**JIRA**

**Day 1 - Jira Core + Project Setup**

**Learnt**

1. What Jira is used for in real companies?

Jira is widely used in real companies to manage software development work, track bugs, plan releases, and collaborate across teams. It acts as a single source of truth where requirements, development tasks, testing activities, and delivery status are maintained.

2. Jira Software vs Service Management

Understood the difference between **Jira Software** (used mainly by development and QA teams for Agile projects) and **Jira Service Management** (used for IT support, ticketing, and service requests).

3. Scrum vs Kanban (when to use what)

Learned when to use **Scrum** (fixed sprint cycles, planned work, product‑based teams) versus **Kanban** (continuous flow, operational/support work, no fixed sprints).

**Practical**

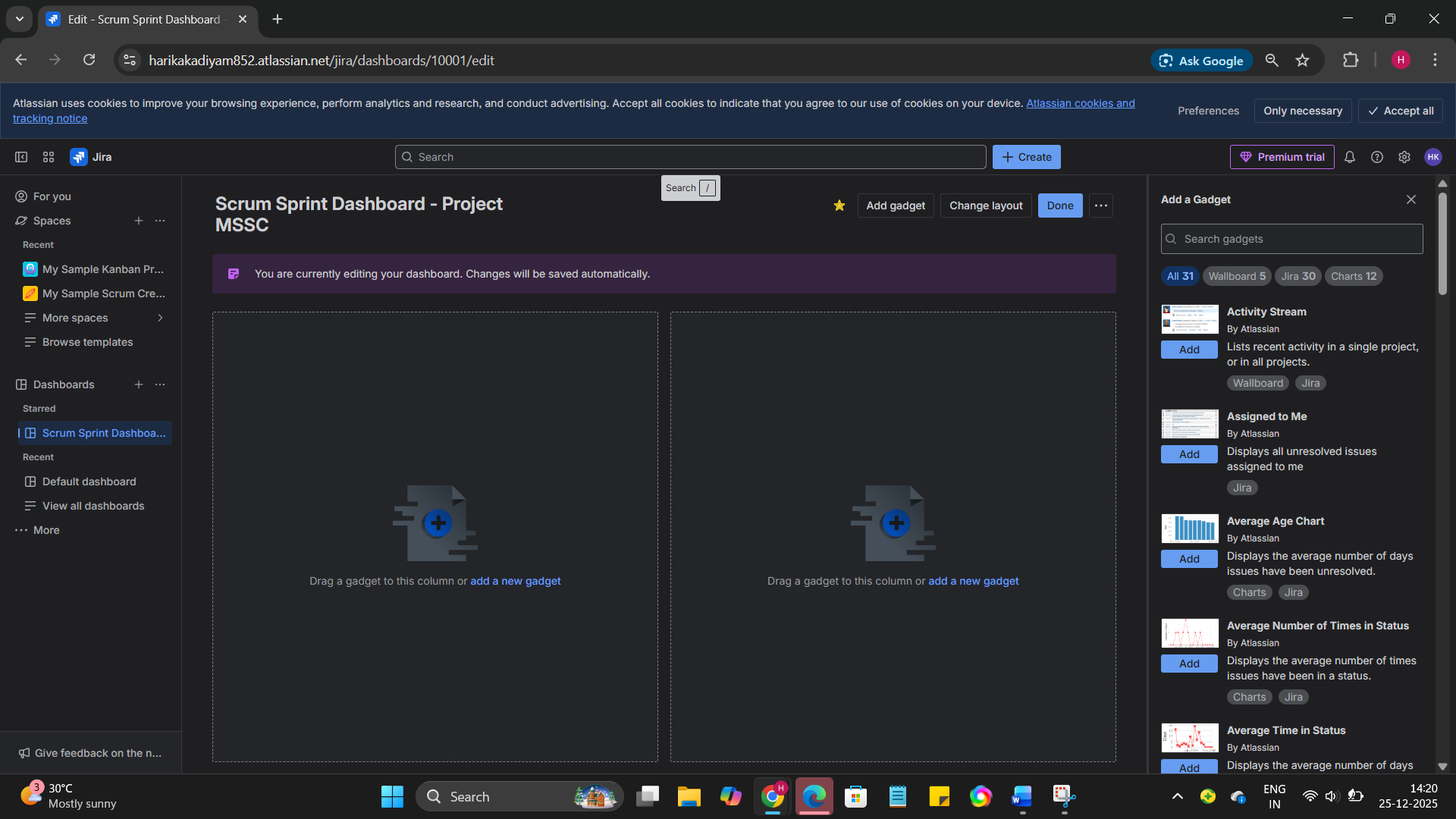
1**.** Created a **Scrum project**

Created a **Scrum project** from scratch using Jira Software.

