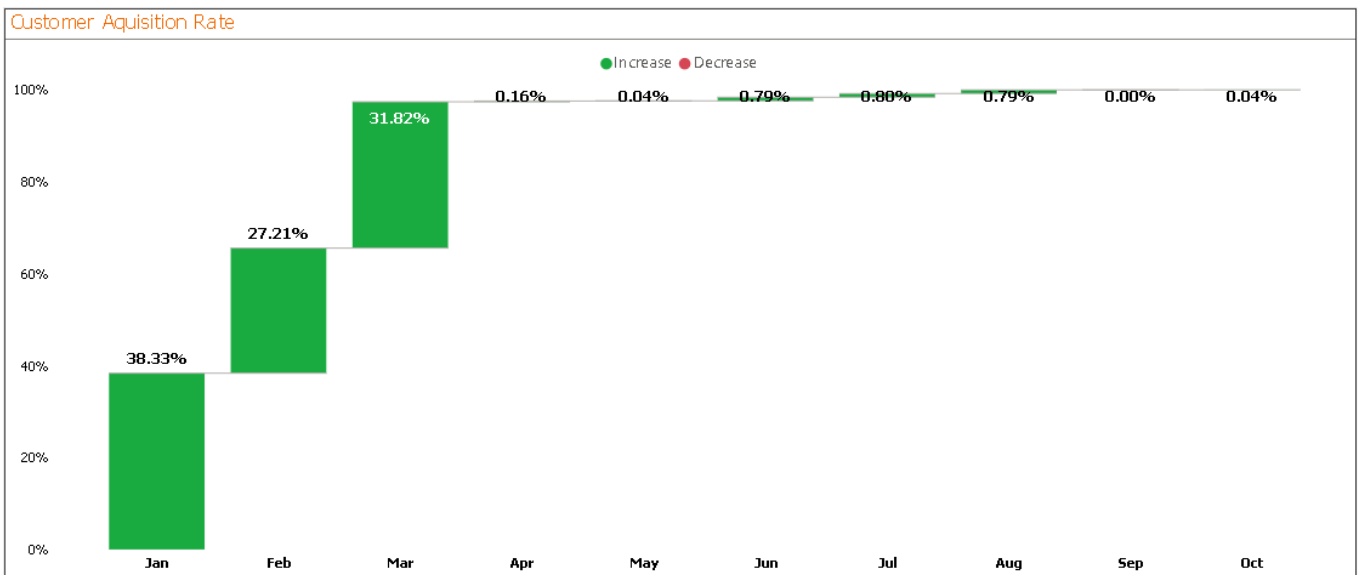
sql query used :-

```
WITH customer_counts AS (
    -- Count occurrences of each customer per month
    SELECT phone_number, month, month_number, COUNT(*) AS occurrences
    FROM bambaswap_combined_jan_oct_2024
    GROUP BY phone_number, month, month_number
),
cumulative_customers AS (
    -- Track when a customer first appeared and their total count before each month
    SELECT c1.phone_number, c1.month, c1.month_number,
        (SELECT COUNT(*)
         FROM customer_counts c2
         WHERE c2.phone_number = c1.phone_number
         AND c2.month_number < c1.month_number) AS prior_count
    FROM customer_counts c1
),
returning_customers AS (
    -- Identify customers who had 3+ occurrences before a given month
    SELECT phone_number, month, month_number
    FROM cumulative_customers
    WHERE prior_count >= 3
),
monthly_customers AS (
    -- Get unique customers per month
    SELECT DISTINCT phone_number, month, month_number
    FROM bambaswap_combined_jan_oct_2024
),
new_customers AS (
    -- Find new customers (not in returning customers for that month)
    SELECT m.phone_number, m.month, m.month_number
    FROM monthly_customers m
```

```

LEFT JOIN returning_customers r
ON m.phone_number = r.phone_number AND m.month = r.month
WHERE r.phone_number IS NULL
)
-- Calculate CAR per month and replace NULL values
SELECT
    m.month,
    m.month_number,
    -- Ensure no NULL values by using COALESCE and GREATEST
    COALESCE((COUNT(n.phone_number) * 100.0 / GREATEST(COUNT(r.phone_number), 1)),
0) AS acquisition_rate
FROM monthly_customers m
LEFT JOIN new_customers n ON m.phone_number = n.phone_number AND m.month = n.month
LEFT JOIN returning_customers r ON m.phone_number = r.phone_number AND m.month =
r.month
GROUP BY m.month, m.month_number
ORDER BY m.month_number;

```



A powerBI generated waterfall graph showing the monthly changes in the average aquisition rates for SwapFintech in the month of October.

From the analysis and graph respresentation, the following conclusions can be drawn.

### 1 Drastic Drop Compared to Early Months

- In **January and February**, CAR was extremely high (521,000% and 369,800%), likely due to a surge in new customers.
- By **October**, CAR fell to 591.2%, indicating a much lower rate of new customer acquisition compared to the earlier months.

### 2 Steady CAR in Mid-Year, but October Is Higher than September

- From **June to August**, CAR was around 10,700–10,900%, showing stable acquisition rates.

