

| **MINI PROJECT ON SQL**  **Checkout Flow Optimization Analysis** |
| --- |

| **Developed by**  **Members:**   | **No.** | **Student Name** | **Student ID** | | --- | --- | --- | | 1 | Phạm Thị Quỳnh Lâm | Student1521935 | | 2 | Nguyễn Hữu Thắng | Student1503296 | | 3 | Nguyễn Thanh Bình | Student1521236 |  * **Class No.: DT2307L-NK** * **Start Date:** **13/11/2023** * **End Date: 22/11/2023** * **Name of the Coordinator: Ho Nhut Minh** * **Date of Submission: 02/01/2024** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

| **1. Case Description** |
| --- |

**Overview:** In this project, we invite you to embark on a practical, real-world case centered around optimizing an online platform’s checkout flow. The goal of the analysis is to enhance the purchase experience for users on the webpage—a critical aspect of online retail that directly impacts conversion rates and customer satisfaction. You'll delve into an actual database, discover crucial insights, and acquire practical experience enhancing the website's purchase checkout system.

**Objective:** The primary goal is to analyze the platform’s checkout process by devising and building from the ground up a three-page story-based dashboard, displaying key metrics, insights, and visualizationsfor 07-01-2022 to 01-31-2023. Based on your analysis, you must develop a strategy to improve the checkout process and enhance the user payment experience.

Note below how we categorize users based on their interaction with subscriptions and payments.

* **Successful Checkout**: A user completes a payment for a subscription without any issues.
* **Failed Checkout**: A user encounters problems while paying and cannot complete the transaction.
* **Abandoned Cart**: A user adds a subscription to their cart but leaves without finalizing the payment.

Consider the vital metrics our dashboard must address for a comprehensive checkout investigation:

* **Monthly Checkout Success Rate** measures the percentage of successful checkouts compared to monthly attempts. A high rate shows our checkout process is efficient, while a low rate suggests potential areas of improvement.
* **Monthly Cart Abandonment Rate** shows the percentage of users who added items to their cart but didn't buy. If this rate is high, issues like complicated checkout processes or pricing concerns might stop customers from buying.
* **Most Common Checkout Errors and Device Correlations** must be identified during checkout to see if specific devices are more prone to these issues. This helps target tech improvements and suggests ways to boost our checkout success rate.

| **2. Retrieving Checkout Steps Information** |
| --- |

Use the tables checkout\_actionsand checkout\_carts from the **checkout\_database** to retrieve a result set covering the entire period and containing the following fields:

* action\_date: the day on which the checkout activity took place
* count\_total\_carts: the count of shopping carts created each day during the specified timeframe
* count\_total\_checkout\_attempts: the count of purchase attempts each day
* count\_successful\_checkout\_attempts: the count of successful purchases each day

Creating such a result set is crucial because it provides a consolidated view of daily checkout activities over the specified period. By analyzing this data, we can better understand user behavior patterns, identify potential issues in the checkout process, and pinpoint areas for optimization to enhance the overall user experience.

To build the result set, we advise you to utilize common table expressions (CTEs) for capturing the various stages of a subscription attempt. You’ll use the WITH clause to create such common table expressions that can later be easily referenced with the help of a SELECT statement.

Save the result set as a CSV file and call it checkout\_steps.csv.

In the first three steps of the instructions, we’ll create the following CTEs:

* One that keeps all carts created
* One that stores all checkout attempts
* And one that keeps only the successful attempts.

Follow the outlined steps and substitute the question marks with the correct values and expressions when necessary.

1. Let’s start by building the CTE that keeps all created carts. First, initialize the WITH clause and then, select all fields from the checkout\_carts table.

Note the following skeleton of this query:

WITH total\_carts\_created AS (

SELECT \*

FROM checkout\_carts),

2. Now, using the table you've just set up, pull the data on all users who have created a purchase cart and have also attempted to finalize a purchase. Examine the appropriate action\_name in the relevant table to sift through and collect the required data into a new temporary result.