2. Explored:

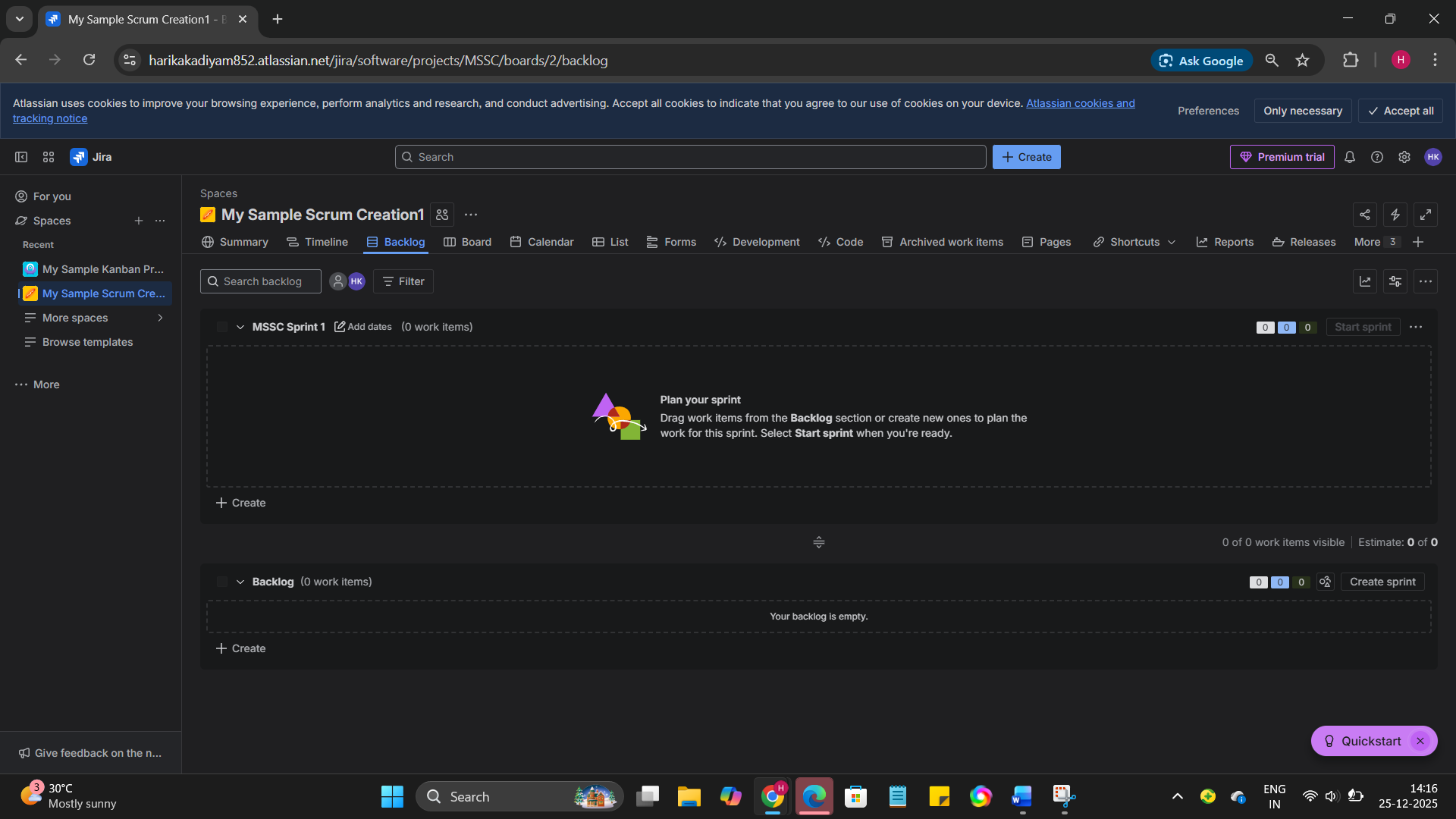
* Dashboard

Explored the Jira **Dashboard** to understand how project status and gadgets are displayed.