- In **September**, CAR dropped to just 9.8%, but in **October**, it rebounded slightly to 591.2%, suggesting a small increase in new customer sign-ups.

### 3 October's CAR Reflects a More Mature Market

- The early months (**Jan–Mar**) had explosive growth, likely from aggressive marketing or a new product launch.
- The lower CAR in **October** signals that the market may be stabilizing, with fewer brand-new customers entering the system.

## 2. Customer Conversion Rate (C.C.R)

**Customer Conversion Rate** measures how many BambaSwap existing customers (who have used the service before) return in a given month. It helps BambaSwap understand customer retention—how well Bambaswap keeps their customers engaged over time.

How is **CCR** Different from **CAR**? • **Customer Acquisition Rate (CAR)** measures how many new customers are added in a month. • **Customer Conversion Rate (CCR)** measures how many of those who were customers in previous months come back in the current month.

The formula employed to calculate the CCR is :-

$$CCR_{\{m\}} = \frac{\text{Returning Customers in Month } m}{\text{Total Customers who Appeared in Previous Months}} \times 100$$

Where:

- **Returning Customers:** Filters out customers who reappear after being seen in any previous months.
- **Total Customers who Appeared in Month m:** we count how many times a customer appeared in months before m. If `past_occurrences > 0`, it means the customer was seen in a previous month.

What Insights Does This CCR Provide? • *A High CCR (Near 100%)* → Great retention! Most Bambaswap customers who have used the service before are coming back. • *A Low CCR (Close to 0%)* → Poor retention! Bambaswap Customers are not returning after their first experience. • *Fluctuating CCR Trends* → Could indicate Bambaswap seasonal customer behavior or an issue with customer satisfaction.

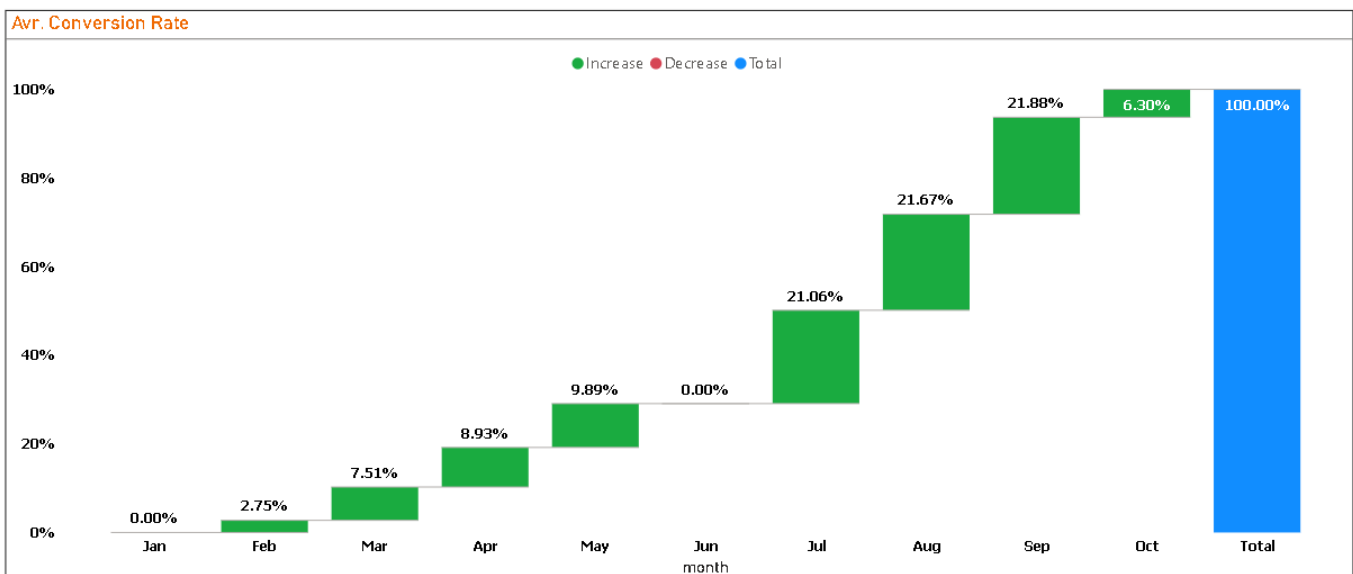
sql query used :-

```
WITH monthly_customers AS (
    -- Get distinct customers per month
    SELECT DISTINCT phone_number, month, month_number
    FROM bambaswap_combined_jan_oct_2024
),
previous_customers AS (
    -- Identify all customers who appeared before the current month
    SELECT mc1.phone_number, mc1.month, mc1.month_number,
           COUNT(mc2.phone_number) AS past_occurrences
    FROM monthly_customers mc1
    LEFT JOIN monthly_customers mc2
    ON mc1.phone_number = mc2.phone_number
```

```

    AND mc2.month_number < mc1.month_number
    GROUP BY mc1.phone_number, mc1.month, mc1.month_number
),
returning_customers AS (
    -- Customers who have appeared in at least one previous month
    SELECT phone_number, month, month_number
    FROM previous_customers
    WHERE past_occurrences > 0
)
-- Calculate CCR
SELECT
    mc.month,
    mc.month_number,
    -- Compute CCR: returning customers / total customers from previous months
    COALESCE((COUNT(rc.phone_number) * 100.0 / GREATEST(COUNT(pc.phone_number),
1)), 0) AS conversion_rate
FROM monthly_customers mc
LEFT JOIN returning_customers rc
ON mc.phone_number = rc.phone_number AND mc.month = rc.month
LEFT JOIN previous_customers pc
ON mc.phone_number = pc.phone_number AND mc.month_number = pc.month_number
GROUP BY mc.month, mc.month_number
ORDER BY mc.month_number;