Note the following skeleton of this query:

total\_checkout\_attempts AS (

SELECT cc.user\_id,

ca.action\_name,

ca.action\_date

FROM checkout\_carts cc

LEFT JOIN checkout\_actions ca

ON cc.user\_id = ca.user\_id

WHERE ca.action\_name LIKE '%checkout%'

AND ca.action\_date BETWEEN '2022-07-01' AND '2023-01-31'),

3. Next, construct a temporary result set that captures only the successful checkout attempts. Pull the necessary data from the previous CTE, focusing on events that resulted in success.

Note the following skeleton of this query:

successful\_checkout\_attempts AS (

SELECT \*

FROM total\_checkout\_attempts

WHERE action\_name LIKE '%success%'),

count\_total\_carts\_created AS

(SELECT action\_date,

COUNT(\*) AS count\_total\_carts\_created

FROM total\_carts\_created

GROUP BY action\_date),

4. With the relevant CTEs in place, it's time to implement a method to count the specific records we're targeting. We'll use additional CTEs to capture the necessary count. Begin by counting the total number of carts from the CTE containing all purchased carts daily.

Note the following skeleton of this query:

count\_total\_carts\_created AS

(SELECT action\_date,

COUNT(\*) AS count\_total\_carts\_created

FROM total\_carts\_created

GROUP BY action\_date),

5. Construct a query that counts the total daily checkout attempts following the same approach.

Note the following skeleton of this query:

count\_total\_checkout\_attempts AS (

SELECT action\_date,

COUNT(\*) AS count\_total\_checkout\_attempts

FROM total\_checkout\_attempts

GROUP BY action\_date),

6. Now, construct a query to count only the successful daily attempts.

Note the following skeleton of this query:

count\_successful\_checkout\_attempts AS (

SELECT action\_date,

COUNT(\*) AS count\_successful\_checkout\_attempts

FROM successful\_checkout\_attempts

GROUP BY action\_date)

7. Finally, select the pertinent data from the last three CTEs and pull out the desired columns for your checkout\_steps result set. Use the IFNULL function to substitute the null values with a value. Order the record by action\_date to obtain the information chronologically.

Note the following skeleton of this query:

SELECT t1.action\_date,

count\_total\_carts\_created,

IFNULL(count\_total\_checkout\_attempts, 0) AS count\_total\_checkout\_attempts,

IFNULL(count\_successful\_checkout\_attempts, 0) AS count\_successful\_checkout\_attempts

FROM count\_total\_carts\_created t1

LEFT JOIN count\_total\_checkout\_attempts t2 ON t1.action\_date = t2.action\_date

LEFT JOIN count\_successful\_checkout\_attempts t3 ON t1.action\_date = t3.action\_date

WHERE t1.action\_date BETWEEN '2022-07-01' AND '2023-01-31'

ORDER BY t1.action\_date;

8. After executing the final query, save the result set as a CSV file called checkout\_steps.csv.

9. To retrieve payment information by month, we can replace the query in section 7 with the following query:

SELECT MONTH(t1.action\_date) AS month,

SUM(count\_total\_carts\_created) AS count\_total\_carts\_created,

SUM(IFNULL(count\_total\_checkout\_attempts, 0)) AS count\_total\_checkout\_attempts,

SUM(IFNULL(count\_successful\_checkout\_attempts, 0)) AS count\_successful\_checkout\_attempts,

SUM(IFNULL(count\_successful\_checkout\_attempts, 0))/SUM(IFNULL(count\_total\_checkout\_attempts, 0)) AS checkout\_success\_rate,

SUM(count\_total\_carts\_created) - SUM(IFNULL(count\_total\_checkout\_attempts, 0)) AS abandonment\_carts,

(SUM(count\_total\_carts\_created) - SUM(IFNULL(count\_total\_checkout\_attempts, 0)))/SUM(count\_total\_carts\_created) AS abandonment\_carts\_rate

FROM count\_total\_carts\_created t1

LEFT JOIN count\_total\_checkout\_attempts t2 ON t1.action\_date = t2.action\_date

LEFT JOIN count\_successful\_checkout\_attempts t3 ON t1.action\_date = t3.action\_date

WHERE t1.action\_date BETWEEN '2022-07-01' AND '2023-01-31'

GROUP BY MONTH(t1.action\_date)

ORDER BY t1.action\_date;

| 3. Retrieving Checkout Errors Information |
| --- |

Use the tables checkout\_actionsand checkout\_carts from the **checkout\_database** to retrieve a new result set called checkout\_errors containing the following fields:

* user\_id – the identification number of the student attempting to checkout
* action\_date – the date of the checkout attempt
* action\_name – the text detailing the specific action performed by the user
* error\_message – the text with the received error (if any)
* device – the type of the used device (desktop or mobile)

Generating this result set is essential because it offers a comprehensive overview of checkout errors encountered by users during the defined time frame. Through its analysis, we can identify common issues students face during checkout—enabling us to make informed decisions on potential enhancements or changes to the platform.

