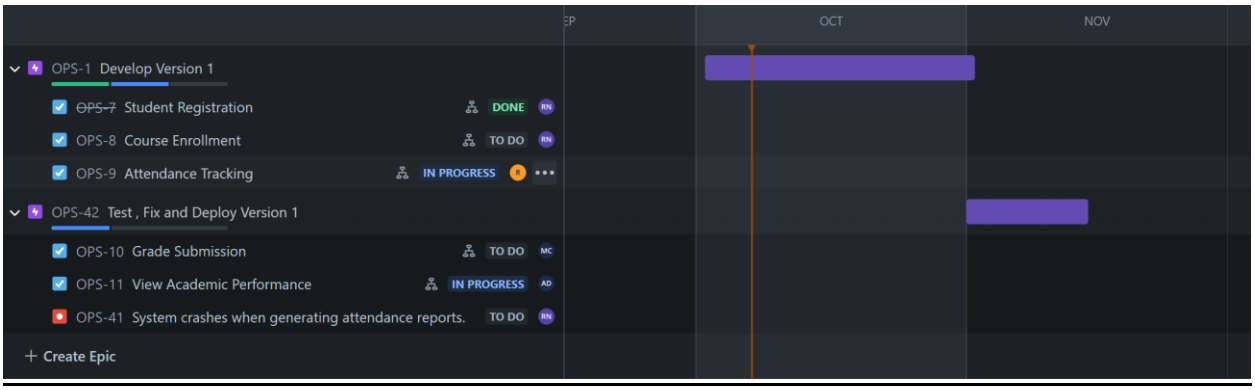
JIRA EXPLORATION ASSIGNMENT

Name: Raghav Narayan Ramachandran

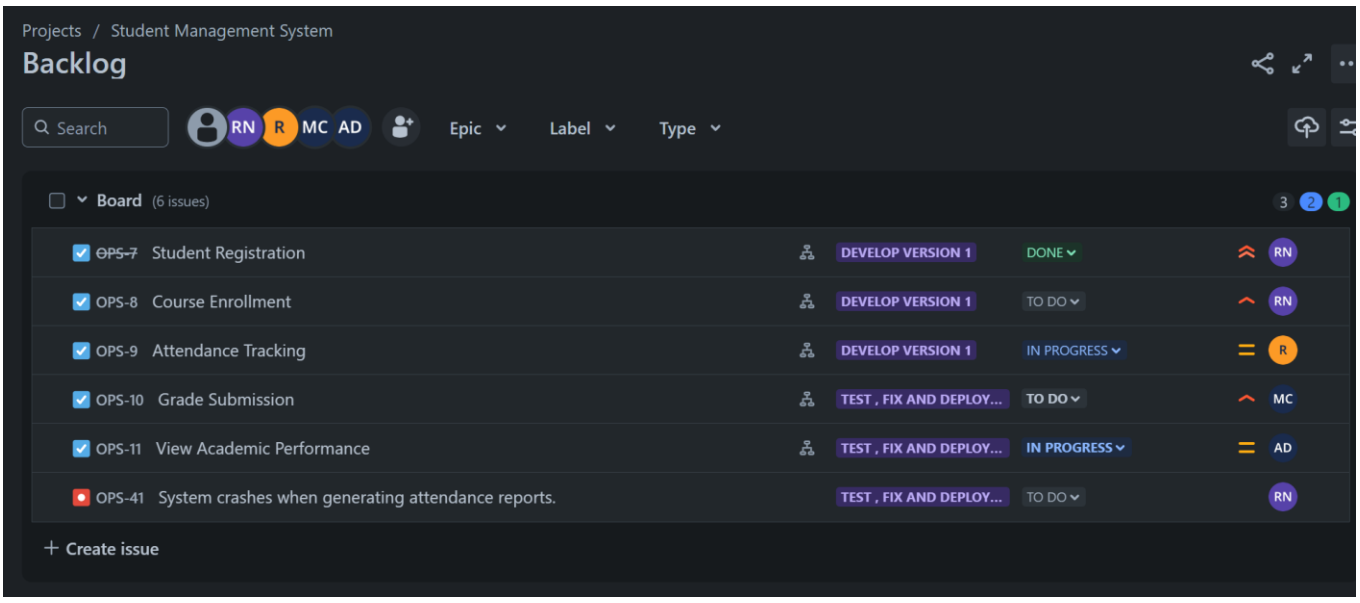
ID: 1002140654

PROJECT SETUP:

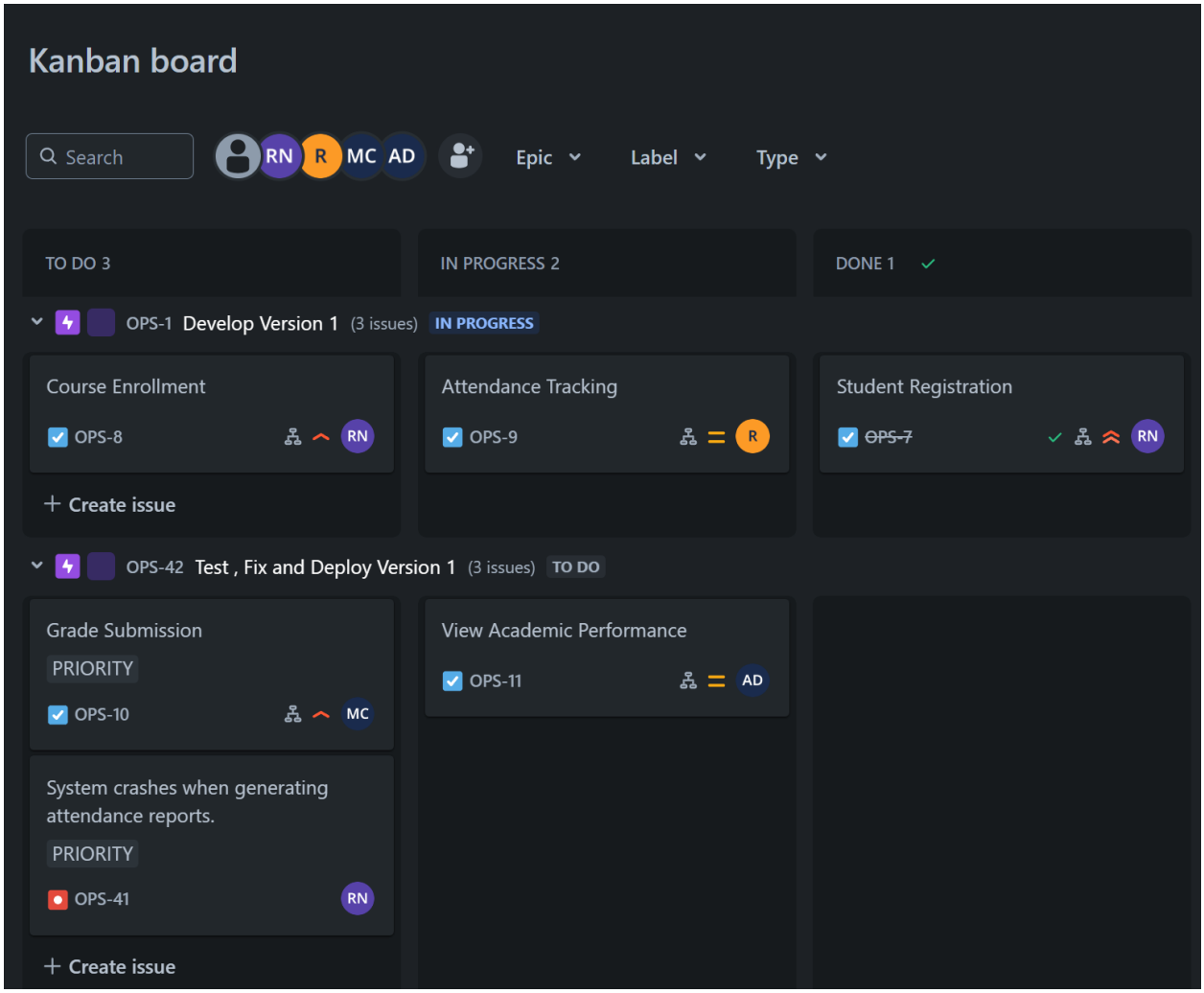
TIMELINE:



BACKLOG:



Kanban Board showing the project overview task progression:

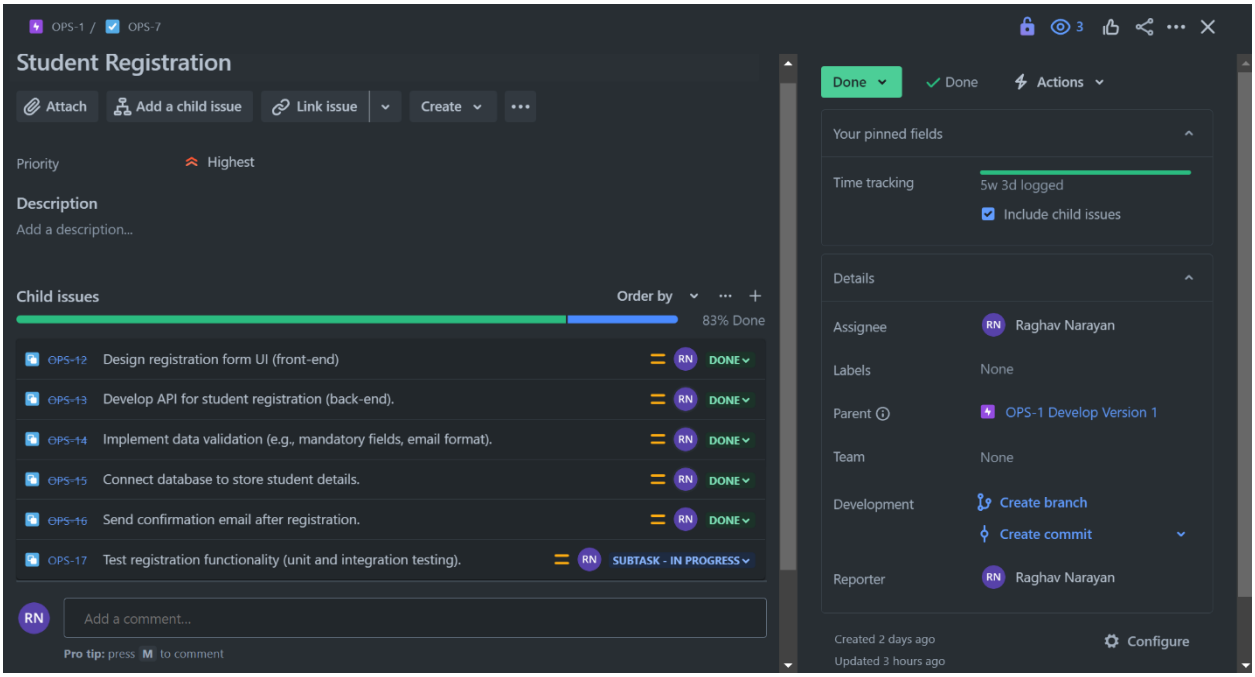


USER STORIES:

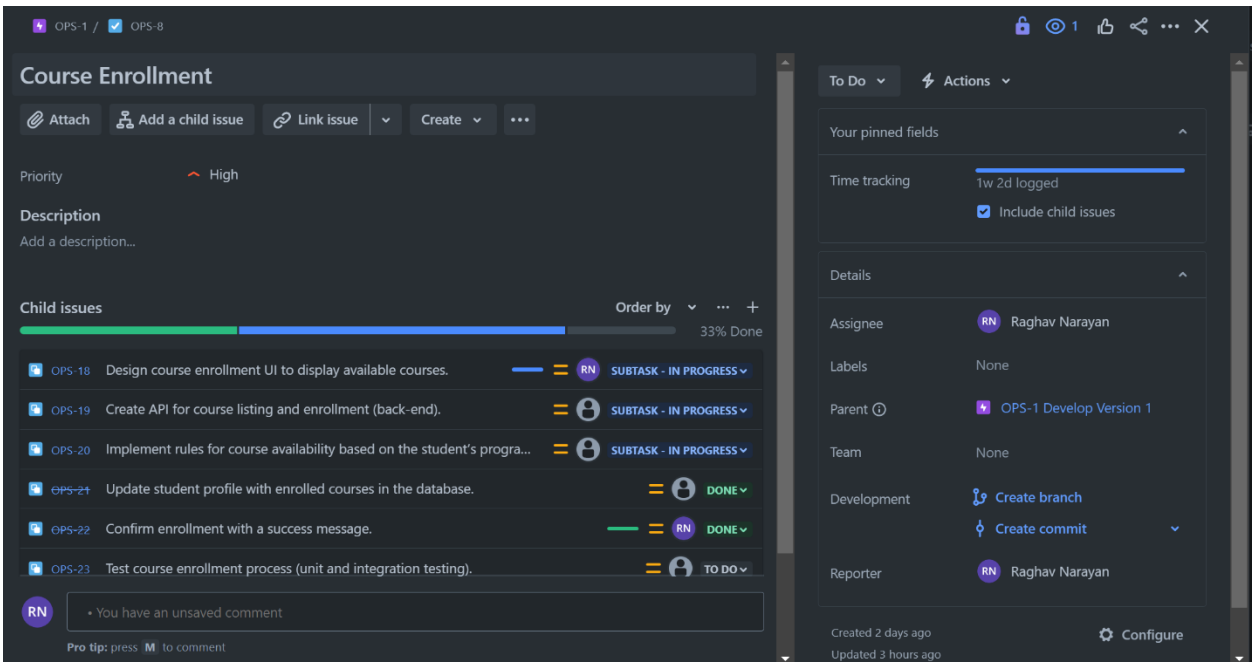
- Student Registration
- Course Enrollment
- Attendance Tracking
- Grade Submission
- View Academic Performance