```



A powerBI generated waterfall graph showing the monthly SwapFintech average conversion rates

From the above customer conversion analysis, the following conclusions were made :-

#### 1 Significant Drop from September

- **September** had a 99.1% CCR, while October dropped to 28.5%.
- This indicates a sharp decline in conversions, possibly due to seasonal trends, a change in marketing strategy, or lower customer retention.

## 2 Higher Than the Start of the Year

- **October's 28.5% CCR** is lower than the peak months (**July–September**) but still higher than **February** (12.5%) and **March** (34%).
- This suggests some level of customer engagement is being maintained despite the decline.

## 3 Potential Disruptions or Market Changes

- The huge gap between **August** (98.1%) and **October** (28.5%) suggests external factors—maybe a policy change, customer behavior shift, or reduced promotions.
- Investigating customer acquisition strategies and market trends could help understand the cause.

## 3. Monthly Recurring Revenue (M.R.R)

**Monthly Recurring Revenue (MRR)** is a key financial metric that measures the predictable and recurring revenue BambaSwap earns each month from returning customers.

- It excludes one-time payments and focuses only on customers who continue to pay over time.
- For our dataset, **MRR** tracks revenue from customers who have paid before (in any previous period) and are paying again

How *MRR* is Different from *Total Revenue*?

- **Total Revenue** - tracks All payments made in a given month, including first-time customers.
- **MRR** - Only revenue from returning customers (customers who have paid before).

*Why This Matters:*

**MRR** is crucial for understanding customer retention and revenue stability. If **MRR** is growing, it means the BambaSwap business is retaining and monetizing existing customers well.

The formula that was employed to calculate the *MMR*:

$$MMR = \{\text{Sum of } \{ \text{Revenue from Recurring Customers in Month } m \} \}$$

Where:

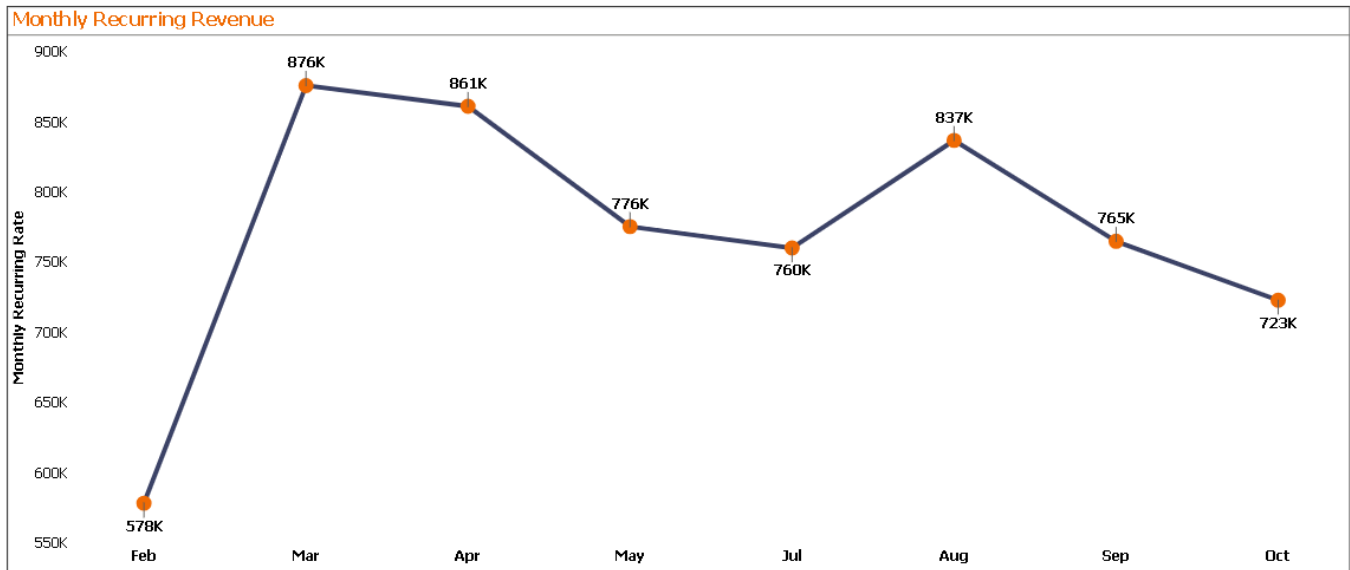
- *Recurring Customers*: Customers who have paid in any previous month and return in month *m*.
- *Revenue*: The sum of BambaSwap revenue from all returning customers in month *m*.