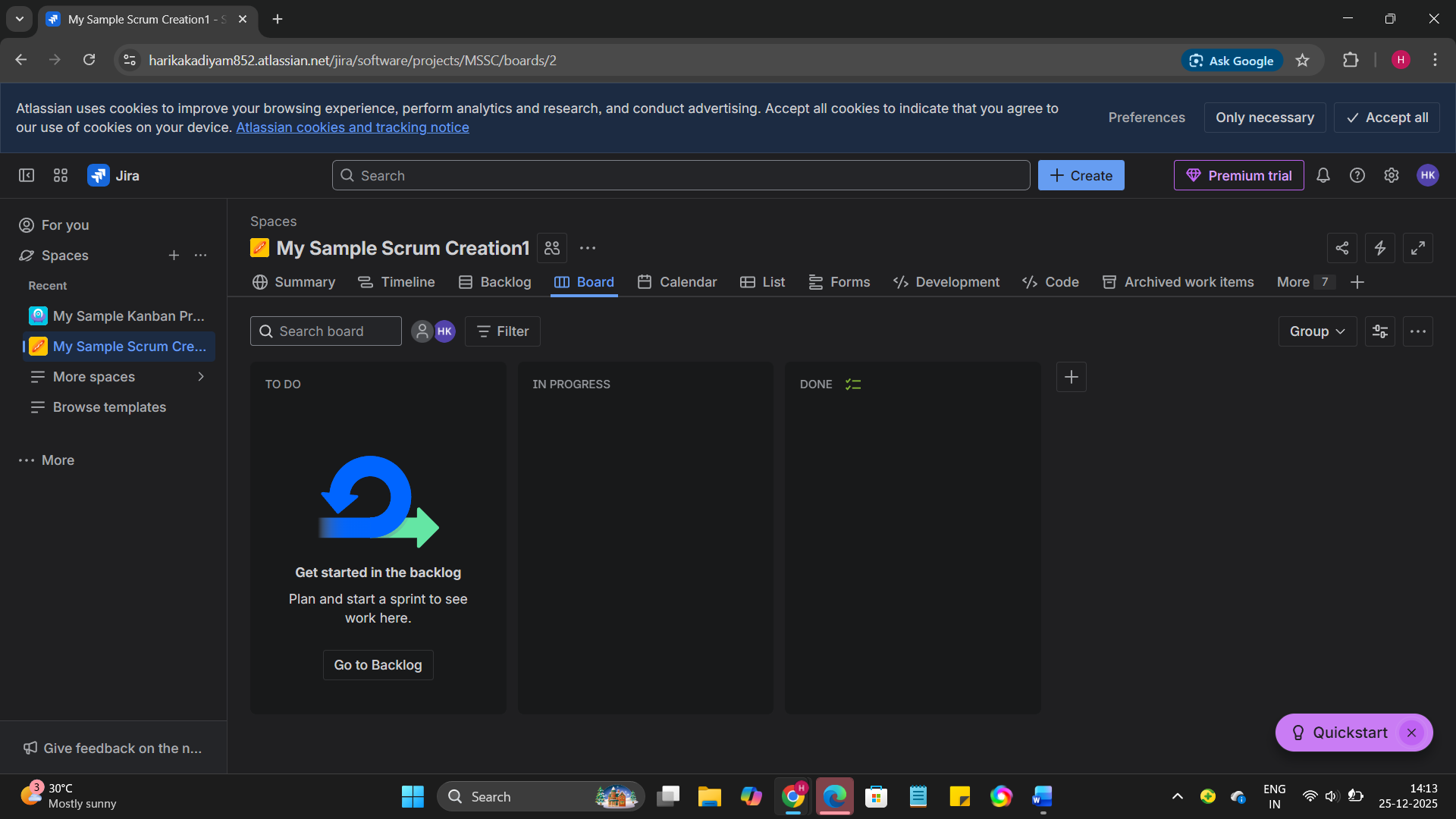
* Backlog

Navigated through the **Backlog** to view Epics, Stories, and Tasks.