USER STORIES SCREENSHOTS

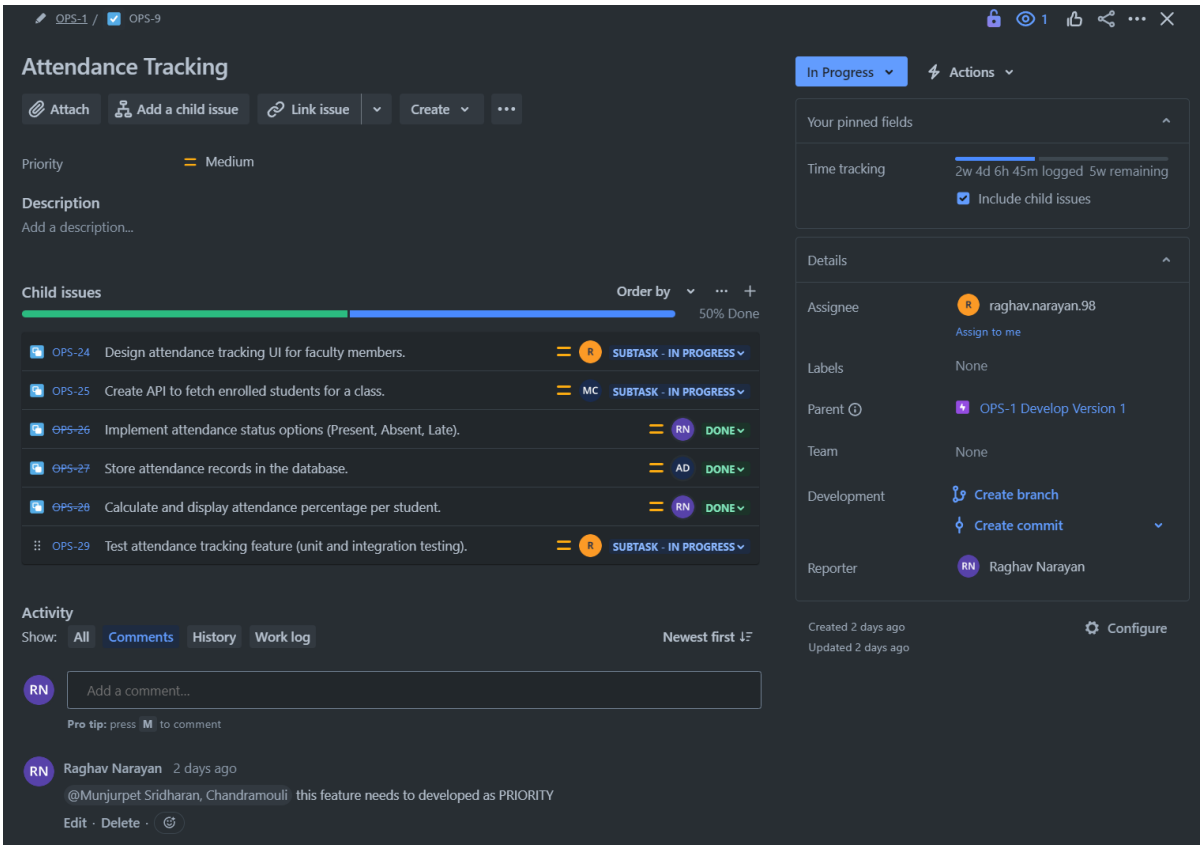
- Student Registration



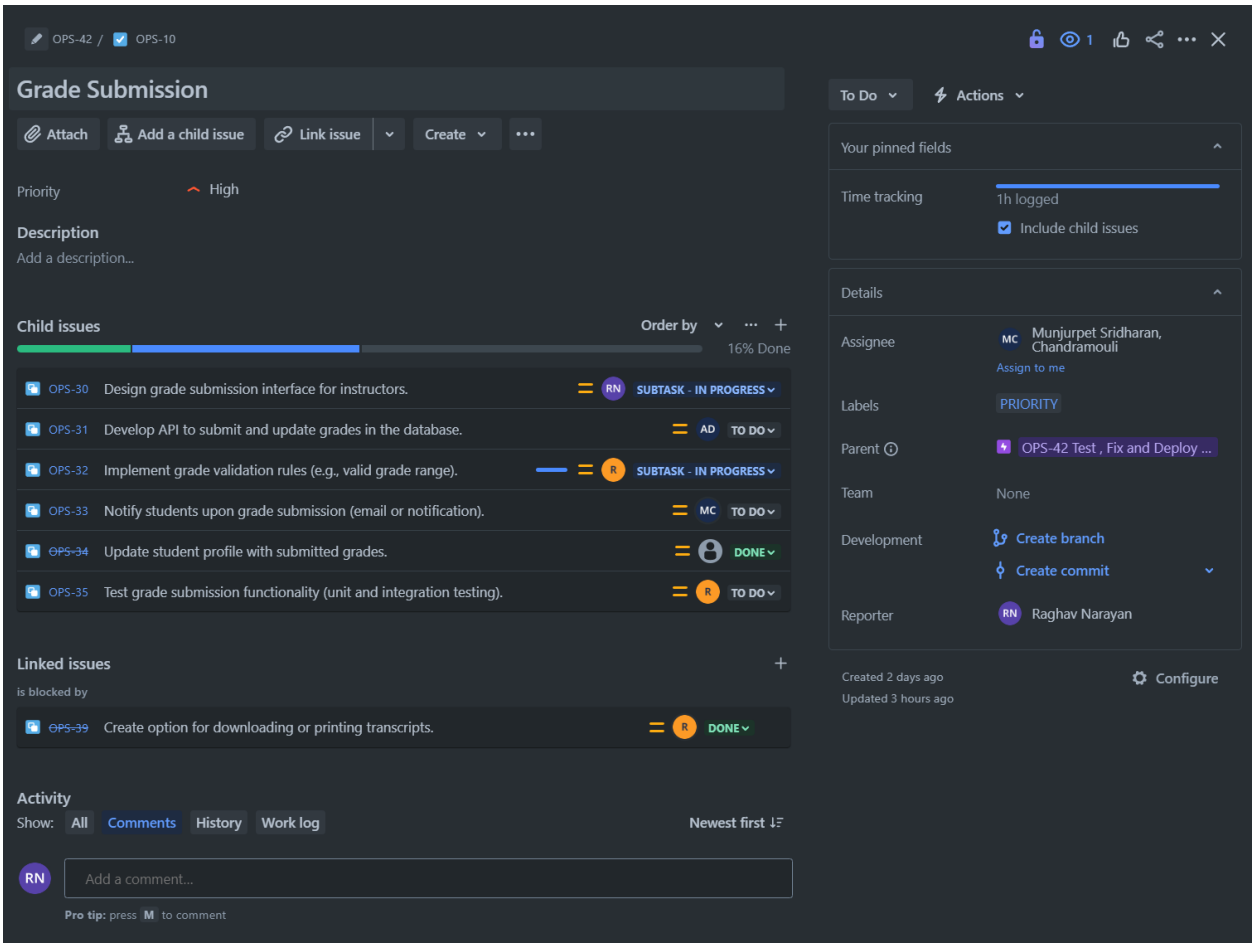
- Course Enrollment