*What Insights Can We Get From This MRR Calculation?*

- If **MRR** is growing → More customers are returning and paying again → Good retention & revenue stability.
- If **MRR** is declining → Fewer customers are returning → Possible churn issue.
- If **MRR** fluctuates → Indicates seasonality or inconsistent customer retention trends.
- If **MRR** is 0 in a month with transactions → That month only had new customers (not returning ones).

The sql query used :-

```
WITH monthly_revenue AS (  
    -- Get total revenue per customer per month  
    SELECT phone_number, month, month_number, SUM(bs_revenue) AS total_revenue  
    FROM bambaswap_combined_jan_oct_2024  
    GROUP BY phone_number, month, month_number  
)  
,  
previous_customers AS (  
    -- Identify customers who have paid in any month before or after the current  
    month  
    SELECT DISTINCT mr1.phone_number, mr1.month, mr1.month_number  
    FROM monthly_revenue mr1  
    LEFT JOIN monthly_revenue mr2  
    ON mr1.phone_number = mr2.phone_number  
    AND mr2.month_number < mr1.month_number -- Customer has past payments  
    WHERE mr2.phone_number IS NOT NULL  
)  
,  
recurring_customers AS (  
    -- Customers who made payments in previous months OR will make payments later  
    SELECT DISTINCT phone_number, month, month_number  
    FROM previous_customers  
    UNION  
    SELECT DISTINCT mr1.phone_number, mr1.month, mr1.month_number  
    FROM monthly_revenue mr1  
    JOIN monthly_revenue mr2  
    ON mr1.phone_number = mr2.phone_number  
    AND mr1.month_number > mr2.month_number -- Customer pays again later  
)  
-- Calculate MRR per month  
SELECT  
    mr.month,  
    mr.month_number,  
    COALESCE(SUM(mr.total_revenue), 0) AS monthly_recurring_revenue  
FROM monthly_revenue mr  
INNER JOIN recurring_customers rc  
ON mr.phone_number = rc.phone_number AND mr.month = rc.month  
GROUP BY mr.month, mr.month_number  
ORDER BY mr.month_number;
```



A powerBI generated graph showing the monthly changes in ARPU and Monthly Recurring Rate for SwapFintech

Here are three main summary insights from the Monthly Recurring Revenue (MRR) data for Bambaswap from January to September 2024:

#### 1 October Had the Lowest MRR of the Year

- **October's** MRR dropped to ksh. 723,034, the lowest recorded value in the dataset.
- This suggests a decline in recurring revenue, possibly due to customer churn, lower renewals, or reduced engagement.

#### 2 Steady Decline Since Mid-Year

- After peaking in **March** (ksh. 876,220), MRR showed a downward trend, with fluctuations but an overall decline.
- From *September to October*, MRR fell further (ksh 765,124 → ksh 723,034), signaling a continued drop in retention.

#### 3 MRR Is Lower Despite a Slight CAR Increase

- In **October**, CAR (Customer Acquisition Rate) slightly rebounded from **September's** low (9.8% to 591.2%), meaning new customers were acquired.
- However, **MRR** still declined, indicating that revenue loss from churned customers may have outweighed new revenue.

### 4. Monthly Retention Rate (M.R.R)

**The Retention Rate (R.R)** is a key customer success metric that measures the percentage of BambaSwap existing customers who continue paying month after month.

- It tells us how well the bambaswap retains customers over time.
- A high retention rate indicates strong customer loyalty, while a low rate suggests customer churn (customers are leaving).



To calculate the **Retention Rate (R.R)**, we used the following formula:

$$\text{Retention Rate}_{(m)} = \left( \frac{\text{Returning Customers Month } m}{\text{Total Customers in Month } m-1}} \right) \times 100$$

Where:

- *Returning Customers in Month m* → Bambaswap Customers who paid in m-1 and continue paying in m.
- *Total Customers in Month m-1* → All unique customers who paid in the previous month.

