

CUSTOMER SUPPORT TICKET SUMMARY ANALYSIS

Prepared by: MD. Farhan Sadik (Intern)

Date: 30/06/2025

Table of Contents

Executive Summary	2
Methodology	3
Findings	4
Issue category summary	7
Root-Cause Analysis	8
Implementation Roadmap	11
Appendix	11

Executive Summary

- **Total number of tickets in dataset:** 8469
- **Total number of tickets analyzed:** 5650
- **Key findings from tickets analyzed:**

- Top 3 ticket type volume:

Ticket Type	Ticket Volume (%)
Refund request	21.0
Technical issue	20.3
Cancellation request	19.7

- Avg. first response time: 12.0 hours
- Avg. resolution time: 5.9 hours
- Avg. customer satisfaction score: 3.0

- **Primary root causes identified:**

- Generic inputs in forms leading to misclassified tickets
- Lack of escalation triggers for critical issues
- Missing resolution documentation and QA

- **Top Recommendations for improvements:**

- Require customers to select the issue type when submitting a ticket
- Automatically tag tickets based on keywords and escalate when needed
- Use standard response templates and review ticket closures

Methodology

1. Dataset description

1.1 **Source:** kaggle.com/datasets/suraj520/customer-support-ticket-dataset

1.2 **Total rows:** 8469, **Total rows analyzed:** 5650

1.3 **Total columns:** 17, **Total columns after analysis:** 26

1. Column names of original dataset: Ticket ID, Customer Name, Customer Email, Customer Age, Customer Gender, Product Purchased, Date of Purchase, Month of Purchase, Year of Purchase, Month-Year of Purchase, Ticket Type, Ticket Subject, Ticket Description, Ticket Status, Resolution, Ticket Priority, Ticket Channel, First Response Time, Time to Resolution, Customer Satisfaction Rating

1.4 Time-range of the dataset analyzed: January 2020 to December 2021

2. Tools and Techniques

2.1 **Excel: Pivot tables** and charts for initial data exploration, trend checks and some categorical data visualization

2.2 **Python:** Using Pandas library for data cleaning, aggregation, computation of important attributes such as “First Response Hrs” & “Resolution Hrs”. Using Matplotlib and Seaborn for visualizing trends, distributions, and correlations.

2.3 **NLTK:** Used to perform NLP tasks such as tokenization and keyword analysis for root-cause identification

