

Issue analysis and reporting investigation

this topic provides a sample report on defects, as well as a description of possible charts on which the report will be compiled.

The report on defect analysis (Issue Analysis Report) is an essential component of the defect management process in a bug-tracking system like Atlassian Jira. This report is usually prepared by the Project Manager or Quality Assurance (QA) specialists to analyze defects identified during testing or software development. The main objective of this report is to ensure product quality, identify potential issues, and take appropriate actions to rectify the defects.

Issue report.

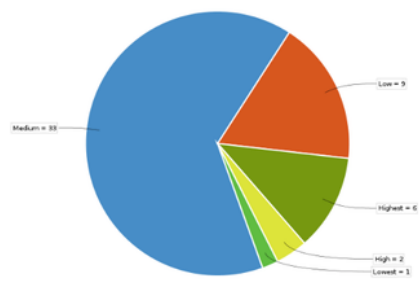
1. **General Information:** This section provides general project details such as the project name, software version, report preparation date, and the name of the person who prepared the report.
2. **Defects Statistics:** In this section, numerical metrics are presented, including the total number of registered defects, the number of new defects, the number of resolved defects, and the number of open defects.
3. **Analyzed Defects:** This section lists all the defects that have been analyzed. Each defect may have the following attributes:
 - **Defect ID:** A unique identifier of the defect in the bug tracking system.
 - **Defect Name:** A brief description of the problem or issue.
 - **Priority:** The importance level of the defect (e.g., low, medium, high).
 - **Status:** The current status of the defect (open, fixed, verified, etc.).
 - **Assignee:** The name or identifier of the person responsible for fixing the defect.
 - **Root Cause:** An explanation of the reason for the occurrence of the defect or its root cause.
 - **Planned Actions:** Steps to be taken to rectify the defect.
4. **Trends:** This section may include graphs or charts that depict the trends of defect changes over a specific period. It helps identify whether the product's state improves with each new release or if the number of reported defects increases.
5. **Conclusions:** This section summarizes the overall conclusions drawn from the defect analysis, highlighting the main issues identified, successful actions taken to address the defects and recommendations for improving the development process.
6. **Recommendations:** Specific recommendations for further enhancing product quality are provided in this section. These may include suggestions for improving the testing process, enhancing the team's skills, or implementing test automation.

The Issue Analysis Report facilitates transparency and effectiveness in the defect management process. It provides the project team with a clear picture of the product's status, helps identify issues in a timely manner, and enables informed decisions to improve software quality.

In our test report we can use the following charts:

- [Pie Chart Report](#) - тут ми можемо аналізувати issues відповідно до пріоритету , версії та компоненту(платформи) і так далі.
Наприклад - кількість багів та їх пріоритет по проекту Quran, IOS platform, Tablet release.

Chart

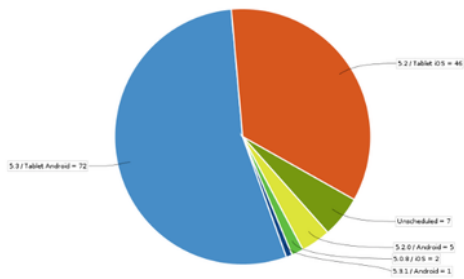


Data Table

	Issues	%
Medium	33	64%
Low	9	17%
Highest	6	11%
High	2	3%
Lowest	1	1%

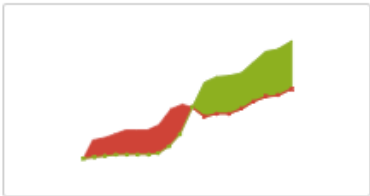
також можемо виводити данні по багам та їх пріоритету в розрізі платформ Android + IOS tablet release.

Chart



Data Table

	Issues	%
5.3 / Tablet Android	72	54%
5.2 / Tablet iOS	46	34%
Unscheduled	7	5%
5.2.0 / Android	5	3%
5.0.8 / iOS	2	1%
5.3.1 / Android	1	0%



Created vs. Resolved Issues Report

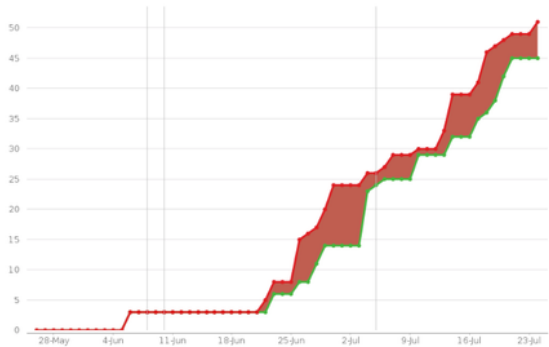
Maps created issues versus resolved issues over a period of time. This can help you understand whether your overall backlog is growing or shrinking.

- [Created vs. Resolved Issues Report](#) - за допомогою цього графіку можна вивести данні по к-сті баг відкриті+закриті по конкретній платформі за певни період часу. IOS platform tablet release.

Filter: [Test filter for Quran project](#)

Chart

This chart shows the number of issues **created** vs. the number of issues **resolved** in the last 60 days.



Data Table

Period	Created	Resolved
26-May-2023	0	0
27-May-2023	0	0
28-May-2023	0	0
29-May-2023	0	0
30-May-2023	0	0

Всі графіки ми можемо будувати на основі дефолтних фільтрів або кастомних(збережених).

Обов'язково потрібно пріоритизувати баги по єдиному стандарту

UI - low

Functional issue - mid

Crashes/nonworking feature - high