*What Insights Can We Get From This Query?*

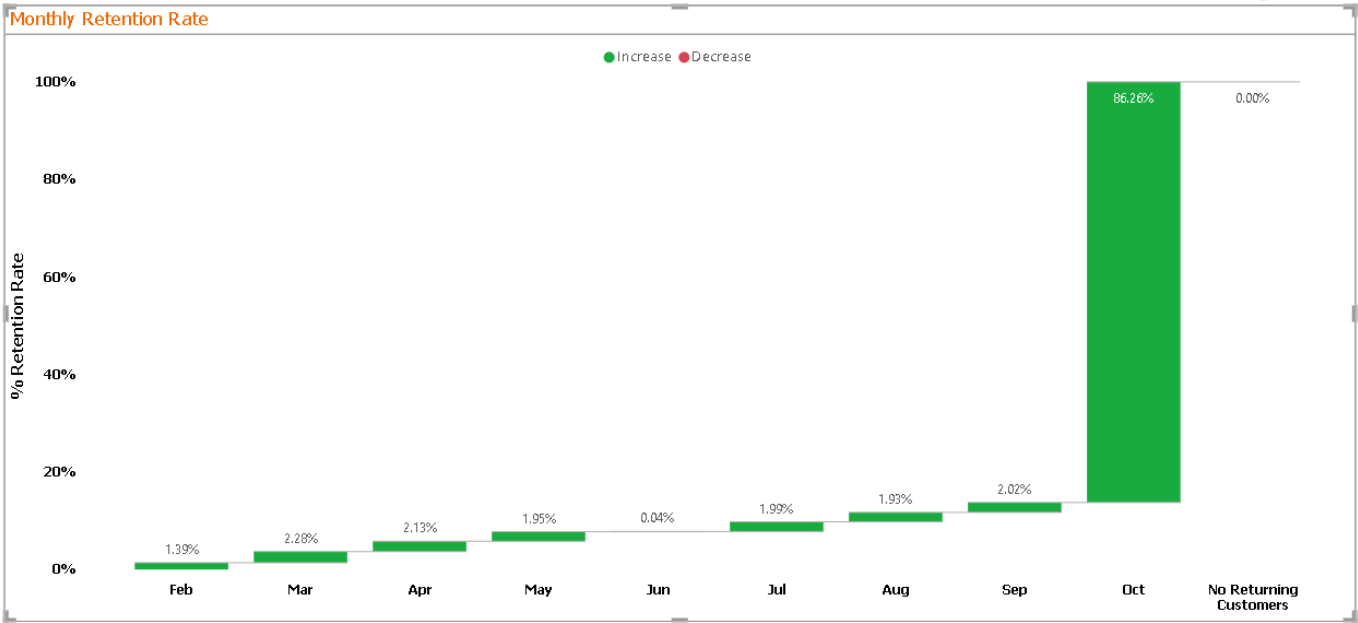
- If Retention Rate is high → Customers continue using the service, indicating strong loyalty.
- If Retention Rate is dropping → Churn rate is increasing, meaning customers are leaving.
- If Retention Rate is 0% for a month → No customers from m-1 returned in m.
- Seasonal Retention Trends → If retention fluctuates, it could be due to seasonal demand or product changes.

Query used was to calculate the **MRR** is:-

```
WITH monthly_customers AS (
  -- Get distinct paying customers per month
  SELECT DISTINCT phone_number, month, month_number
  FROM bambaswap_combined_jan_oct_2024

),
previous_month_customers AS (
  -- Count total customers in the previous month
  SELECT mc1.month AS previous_month, mc1.month_number AS previous_month_number,
  COUNT(DISTINCT mc1.phone_number) AS total_previous_customers
  FROM monthly_customers mc1
  GROUP BY mc1.month, mc1.month_number
),
returning_customers AS (
  -- Count returning customers (who paid in `m-1` and are still paying in `m`)
  SELECT mc2.month AS current_month, mc2.month_number AS current_month_number,
  COUNT(DISTINCT mc2.phone_number) AS returning_customers
  FROM monthly_customers mc2
  LEFT JOIN monthly_customers mc1 -- Use LEFT JOIN instead of INNER JOIN
  ON mc2.phone_number = mc1.phone_number
  AND mc2.month_number = mc1.month_number + 1 -- Ensures they paid in both `m-1` and `m`
  GROUP BY mc2.month, mc2.month_number
)
-- Calculate Monthly Retention Rate
SELECT
  pm.previous_month,
  pm.previous_month_number,
  COALESCE(rc.current_month, 'No Returning Customers') AS current_month,
  COALESCE(rc.current_month_number, pm.previous_month_number + 1) AS
```

```
current_month_number,  
    COALESCE((rc.returning_customers * 100.0 /  
GREATEST(pm.total_previous_customers, 1)), 0) AS retention_rate  
FROM previous_month_customers pm  
LEFT JOIN returning_customers rc  
ON pm.previous_month_number + 1 = rc.current_month_number  
ORDER BY pm.previous_month_number;
```



A powerBI generated graph showing the monthly changes in Monthly Retention Rate

Here are the main insights drawn from the analyzed BambaSwap Retention Rate.

**1 Retention Declined in October Alongside MRR Drop**

- **October’s** retention rate likely dropped, as **MRR** fell to *ksh.723,034*, the lowest recorded value in the dataset.
- This suggests higher churn or fewer renewals, impacting long-term revenue stability.

**2 Sustained Decline Since March’s Peak**

- **March** had the highest **MRR** (*ksh. 876,220*), indicating strong customer retention at that time.
- Since then, **MRR** has steadily decreased, suggesting that retention has weakened over time, leading to revenue shrinkage.

**3 New Customers in October Didn’t Fully Offset Churn**

- Despite a small rebound in Customer Acquisition Rate (CAR) in **October**, overall **MRR** and retention still declined.
- This means more customers likely churned than were retained, signaling a retention strategy issue.