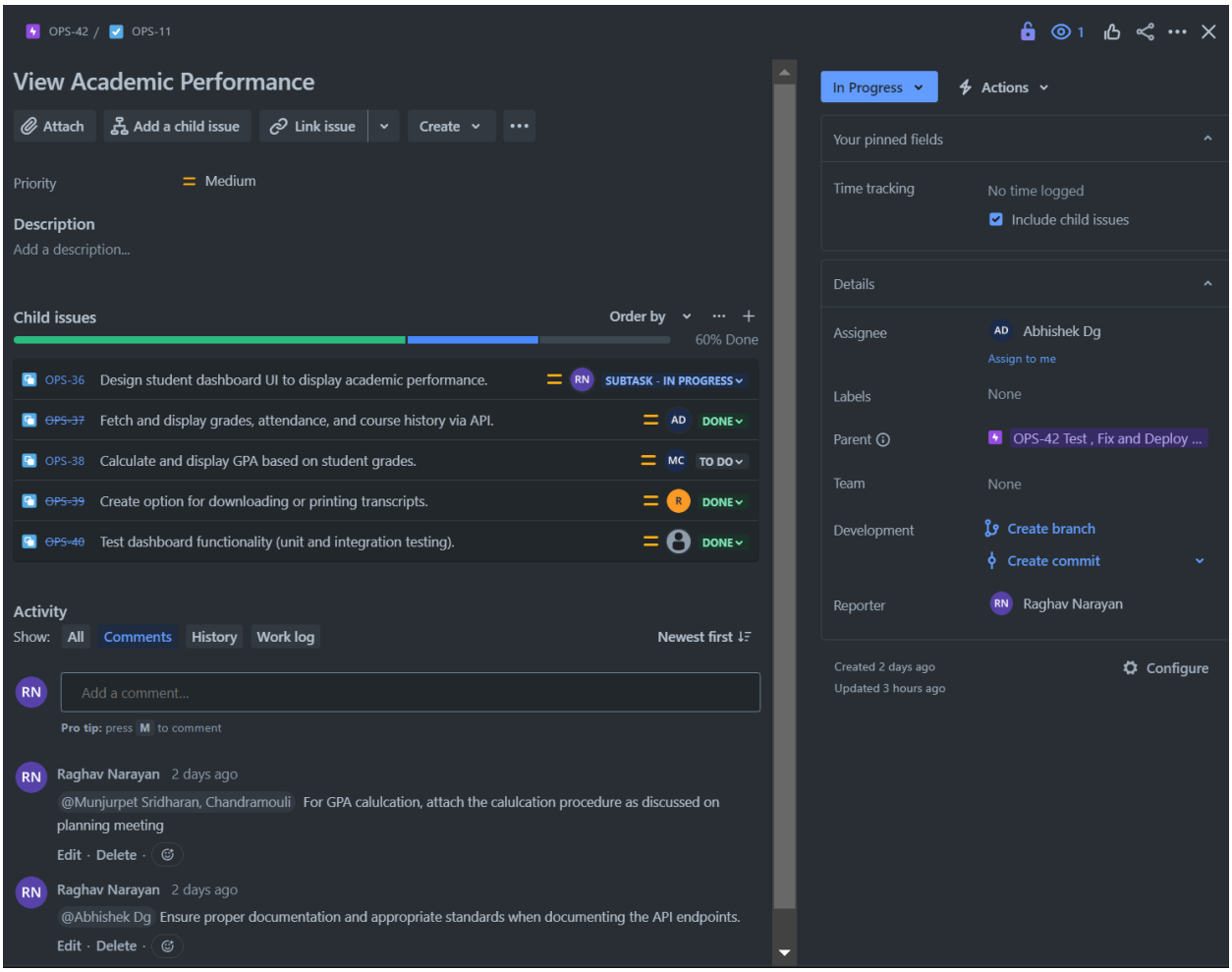
- Attendance Tracking



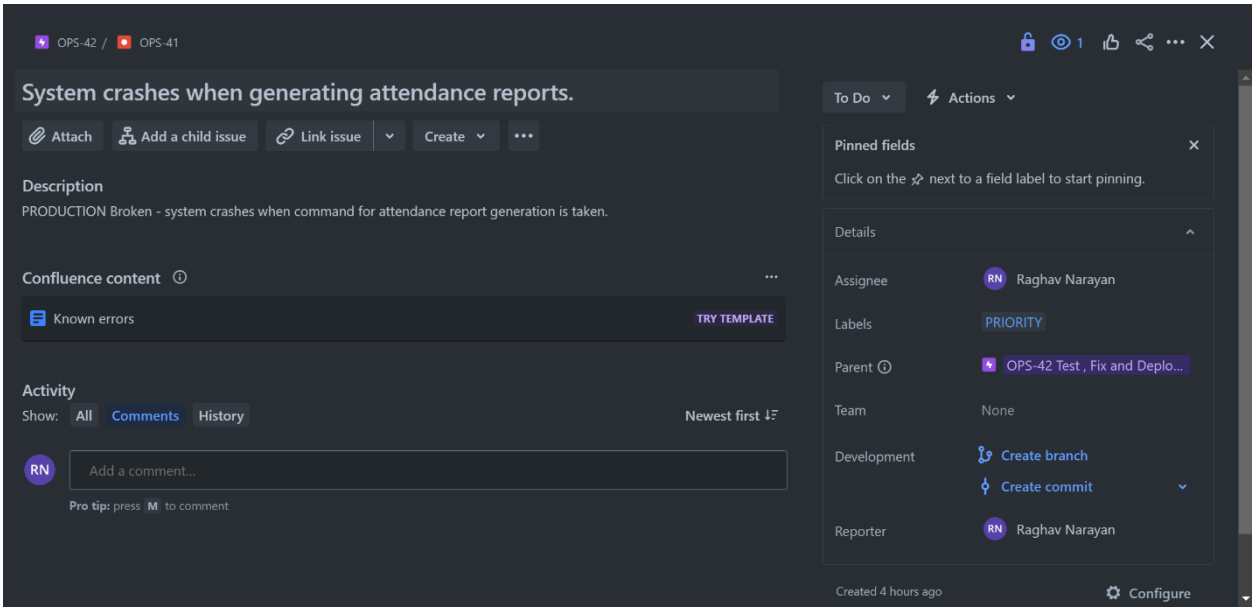
- Grade Submission



