

Introduction (1 minute) Presenter 1 (You - Overall Analysis)

"Good [morning/afternoon], everyone. We are here today to present our analysis of GitHub issues. Our objective was to understand the patterns and trends related to the issues reported on GitHub, focusing on key metrics like issue creators, issue states, labels used, and time to resolution. We've divided our work into four major areas, and each of us will walk you through one of these aspects. Let's dive in.

We'll start by looking at the overall trends in the issues dataset."

Overall Analysis (3 minutes) Presenter 1 (You)

"We started by exploring some basic statistics and visualizations. First, we analyzed the total number of issues and events across the dataset. We found that there was a large number of issues, and interestingly, a high volume of events tied to each issue, which tells us that each issue likely goes through multiple stages and discussions.

To better understand the contributors, we visualized the **top 30 issue creators**. This allowed us to identify who's most active in creating issues in the repository.

We also examined the **issue state distribution**. GitHub issues can have different states, like 'open', 'closed', or 'in progress', and this chart helped us see the balance between these states. As you can imagine, understanding how many issues are open vs closed is key for project managers and contributors.

Then, we turned our attention to the **labels** used across issues. Labels are crucial because they categorize issues, helping developers quickly filter them. We looked at the top 10 labels and found some insightful trends. For instance, certain labels are more frequently associated with high-priority issues.

Finally, we analyzed the **time to resolution** for closed issues. By calculating how long it takes to resolve issues, we could see how efficiently the community is addressing problems. We'll discuss more on this in a moment, but it was fascinating to see how long some issues took to be closed, with some lingering for much longer than others."

Analysis of Labels and Issue Creators (4 minutes) Presenter 2 (Teammate 2)

"Now that we've seen the high-level view, let's dig deeper into the relationship between **labels** and **issue creators**. In this part of the analysis, we focused on filtering issues based on labels or specific creators.

We used the ability to filter issues by creator and label to answer questions like: 'How often does a certain creator use specific labels?' and 'Which issues have been created by certain developers over time?'

One of the key aspects we focused on was the **number of issues opened vs. closed** for each creator and label. This helps us understand which users are actively resolving the issues they report. For example, we observed that some users frequently open new issues but have fewer closed ones, which could indicate that they are less involved in resolving them. By visualizing this data, we could easily compare the issue counts for each creator, segmented by labels, to uncover insights.

Another key visualization we created was the **frequency of labels used by each creator**. This was interesting because it allowed us to see what categories of issues certain creators are most involved in, and how those issues are distributed across different labels. This data can be very useful for understanding the areas of focus within a project."

Monthly Issue Analysis (3 minutes) Presenter 3 (Teammate 3)

"Building on the overall issue trends, we then focused on the **monthly creation of issues**. By tracking issues over time, we wanted to answer questions like: 'Are there any seasonal spikes in issue creation?' or 'Is there a particular time period when the team is more active in reporting issues?'

We took a deeper look at the **monthly issue counts**, creating time-series plots that showed how many issues were opened each month. This analysis gave us a clear picture of activity over time, and we found that there were some months with noticeable spikes in issue creation. This could be indicative of project releases or moments when the community is more engaged.

By analyzing these trends, we can not only track how the project evolves but also plan better for future work. This information can be used to predict when new issues are likely to arise, enabling better resource allocation."

Issue Close Time Analysis (3 minutes) Presenter 4 (Teammate 4)

"Now, let's take a look at the **time to close** analysis. We were particularly interested in understanding how quickly issues are being resolved once they are opened. By calculating the time between the creation and closure of an issue, we could assess the efficiency of issue resolution.

This is a critical metric because it tells us whether the project is addressing problems in a timely manner. We found that some issues were closed very quickly, while others took an unexpectedly long time to resolve.

To make this clearer, we plotted the distribution of issue close times and calculated the **average time to resolution**. This allowed us to understand the overall efficiency and identify any outliers—issues that are taking much longer than usual to resolve. By focusing on these outliers, the project team could investigate potential bottlenecks and work to speed up issue resolution.

Additionally, analyzing the time to close in relation to labels could help us identify which types of issues are resolved more quickly and which ones tend to linger, so we can improve the workflow."

Conclusion (1 minute) Presenter 1 (You)

"To wrap up, our analysis provided a comprehensive view of GitHub issues, with insights into the contributions of different users, the types of issues reported, the efficiency of issue resolution, and trends over time. We hope this analysis helps the project team not only to understand the current state of the project but also to make data-driven decisions moving forward.

Thank you for your time, and we're happy to take any questions you may have."