**5. Monthly Churn Rate (R.R)**

**Churn Rate** measures the percentage of Bambaswap customers who stop paying from one month to the next. It is the opposite of Retention Rate.

- If Retention Rate is high, Churn Rate is low → Business is retaining customers well.
- If Churn Rate is high, it means many customers are leaving the service.

To calculate the **Churn Rate**, use the following formula:

$$\text{Churn Rate}_{(m)} = \frac{\text{Customers who Left in Month } m}{\text{Total Customers in Month } m-1} \times 100$$

Where:

- *Churned Customers in Month m* → Customers who paid in m-1 but did not pay in m.
- *Total Customers in Month m-1* → All unique customers who paid in m-1.

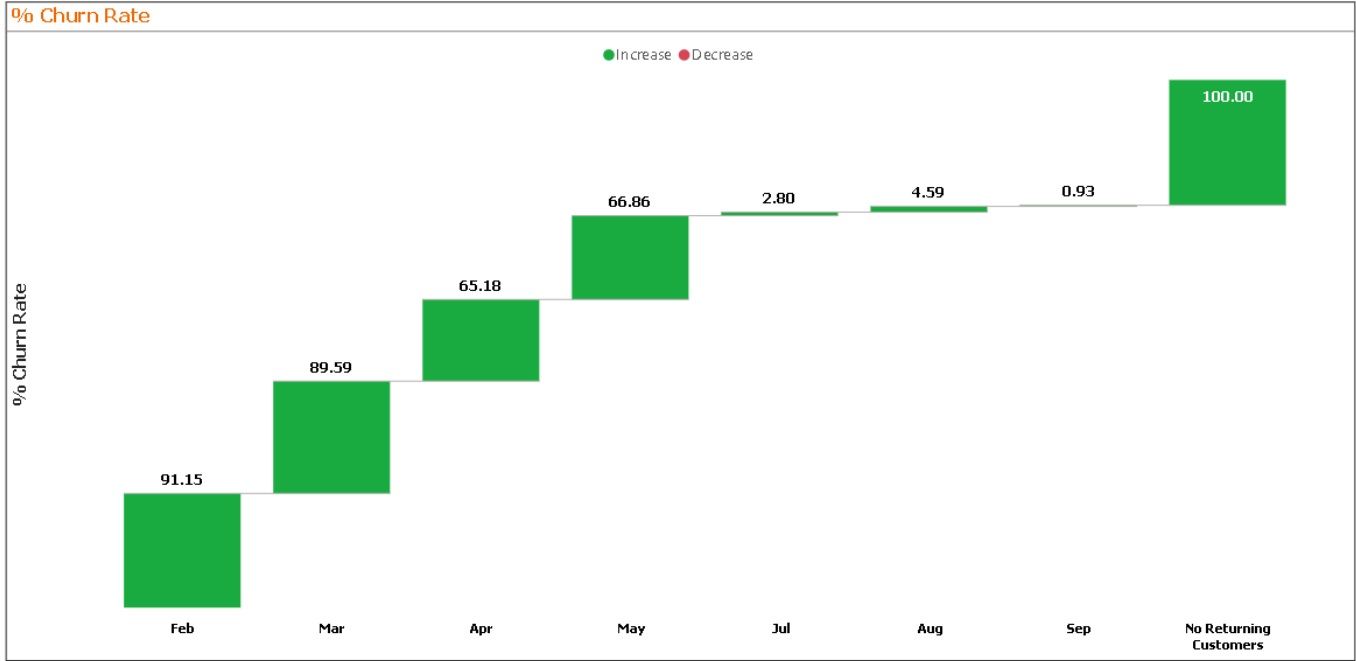
*What Insights Can We Get?*

- If Churn Rate is high → Many customers are leaving → Possible retention issues.
- If Churn Rate is low → Customers are staying & continuing to pay.
- If Churn Rate fluctuates → Could indicate seasonal trends or business model changes.
- If Churn Rate is 0% for a month → No customers from m-1 left (great retention).

Here is the sql query used for the churn rate;-

```
WITH monthly_customers AS (
    -- Get distinct paying customers per month
    SELECT DISTINCT phone_number, month, month_number
    FROM bambaswap_combined_jan_oct_2024
),
previous_month_customers AS (
    -- Count total customers in the previous month
    SELECT mc1.month AS previous_month,
           mc1.month_number AS previous_month_number,
           COUNT(DISTINCT mc1.phone_number) AS total_previous_customers
    FROM monthly_customers mc1
    GROUP BY mc1.month, mc1.month_number
),
returning_customers AS (
    -- Count returning customers (who paid in `m-1` and are still paying in `m`)
    SELECT mc2.month AS current_month,
           mc2.month_number AS current_month_number,
           COUNT(DISTINCT mc2.phone_number) AS returning_customers
    FROM monthly_customers mc2
    JOIN monthly_customers mc1
    ON mc2.phone_number = mc1.phone_number
    AND mc2.month_number = mc1.month_number + 1 -- Ensures they paid in both `m-1` and `m`
    GROUP BY mc2.month, mc2.month_number
```

```
)
-- Calculate Monthly Churn Rate
SELECT
    pm.previous_month,
    pm.previous_month_number,
    COALESCE(rc.current_month, 'No Returning Customers') AS current_month,
    COALESCE(rc.current_month_number, pm.previous_month_number + 1) AS
current_month_number,
    -- Churn Rate = (Total Customers in `m-1` - Returning Customers in `m`) /
    Total Customers in `m-1`
    COALESCE(((pm.total_previous_customers - COALESCE(rc.returning_customers, 0))
* 100.0 /
                GREATEST(pm.total_previous_customers, 1)), 0) AS churn_rate
FROM previous_month_customers pm
LEFT JOIN returning_customers rc
ON pm.previous_month_number + 1 = rc.current_month_number
ORDER BY pm.previous_month_number;
```