2.4 **5-Whys:** For a structured root cause analysis of ticket type

Findings

- Ticket Volume & SLA Trends

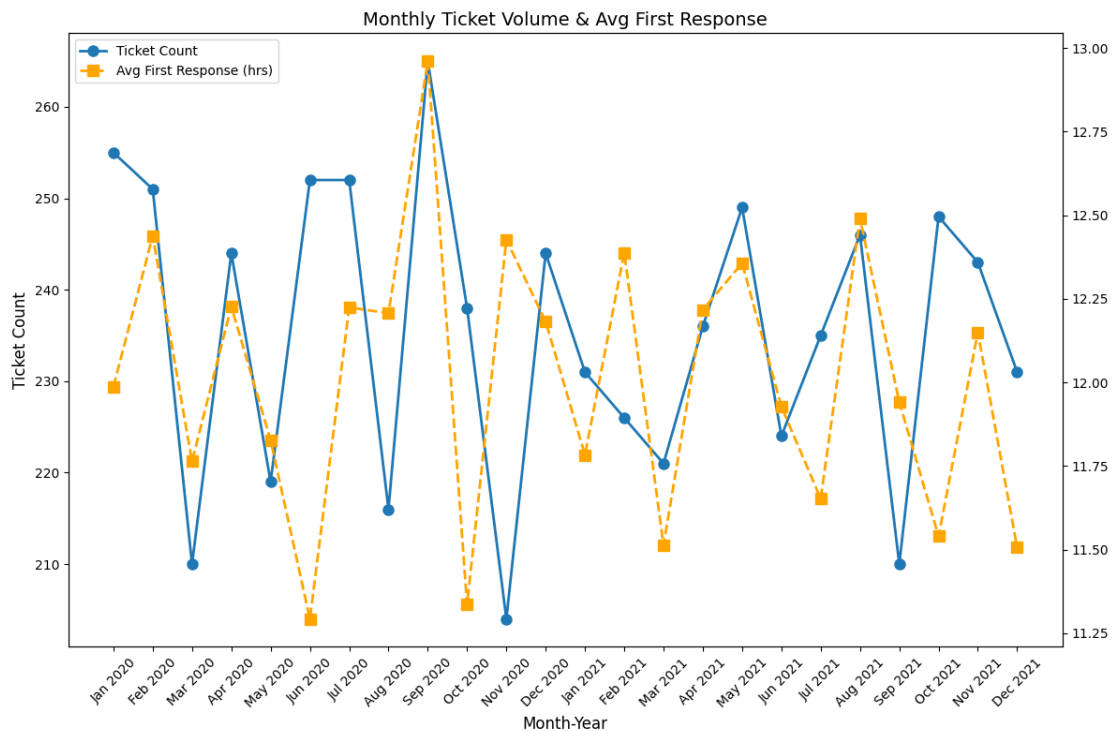


Fig. 1: Ticket volume and avg. first response over time

The monthly ticket volume and average first response hours from January 2020 to December 2021 as shown in Fig. 1 which shows a volatile and challenging environment. There is a clear positive correlation between the monthly ticket count and the average first response time, most notably in September 2020, when the highest ticket volume of approximately 268 coincided with the slowest average first response time of 13 hours. Higher ticket volumes generally indicate slow responses except for some anomalies such as November 2020, where despite ticket volume being at their lowest, the response time remained high. This suggests that external factors might be (i.e., staffing issues, ticket complexity) are also influencing response times.

Assuming a standard 12-hour Service Level Agreement (SLA) for first response, the data indicates frequent breaches, with the average response time often exceeding this threshold, highlighting a consistent struggle to meet service targets, especially during periods of high demand.

- **Channel & Priority Breakdown**

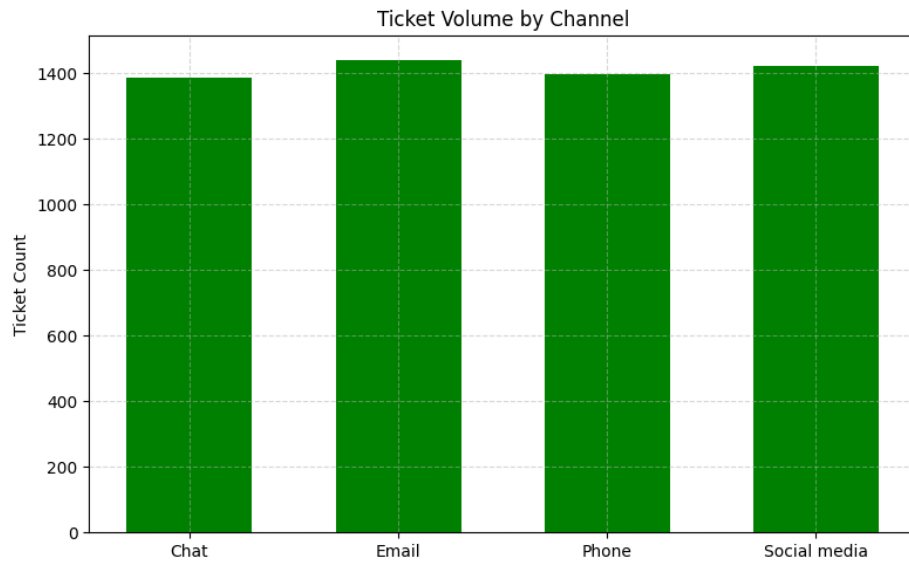


Fig. 2: Bar chart of ticket volume by channel

The bar chart in Fig. 2 describes how much tickets are coming through 4 different communication channels. The most prominent feature of the chart is the remarkably even distribution of ticket volume across all four channels, with each generating a nearly identical number of tickets. More than 1400 tickets are being received through emails and social media platforms while through support contact lines such as phones received 1400 tickets. Chat has the lowest volume of the four, but is still very close with approximately 1380 tickets. From a business or operational standpoint, it is crucial to maintain consistent staffing, resources, and quality across all four channels, as each one represents a major point of contact for customers. No single channel can be prioritized at the expense of another.