- View Academic Performance

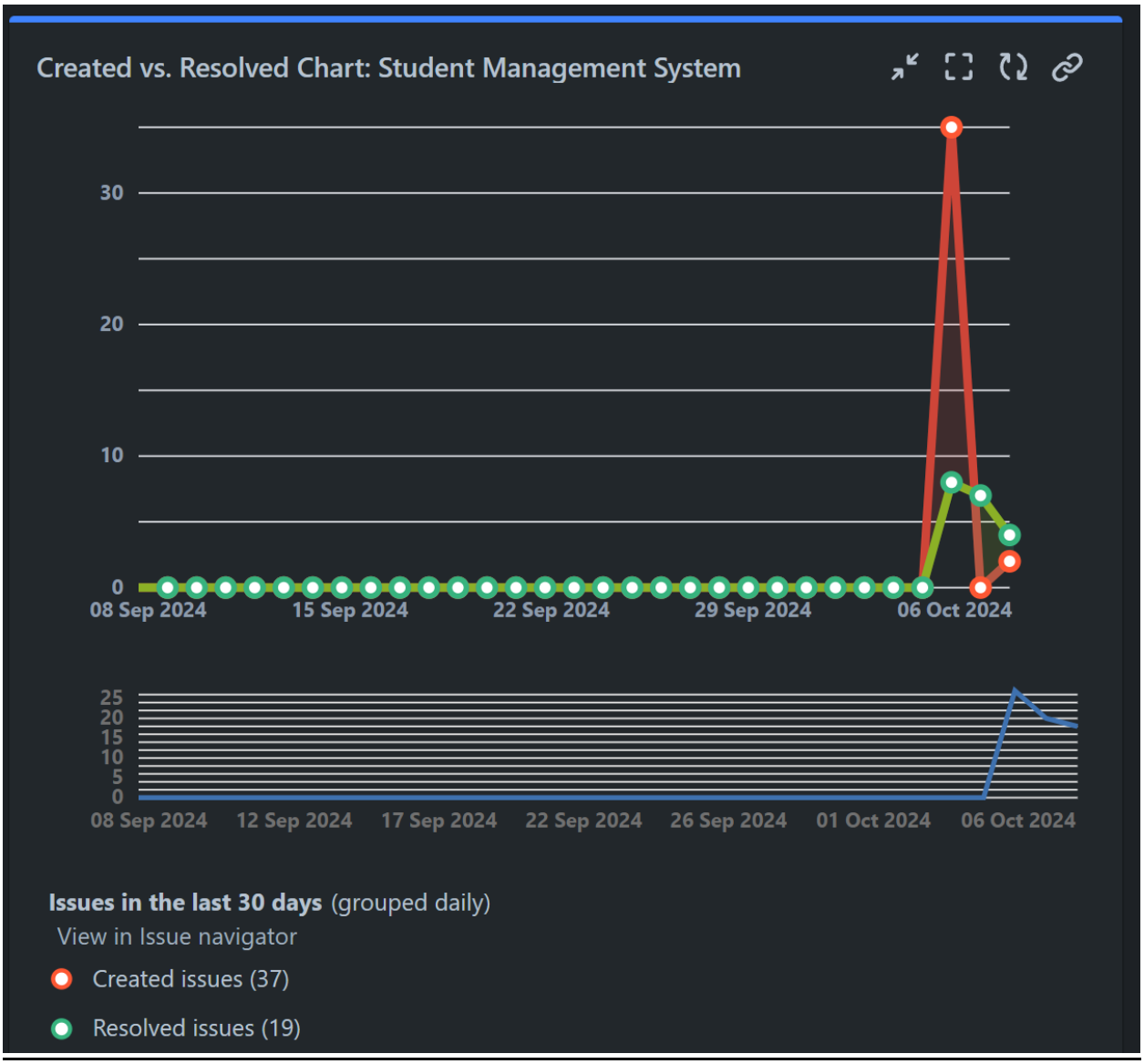


- PRODUCTION BUG



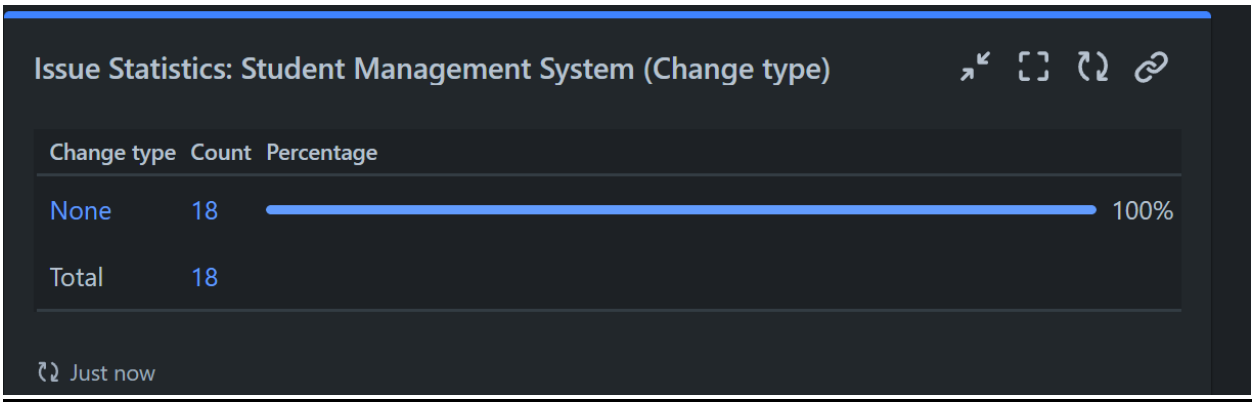
REPORT GENERATION:

Created vs. Resolved Chart:

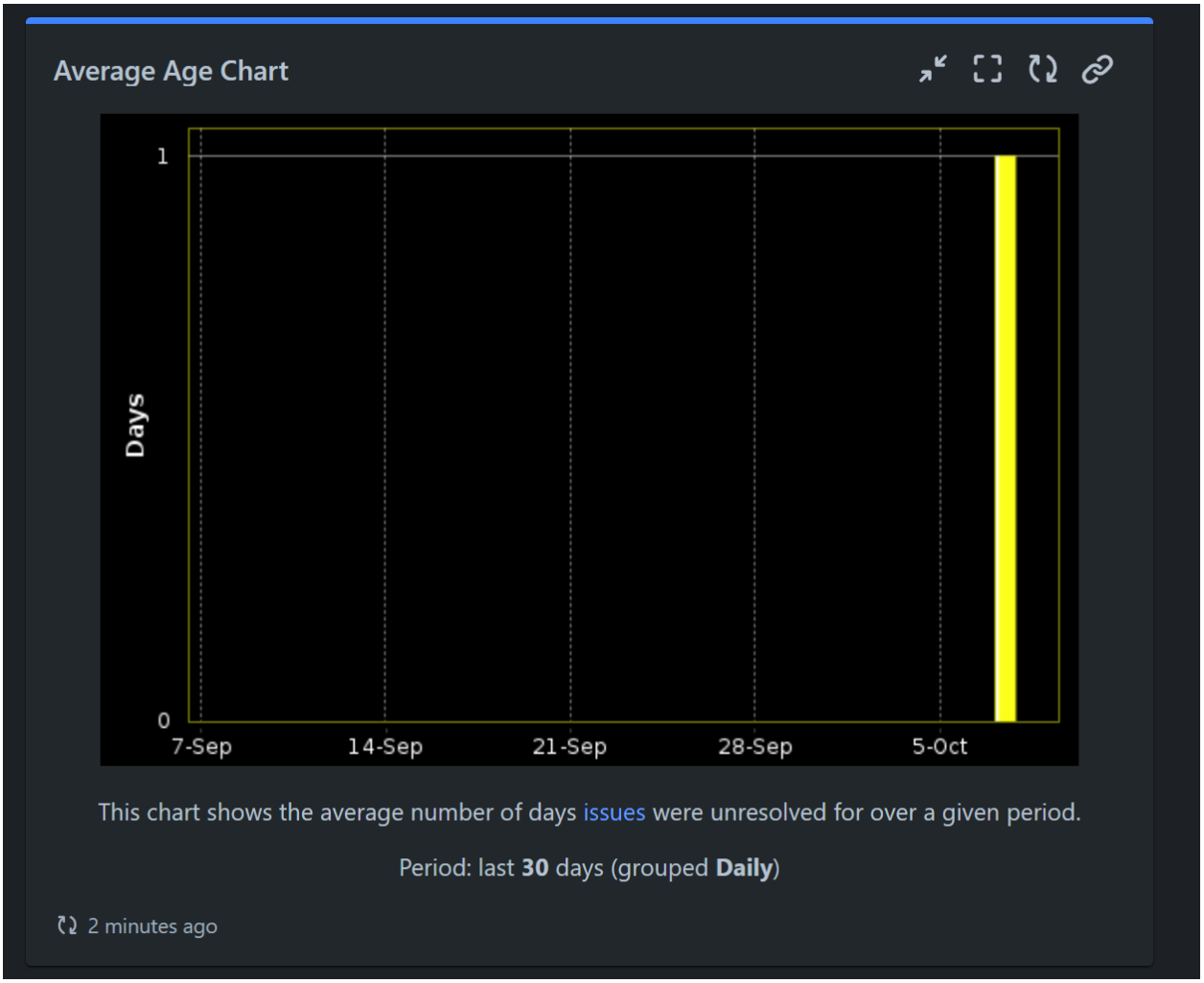


A significant spike in created issues occurred around October 5th. This indicates a potential surge in student or administrative activity during this period. While there was a general increase in resolved issues following the spike in created issues, there appears to be a lag in resolving them. This might suggest a need for additional resources or process improvements to address the backlog.

Issue Statistics (Change Type):