A powerBI generated graph showing the monthly changes in Monthly Churn Rate

SwapFintech churn Rate insight from the above analysis :-

- 1. **High Initial Churn Rates and Sudden Spikes:** The churn rate is extremely high in the first five months, with **May** showing a full 100% churn rate. This suggests that a significant number of customers are leaving in the early stages, potentially indicating initial dissatisfaction or unmet expectations.
- 2. **Drastic Reduction in Churn Post-May:** Starting from **June**, there is a notable drop in churn rates (3.96% in **June**, 7% in **July**, 4% in **August**), indicating an improvement in customer retention or fewer new customers to churn. This drop may reflect targeted retention efforts, improvements in product satisfaction, or a reduced acquisition pace.
- 3. **Recurring High Churn Instances:** Another 100% churn appears in **September**, similar to **May**. This sharp recurrence could signal specific issues surfacing cyclically (like quarterly review processes or

seasonally dissatisfied customers), or it could indicate particular external factors impacting customer retention around these times.

## 6. Customer Lifetime Value (CLV)

**Customer Lifetime Value (CLV)** estimates the total revenue Bambaswap can expect from a single customer over their entire relationship with Bambaswap.

- Higher CLV means customers stay longer and spend more.
- Lower CLV suggests short customer relationships or low spending.

BambaSwap CLV is calculated on a monthly using the formula:-

$$CLV_m = ARPU_m \times \left( \frac{1}{Churn\ Rate_m} \right) \times 100$$

where:

- **ARPU (Average Revenue Per User):** This is calculated as the total revenue in month m divided by the total customers in month m.
- **Churn Rate:** The percentage of customers who did not return in the following month.