This insight is valuable in optimizing the checkout experience and minimizing obstacles, which can directly contribute to increased user satisfaction and revenue.

Follow the steps outlined below.

Ensure to substitute the question marks with the correct values and expressions.

Select the pertinent columns from the checkout\_actions table that offer details about the error received post-action and the device utilized during the process. You can filter exclusively for events that resulted in errors or retrieve all actions and remove the irrelevant ones later. Our primary interest lies in the error messages and the devices used.  
Note the following skeleton of this query:  
WITH check\_out\_errors\_info AS (

SELECT

user\_id, action\_date, action\_name, error\_message, device

FROM

checkout\_actions

WHERE action\_date BETWEEN '2022-07-01' AND '2023-01-31' AND action\_name like '%checkout%'

GROUP BY user\_id

ORDER BY action\_date)

SELECT error\_message,

SUM(CASE WHEN device = 'desktop' THEN 1 ELSE 0 END) AS total\_errors\_on\_desktop,

SUM(CASE WHEN device = 'mobile' THEN 1 ELSE 0 END) AS total\_errors\_on\_mobile,

COUNT(\*) AS total\_errors,

SUM(CASE WHEN device = 'desktop' THEN 1 ELSE 0 END)/COUNT(\*) AS errors\_on\_desktop\_rate,

SUM(CASE WHEN device = 'mobile' THEN 1 ELSE 0 END)/COUNT(\*) AS errors\_on\_mobile\_rate

FROM check\_out\_errors\_info

GROUP BY error\_message

ORDER BY total\_errors DESC;

After executing the query, save the result set as a CSV file called checkout\_errors.csv.

**In addition, from the given database, we can query to retrieve some information for the analysis process as follows:**

* **Type of purchase:**

WITH type\_of\_purchase\_table AS(

SELECT (CASE WHEN action\_name LIKE '%lifetime%' THEN 'lifetime package'

WHEN action\_name LIKE '%annual%' THEN 'annual package'

WHEN action\_name LIKE '%quarterly%' THEN 'quarterly package'

WHEN action\_name LIKE '%monthly%' THEN 'monthly package'

ELSE 'single course' END) AS type\_of\_purchase

FROM checkout\_actions

WHERE action\_name LIKE '%checkout%')