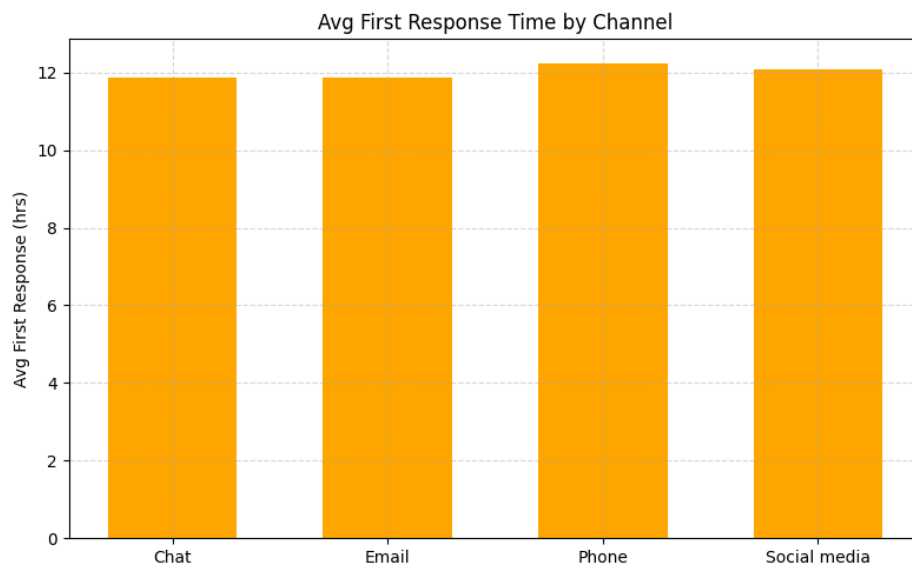


Fig. 3: Bar chart of avg. response times by channel

The bar chart in Fig. 3 shows the average first response times in hours across 4 different communication channels: chat, email, phone, social media. All four channels show very similar average first response times, hovering just under 12 hours, with phones appearing to have a slightly higher average response time compared to chat, emails and social media which are all nearly identical. Needless to say, first responses on average take almost the same time across all channels are around 12 hours so that can be assumed to be the standard SLA (Service Level Agreement) for support.

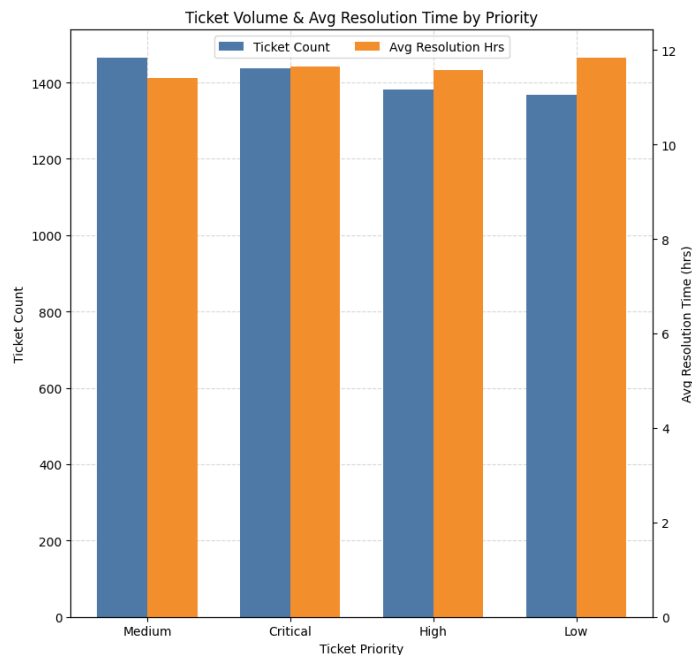


Fig. 4: Bar chart of ticket priority by average resolution hours

This bar chart in Fig. 4 displays the number of tickets (Ticket Count) and the average resolution time in hours (Avg Resolution Hrs) for different ticket priorities: Medium, Critical, High, and Low.