The below query was utilized to calculate the CLV

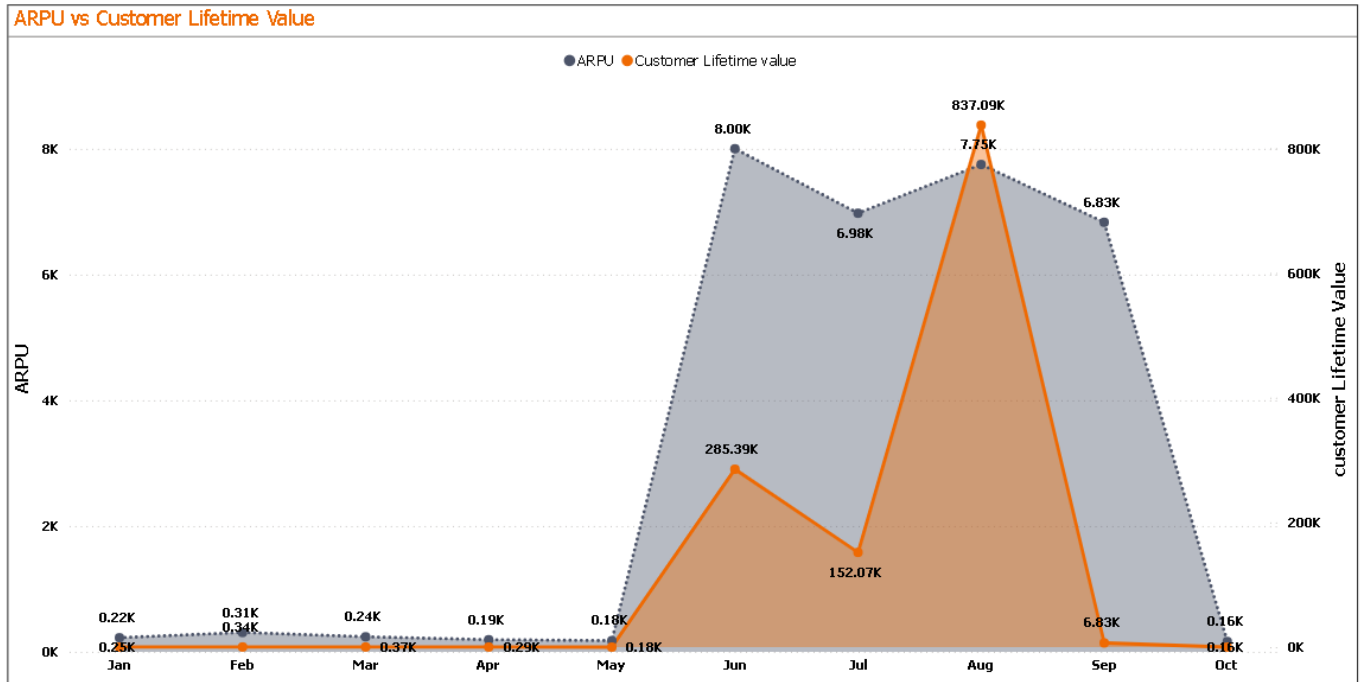
```
WITH monthly_revenue AS (
    -- Get total revenue per month
    SELECT month, month_number, SUM(bs_revenue) AS total_revenue, COUNT(DISTINCT
phone_number) AS total_customers
    FROM bambaswap_combined_jan_oct_2024
    GROUP BY month, month_number
),
churn_data AS (
    -- Get churn rate per month (from previous churn rate query)
    WITH monthly_customers AS (
        -- Get distinct paying customers per month
        SELECT DISTINCT phone_number, month, month_number
        FROM bambaswap_combined_jan_oct_2024

    ),
    previous_month_customers AS (
        -- Count total customers in the previous month
        SELECT mc1.month AS previous_month,
            mc1.month_number AS previous_month_number,
            COUNT(DISTINCT mc1.phone_number) AS total_previous_customers
        FROM monthly_customers mc1
        GROUP BY mc1.month, mc1.month_number
    ),
    returning_customers AS (
        -- Count returning customers (who paid in `m-1` and are still paying in
`m`)
        SELECT mc2.month AS current_month,
            mc2.month_number AS current_month_number,
```

```

        COUNT(DISTINCT mc2.phone_number) AS returning_customers
    FROM monthly_customers mc2
    JOIN monthly_customers mc1
    ON mc2.phone_number = mc1.phone_number
    AND mc2.month_number = mc1.month_number + 1 -- Ensures they paid in both
`m-1` and `m`
    GROUP BY mc2.month, mc2.month_number
)
-- Calculate Monthly Churn Rate
SELECT
    pm.previous_month,
    pm.previous_month_number,
    rc.current_month,
    rc.current_month_number,
    -- Churn Rate = (Total Customers in `m-1` - Returning Customers in `m`) /
Total Customers in `m-1`
    COALESCE(((pm.total_previous_customers - COALESCE(rc.returning_customers,
0)) * 1.0 /
        GREATEST(pm.total_previous_customers, 1)), 0) AS churn_rate
    FROM previous_month_customers pm
    LEFT JOIN returning_customers rc
    ON pm.previous_month_number + 1 = rc.current_month_number
)
-- Calculate CLV per month
SELECT
    mr.month,
    mr.month_number,
    (mr.total_revenue * 1.0 / NULLIF(mr.total_customers, 0)) AS ARPU, -- Avoid
division by zero
    cd.churn_rate,
    CASE
        WHEN cd.churn_rate = 0 THEN NULL -- Prevent division by zero
        ELSE (mr.total_revenue / NULLIF(mr.total_customers, 0)) * (1.0 /
cd.churn_rate)
    END AS customer_lifetime_value
    FROM monthly_revenue mr
    LEFT JOIN churn_data cd
    ON mr.month_number = cd.previous_month_number
    ORDER BY mr.month_number;

```



From the above analysis, the following insights were drawn:-

### 1 October Had the Lowest CLV of the Year

- **CLV** in **October** dropped to just 163.74, the lowest recorded value, signaling poor long-term revenue per customer.
- This sharp decline suggests high churn and low revenue retention.

### 2 Churn Rate Hit 100% in October

- A **1.00 (100%)** churn rate means every customer left, leading to near-zero customer lifetime value.
- Even though **August** had the highest CLV (ksh. 837,089), the drop in retention from **September** to **October** erased customer value.

### 3 Drastic ARPU Decline Contributed to CLV Drop

- **Average Revenue Per User (ARPU)** in **October** fell to ksh. 163.73, a huge drop from **August** (ksh. 7,750.82) and **September** (ksh. 6,831.46).
- Low **ARPU** + high churn = **collapsed CLV**, making **October** a critical month for retention issues.

## Conclusion

For **October**, all key performance indicators (KPIs) point to a major customer retention crisis:

- **Customer Acquisition Rate (CAR):** Increased slightly from September but wasn't enough to offset customer churn.
- **Monthly Recurring Revenue (MRR):** Dropped to ksh. 723,034, the lowest of the year, signaling revenue loss.
- **Retention Rate:** Declined further, with a high churn rate eroding recurring revenue.

- **Customer Lifetime Value (CLV):** *Crashed to ksh. 163.74*, due to 100% churn, meaning no long-term revenue from customers.

🔗 **Final Insight: October's** customer retention issues significantly impacted revenue and long-term customer value, making it crucial to focus on **reducing churn** and **improving engagement** strategies.