SELECT type\_of\_purchase,

COUNT(\*) AS total

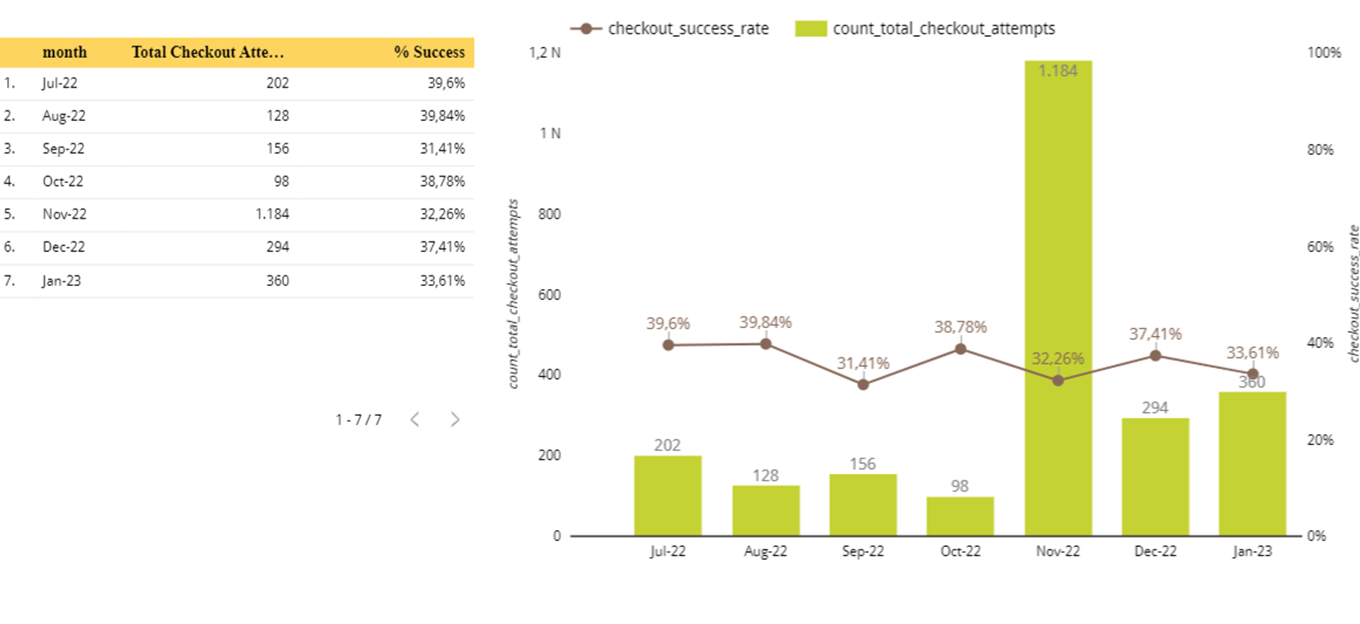
FROM type\_of\_purchase\_table

GROUP BY type\_of\_purchase;

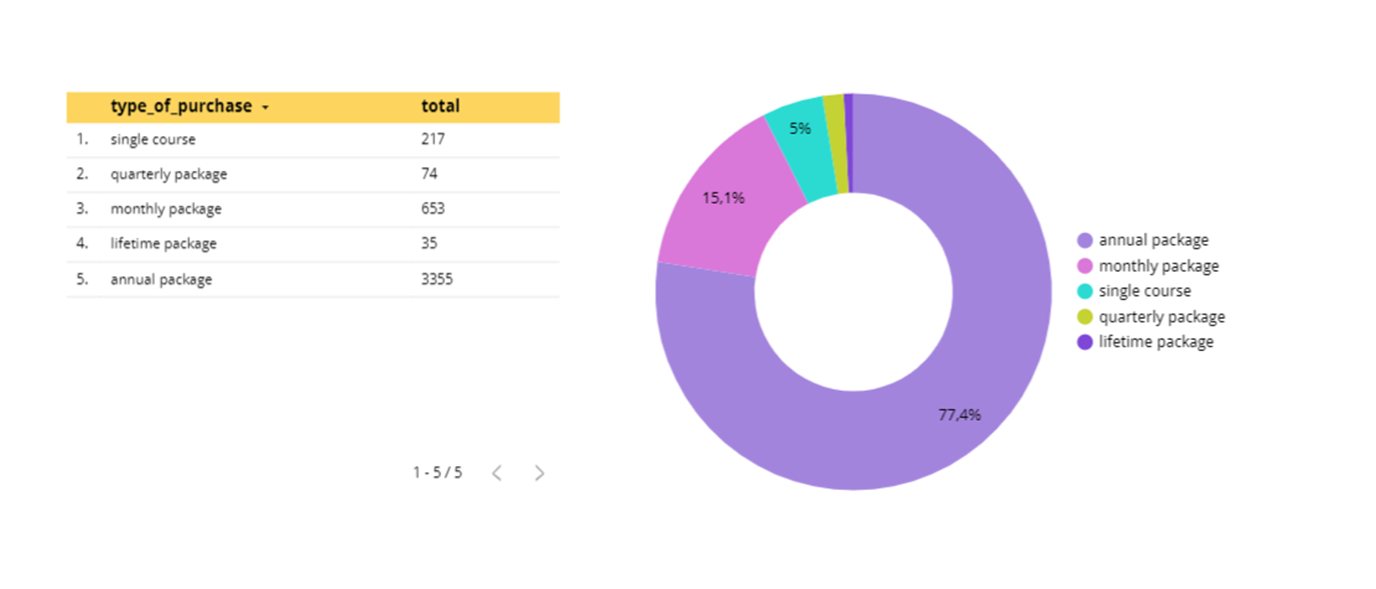
| 4. Creating the Charts: Monthly Checkout Success Rate |
| --- |

To construct the desired story, begin by creating individual sheets. Next, merge these sheets into dashboards and position the dashboards on distinct story points. This approach ensures you have a comprehensive, interactive visualization tool to aid your data analysis.