While there is not a direct inverse relationship between ticket volume and average resolution time across all categories. For instance, tickets prioritized as "Medium" and "Critical" have high volumes, 1400 tickets each but relatively lower resolution times. Conversely, "Low" priority tickets have the highest average resolution time despite having the lowest volume. This indicates that low-priority tickets might not be addressed with the same urgency, leading to longer resolution periods.

"High" priority tickets, while having a moderate volume, also show a high average resolution time, indicating potential challenges in resolving these tickets promptly despite their higher urgency than medium or critical. The relatively lower resolution times for "Critical" and "Medium" priority tickets, despite their high volumes, suggest that the system or team is effectively prioritizing and resolving these more urgent issues. This indicates good processes in place for handling high-impact problems.

Issue category summary

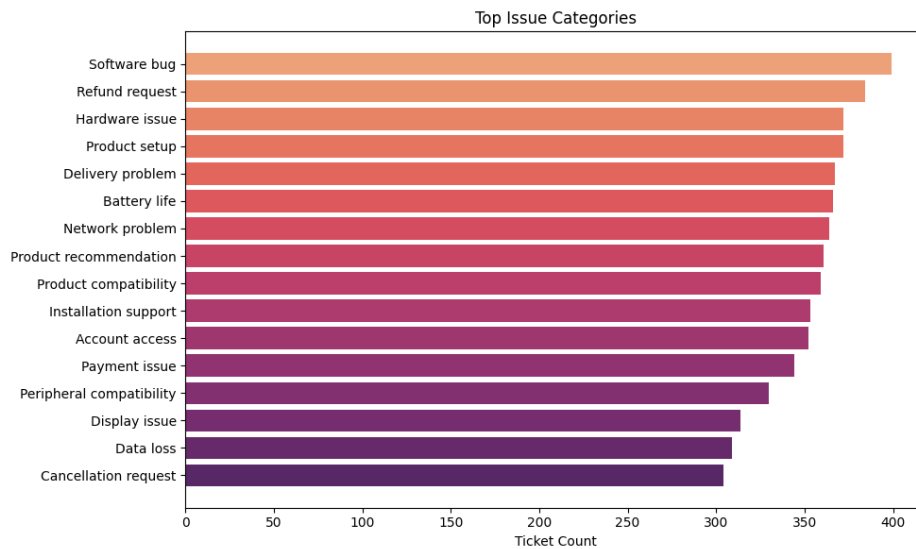


Fig. 5: Top issues categories by ticket count

The horizontal bar chart in Fig. 5 illustrates top issues faced by customer by tickets. “Software bug” seems to be the most common issue faced by customers as nearly 400 tickets describe software bugs. The second and third highest categories are “Refund requests” and “Hardware Issue” respectively. Tickets tagged as “Product setup” seem to be almost equal in quantity to “Hardware Issue” despite being fourth highest issue implying it might the number of tickets might be offset by a very small quantity.

Ticket Type	tickets	pct_total	avg_resp_hrs	avg_res_hrs	avg_csat
Refund request	1188	21.0%	12.21	11.65	2.96
Technical issue	1145	20.3%	12.25	11.47	2.97
Cancellation request	1113	19.7%	11.85	11.64	3.01
Product inquiry	1109	19.6%	11.74	11.81	3.00
Billing inquiry	1095	19.4%	12.03	11.51	3.01

Table 1: Summary Table

Table 1 shows an aggregated summary table showing number of tickets, number of tickets in percentage, avg. response hours, avg. resolution hours and the average customer satisfaction score by different ticket types (separate from issues): Refund requests, Technical issue, Cancellation request, Product inquiry and Biling inquiry.

Root-Cause Analysis

5 random samples of top 3 ticket types were taken from the batch of records being analyzed and they were used to perform the root-cause analysis using 5-Whys analysis.