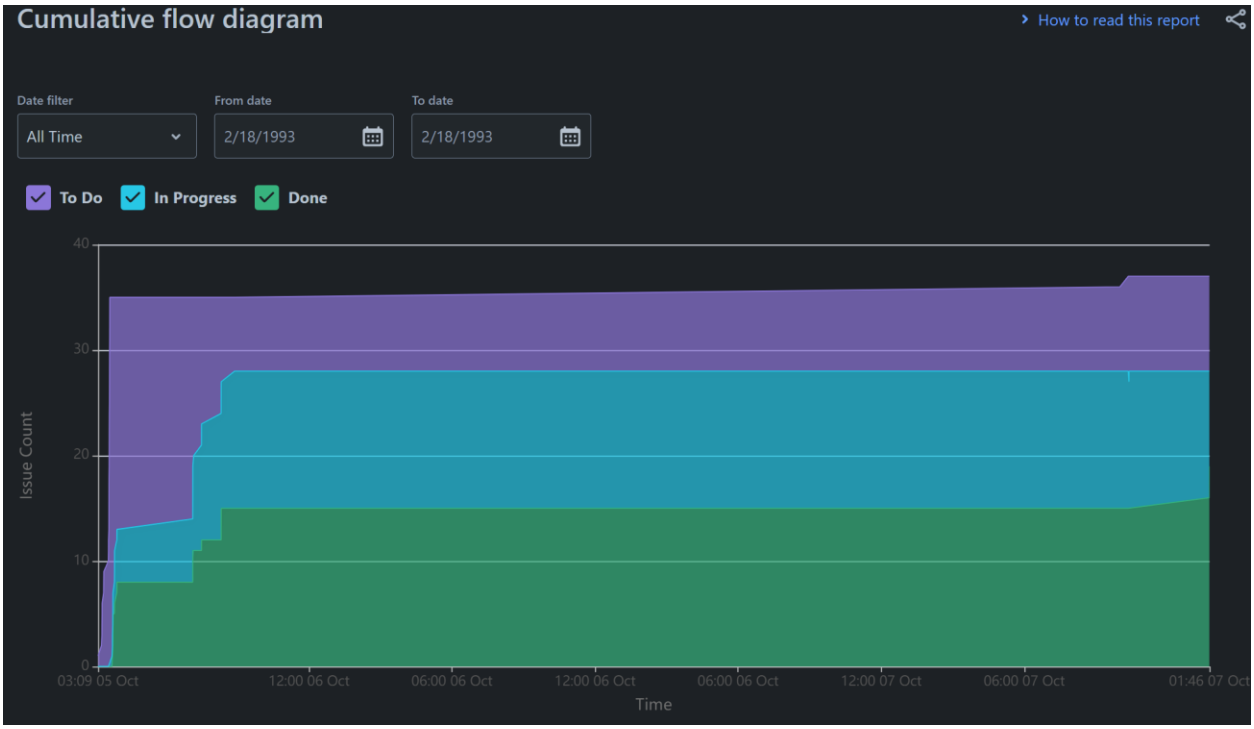


Average Age Chart:



A very high spike in the average age of unresolved issues is evident around October 5th. This indicates a significant backlog of unresolved issues during this period. Following the spike, the average age of unresolved issues drops dramatically. This suggests that there was a concerted effort to address the backlog and resolve outstanding issues

Cumulative Flow Diagram:



The chart shows a significant initial backlog of issues in the "To Do" state. Over time, the "To Do" line gradually decreases, indicating that issues are being moved into the "In Progress" state. The "In Progress" line shows a steady increase, suggesting that work is being actively done on the issues. The "Done" line shows a gradual increase, indicating that issues are being completed and moved to the "Done" state.

REFLECTIONS ON THE EXPERIENCE:

Helpful Features in JIRA:

1. **Kanban Board:**
   One of the most intuitive features in JIRA is the Kanban board. This allows for a clear visualization of tasks and their current status. It breaks down the epics or user stories into columns such as **To Do, In Progress** and **Done.** This effectively helps to track the flow of tasks and understand overall progress of a project. This feature keeps the teams organized and allow individual developers or stakeholders to identify tasks that require immediate attention.

2. **Task Assignment and Tracking:**

   JIRA provides a simple method to assign tasks to different team members and track progress. The ability to comment on tasks, log work hours, and update the status in real time helps simulate a collaborative environment where team members can communicate on each issue, epic or task or sub task. This ultimately brings in accountability and full transparency throughout the team, as each individual can see who is tasked with which deliverable and how far they have progressed.

3. **Customizable Workflow:**

   Another great aspect about JIRA is that it lets you adjust workflows to suit the requirements and the way the team works. For instance, columns can be easily added or removed in the Kanban board to reflect the team's development progress. Limits can also be set on tasks that are work in progress, which helps to avoid overloading the team. This capacity makes it effectively to adapt JIRA to the needs of different types of projects, so that every individual of the team stays on track and tasks flow smoothly.

Challenges Faced:

1. **Learning Curve:**

   One of the difficult challenges I faced while exploring JIRA was the initial learning curve. While I observed JIRA to be effective and powerful, it quickly became overwhelming for initial new users like me due to wide arsenal of features. Comprehending how to customize boards, configuring the settings and organize tasks was not exactly straight forward and took some time to understand. I had to look through different web pages, documentation and tutorials to get familiar.

2. **Complexity of Advanced Settings:**

   Another challenge I faced in navigating JIRA was the exploration of the advanced features such as enabling settings like time tracking, managing user roles and permissions. These features were not enabled by default to every user. Understanding how to manage user roles and permissions for different team members took significant effort. These settings required admin privileges and navigating the right permissions to the required users required deep exploration, which made these aspects very challenging compared to the basic functions.

Benefits in Real-World Software Development

1. **Improved Collaboration and Transparency:**

   JIRA's centralized platform improves collaboration within the team by enabling everyone to have real-time clear visibility on each task mentioned in the board and their progress. With all tasks in one place, members can easily comment, mention others, log their work and get notifications on updates. This transparency is helpful in software development, where many epics or stories are blocked or dependent on other stories or sub tasks. Having clear visibility to see progress on any tasks helps to identify potential issues early and keep things moving smoothly without any major setbacks.

2. **Efficient Task Management:**

   Handling a software development project requires keeping track of many tasks, from bug fixers, testing to new feature implementations. JIRA provides a structured way to manage these tasks by breaking them into smaller, manageable units. The capacity to prioritize tasks ensures that low-risk high value tasks are addressed contributed to better overall efficiency.

3. **Sprint Planning and Agile Practices:**

   Teams that follow Agile methodologies, JIRA offers a streamline method to mange sprints and keeps everyone on the team focused on sprint goals. By tracking the team's velocity and keeping a keen eye on the spring backlog, JIRA ensures software development is on track, delivering continuously and on an incremental basis.