Begin by utilizing the **checkout\_steps** data source to craft a dual graph illustrating the monthly checkout success rate. This graph should display the total number of checkout attempts and the percentage of successful ones. Proceed with the steps provided below to accomplish this.

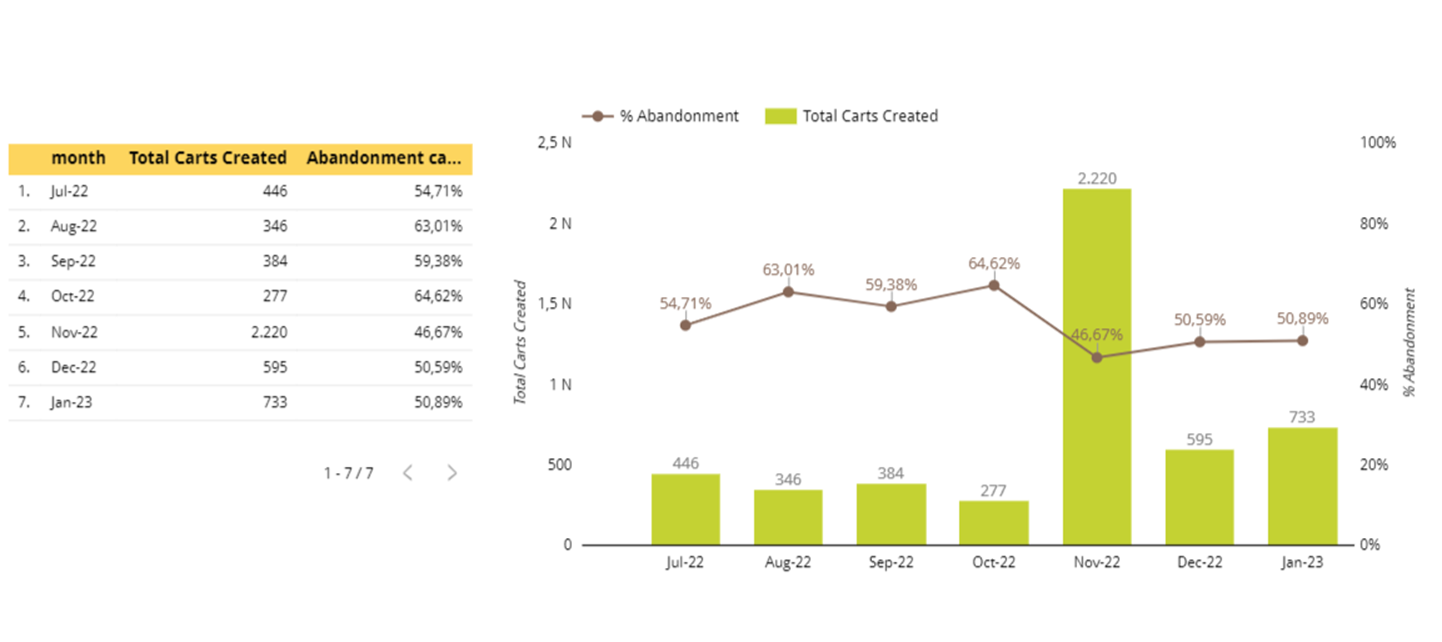


\* Types of purchase:



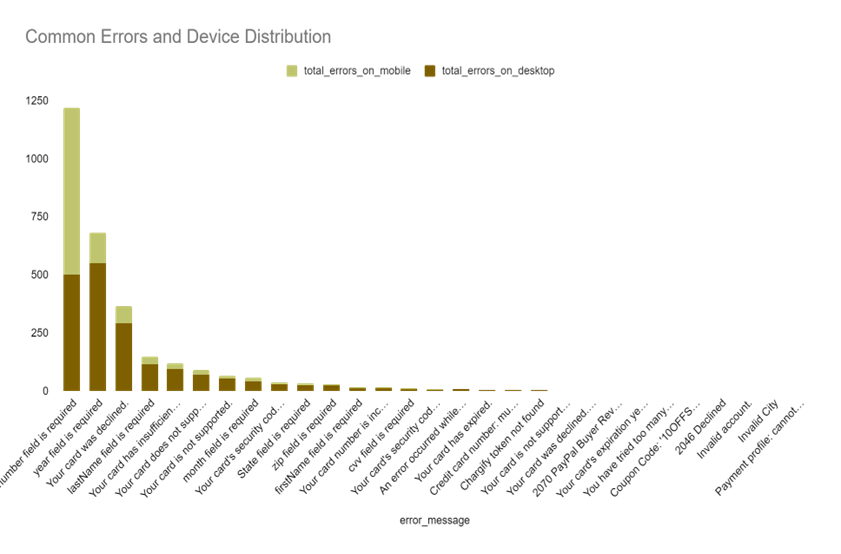
| Creating the Charts: Monthly Cart Abandonment Rate |
| --- |

Now that you're ready with the first visualization, use the **checkout\_steps** table to create another dual graph about the monthly number of purchase carts (a bar chart) and the relevant abandonment rate (a line chart).



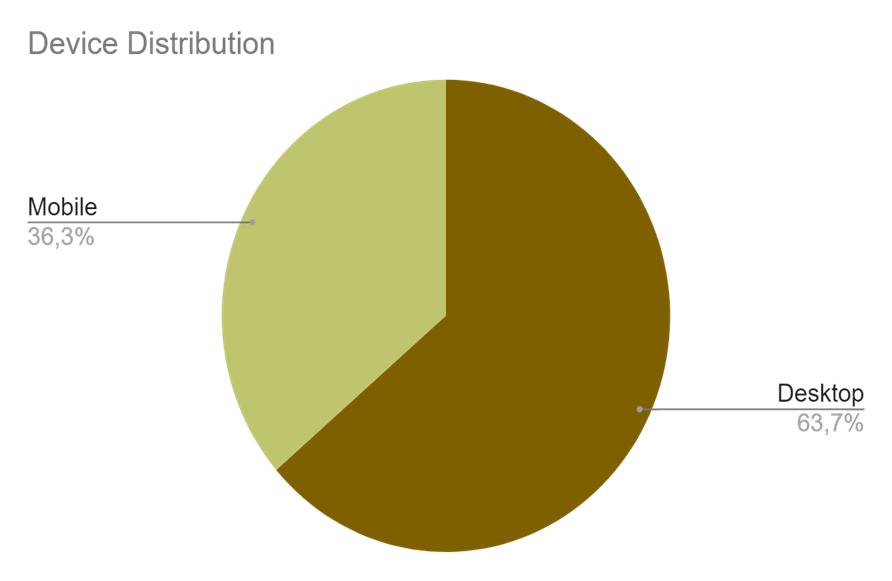
| Creating the Charts: Error Messages |
| --- |

The third visualization for the story shows the most frequent error messages encountered during failed checkout attempts. Create a horizontal bar chart showing the most common errors.



| Creating the Charts: Device Distribution |
| --- |

* For your upcoming task, create a horizontal stacked bar chart that categorizes device usage into percentages for mobile and desktop. This will let you quickly determine which device is more commonly used.



## 

| 5. Creating Story |
| --- |

Well done on completing all the necessary graphs! Now, your subsequent task is to distribute these graphs over three story points.

* **Story Point 1** includes the dual graph of monthly checkout attempts and the relevant success rates & donut chart of types of purchase.
* Observation 1: November's total number of checkout attempts is the highest of the year. Meanwhile, October is the month that has total number of check out attempt

**Reason:** November is when the largest sales of the year, including Black Friday, take place.

* Observation 2: The average checkout success rate is quite low (about 36%). November has nearly the lowest checkout success rate even though it had the most total checkout attempts. Months with a high total number of checkout attempts have lower checkout success rate.