- **Refund request tickets**

Common Patterns: Misclassification, Lacking Resolution, Customer Tone

5-Whys Summary:

Why	Reason
Why open a refund request ticket	Because their software keeps crashing and it disrupts their work.
Why is the software crashing during a specific action?	Possibly due to bugs introduced in a recent update.
Why hasn't this bug been addressed/resolved yet?	There may be no fix documented or no known patch.
Why wasn't the issue triaged or escalated properly?	Possibly due to inconsistent ticket labelling or lack of technical detail in response workflows.
Why was the refund ticket label used for non-refund bugs	Possibly due to users mislabeling tickets or wrong classification by agents.

Root Cause and Recommendations

Root Cause	Recommendation
Mislabeling of tickets	Introduce keyword-based auto-tagging (e.g., "crash", "bug", "update")
Missing or poor-quality resolutions	Implement mandatory resolution fields with review before closing
Unclear refund criteria or policy enforcement	Create predefined templates or escalation paths for technical issues
No automated triaging or tagging process	Add visible refund policy plus decision tree for support agents

- **Technical issue tickets**

Common Patterns: Generic descriptions, Software vs. hardware confusion, Missing or incoherent resolutions, No escalation triggers, No feedback loop into documentation

5-Whys Summary:

Why	Reason
Why did the customer open a "Technical Issue" ticket?	They experienced a problem with their device that persisted even after a factory reset.
Why didn't the factory reset fix their issue?	Because the root cause isn't in user-configurable settings; it's likely a deeper software or hardware fault.

Why isn't there clearer guidance on non-reset fixes?

Why is the knowledge base so sparse on this error?

Why isn't support feeding back recurring issues into documentation?

The knowledge base only covers basic resets; there's no documented advanced troubleshooting for this error. Support docs might not have been updated since last release and there's no feedback loop from support to docs. There's no formal process or tool to capture "repeat-fail" tickets and auto-flag them for documentation.

Root Cause and Recommendations

Root Cause	Recommendation
Knowledge base of troubleshooting only covers basic resets	Expand documentation with "Advanced Troubleshooting" guides for non-reset errors
No feedback loop from support to docs	Implement a "Doc-flag" tag on tickets that survive Tier-1 triage, auto-notifying your documentation team
Gibberish or missing resolutions	Enforce a mandatory "Resolution Quality" review step before closing any technical ticket

- **Cancellation request tickets**

Common Patterns: Generic Issue Plea, Extraneous/irrelevant content, Ambiguous cancellation threat, Missing or gibberish resolutions

5-Whys Summary:

Why	Reason
Why open the ticket?	Because their device stopped working properly after the firmware update, and they fear it's unusable
Why is the updated firmware making the device unusable?	The new firmware introduced a bug or incompatibility that causes the product to malfunction.
Why was this buggy firmware shipped to customers?	Due to lack of sufficient compatibility testing on all supported hardware configurations before release.
Why wasn't broader hardware testing part of the release process?	The QA process focused only on core functionality and missed edge-case devices like this customer's model.
Why does the QA process miss those edge-cases without escalation from support?	Because there's no formal feedback loop or tagging system to surface "post-update failures" back to QA.

Root Cause and Recommendations

Root Cause	Recommendation
All tickets use a generic “help me” opener, obscuring real intent	Introduce a mandatory “Issue Type” selector (e.g. Cancellation vs. Support) in the intake form
Descriptions contain irrelevant noise (dates, tweets, sighs)	Implement guided prompts that only accept relevant fields (reason, affected feature)
Cancellation threats are tacked on after generic support requests	Require a structured “Cancellation Reason” dropdown when Cancellation is selected
Resolutions are missing or nonsensical, offering no closure	Enforce a QA-checked resolution template with predefined options and a one-sentence summary
No mechanism to surface post-update/security issues back to product	Auto-tag tickets mentioning “update” or “security” and route them to Product/QC for review

Implementation Roadmap

1. Roll out new intake form and pivot to structured fields
2. Deploy keyword-based tagging and escalation flows
3. Launch resolution templates and QA dashboard
4. Integrate feedback loop to documentation and product teams

Appendix

- Code Snippets (in /notebooks/ and /Scripts/ directory)
- PivotTables Screenshots (in /figures/ directory)
- Full Token Frequency Lists (in /Datasets/Processed/ directory)
- 5-Whys Worksheets (in /Datasets/Processed/ directory)
- Additional Figures (in /figures/ directory)