**Reason:**

* Complicated process: There are too many difficult steps in the payment process, or there are unclear instructions.
* Freetime: Because there is a lot of free time during July and August, the success rate is high
* Limited method: Apply wrong payment method that is not in list
* Overload system: The payment system is overloaded in the peak months
* Observation 3: The “Annual package” product type has the highest paid volume and “Monthly package” is more popular than the other product.
* **Story Point 2** comprises the dual graph about the monthly number of checkout carts and the relevant abandonment rates.

- Reason: At the end of the year, the need for additional knowledge increases and falls during the big discount seasons of the year.

- Comment: November's total number of carts created is the highest of the year, along with that, November's abandoned cart rate is the lowest.On the contrary, October is the month with the lowest total number of carts created and the highest cart abandonment rate.

- Causes of the monthly number of checkout carts and the relevant abandonment rates:

* Waiting & Sales: Customers wait for big sales that cause rate differences
* Extra cost: Maybe the price ís too high, making customers confused or wondering
* Payment process: Customers no longer want to pay because the payment procedure is too difficult
* Login to pay: The system requires customers must to login to receive payment

→ **Because of that, it leads to yearly repetition in selling products**

-Solution:

* Quick Access: Use "Quick Sign-In" or "One-Click Sign-In“ feature or link to social network acc
* Advertising: Increase advertising of products with high purchase numbers
* Trust: Create trust with product reviews to increase success rate
* Notification: Send clients an email letting them know what's still in their basket so they won't forget
* Discounts: Offer monthly discounts
* Provide more suggestions: Suggest relevant courses on the website and app.
* **Story Point 3** encompasses two horizontal stacked bar charts: one with the most common error messages and one with the proportions of devices used for checkout.

Comments: The desktop usage for payments is higher compared to mobile payments, and there are common error messages, such as: number field, year field, card denied, last name field.

Reason:

· UI/UX: Operations on the desktop easier than on the mobile

· Too much unnecessary field: Customer skip fields that they feel unnecessary

· Payment method: Few payment method/unclearly payment constraints

Solution:

UI/UX

· Test and improve user experience

· Add unique features for mobile

· Enhance mobile applications

· User experience survey

Required

· Add the autofill feature with the existing information

· Use Google Auto-fill complete address

The unnecessary fields

· Remove unnecessary fields

· Using visual cues to guide customer through checkout

Payment

· Diversity of payment methods

· Payment security

| 6. Interpreting the Results |
| --- |

Begin by examining the story pages and identifying trends.

* Are there months that underperform or outperform others significantly?

-There are months like August, September, and October that are less active than the remaining months.

-November is the most active month of all months.

* Can you discern a correlation between the error messages received and the devices on which they occurred?

To summarize the correlation between common errors on desktop and mobile:

1."Number Field" Error:

Compare the error rates on desktop and mobile to assess the positive or negative correlation with the "Number Field."

2."Year Field" Error:

Examine the error rates on both devices to evaluate the positive or negative correlation with the "Year Field."

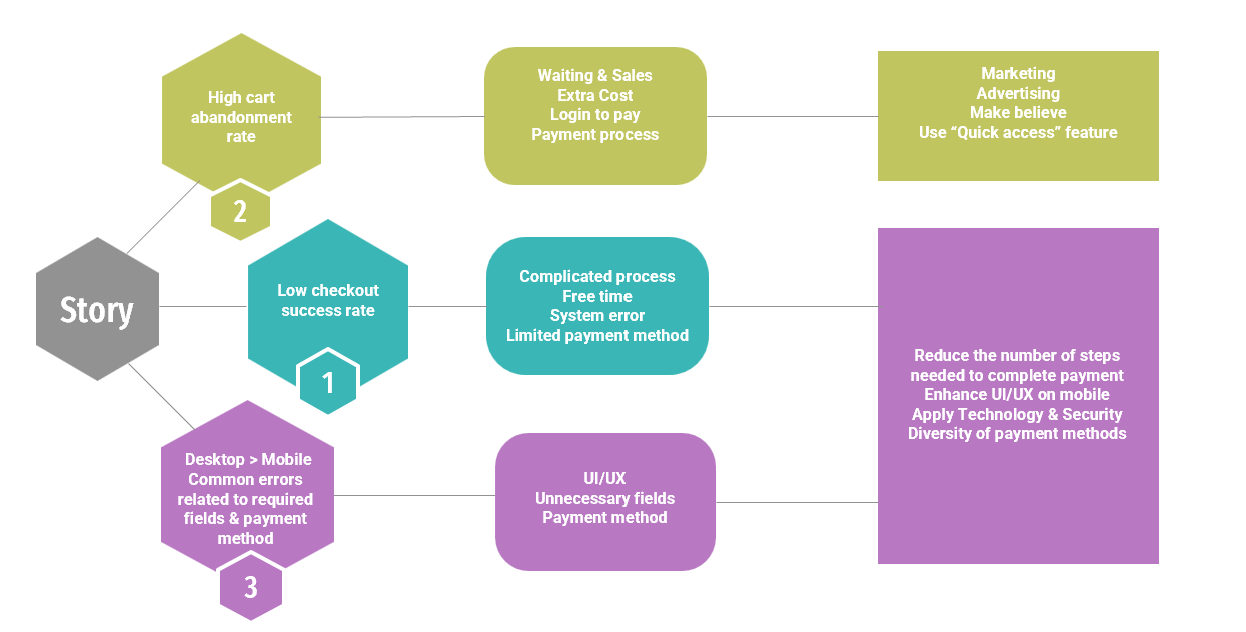
3."Card Denied" Error:

Check the discrepancy in error rates for "Card Denied" between desktop and mobile to assess the correlation.

4."Last Name Field" Error:

Contrast the error rates on desktop and mobile to determine the correlation between the "Last Name Field" and the device used.

* What suggestions can you make to enhance the overall checkout process?



## Quiz

**Question 1:**

In the context of our project on checkout flow optimization, understanding monthly performance is crucial to pinpointing areas of improvement and potential anomalies. Leveraging the story you've constructed, which month—over the entire analysis period—stands out as the most successful in terms of checkout attempts?

- In the context of our project on checkout flow optimization, November stands out as the most prominent month in terms of checkout attempts, with a total of 1184 checkout attempts recorded.

**Question 2:**

Working on such an analysis, it’s essential to identify the peaks and trenches in our monthly performance. By delving into the story you've assembled, can you determine which month—throughout the entire analysis duration—witnessed the lowest number of checkout attempts, and what was that exact figure?

- In the context of our project on checkout flow optimization, October is the month with the lowest number of checkout attempts, with a total of 98 checkout attempts recorded.

**Question 3:**

As we aim to refine the checkout flow and understand user behavior, pinpointing our peak moments can provide significant insights. From your constructed story, can you discern which of the listed months registered the highest count of purchase carts?

- The month with the highest count of purchase carts is November. In this month, the total number of created carts reached 2220.

**Question 4:**

Identifying cart abandonment patterns is crucial to effectively strategizing improvements in our checkout flow. Based on the narrative presented in your story, can you identify the two months that witnessed the most significant cart abandonment rates?

- August and October have the highest cart abandonment rates.

**Question 5:**

Understanding device-related error trends is crucial to finding and rectifying user issues during checkout. Can you determine the error message that most frequently appeared on desktop devices during September 2022?

-“year field is required” is the error message that most frequently appeared on desktop devices during September 2022 (32 times)

**Question 6:**

To ensure the accuracy of our actions moving forward, we must verify our insights against the data presented in the story. Referring to the narrative in your story, can you identify which of the following findings aligns with the information from the analysis?

- About what we sell, about the total number of carts created, about the cart abandonment rate and successful checkout rate

**Question 7:**

Opportunity sizing refers to estimating the potential value or impact of a particular opportunity or solution. In the context of business strategy and decision-making, it helps quantify the potential benefit of an initiative, giving a clearer picture of the stakes at hand.

For our Checkout Flow Optimization project, let's delve into opportunity sizing:

* In January, the checkout success rate was 34% out of 360 attempts.
* Each successful purchase generates approximately $30 in revenue.
* We aim to improve the checkout success rate to 40% in February.

For this exercise, assume that the number of checkout attempts in February will remain consistent with January (360 attempts). Given this assumption, calculate the opportunity size. Compared to January's earnings, how much additional revenue could the company earn in February if we achieve the targeted 40% checkout success rate?

- Additional revenue = (0.4-0.34)\*360\* $30 = $648