* Board

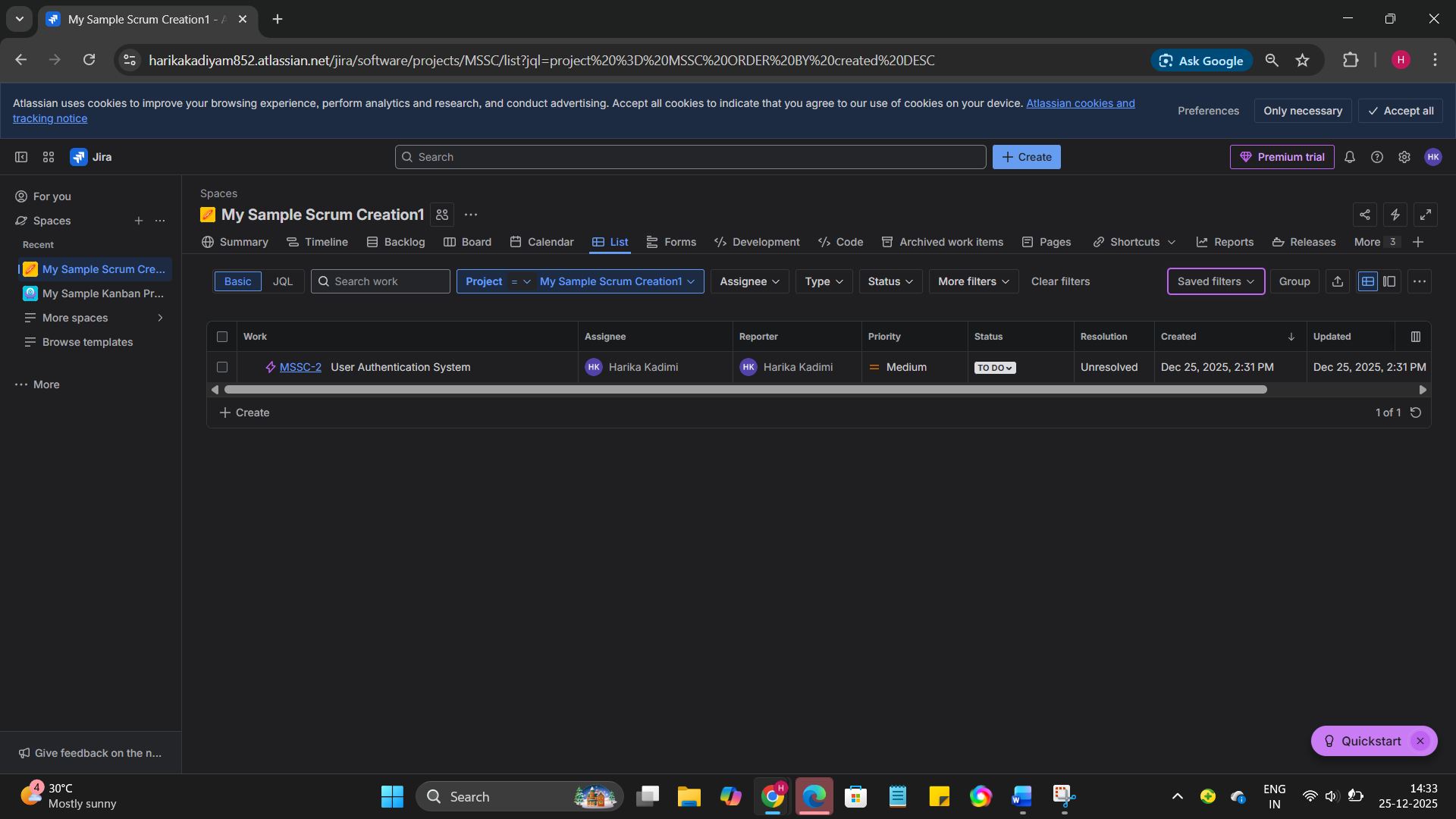
Used the **Scrum Board** to visualize work in different statuses.



3. Created:

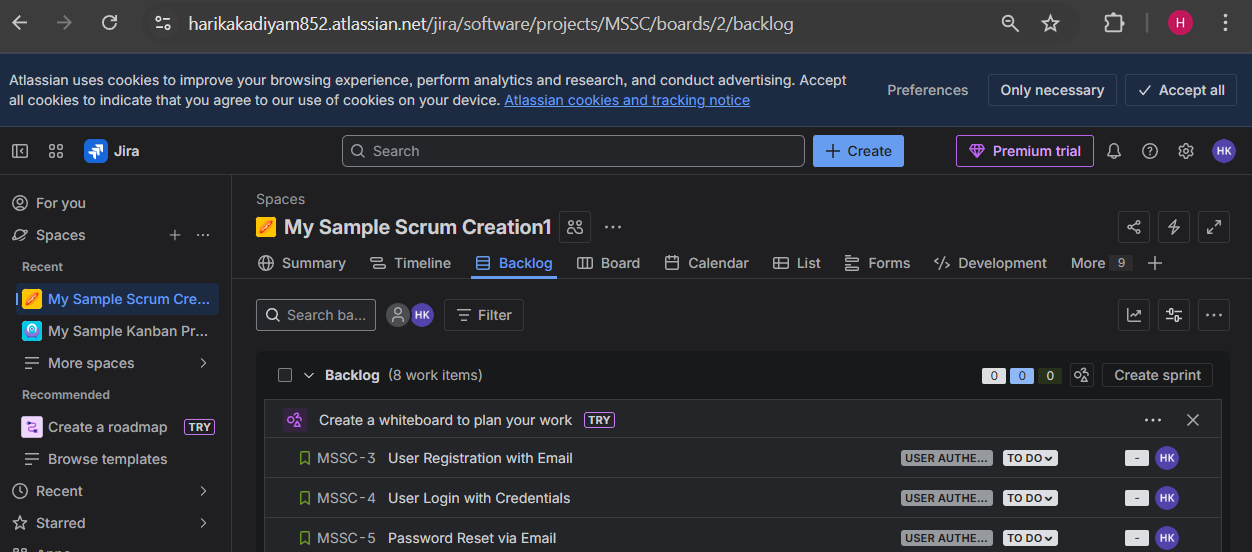
* 1 Epic

Created **1 Epic** named **User Authentication System** to represent a high‑level business requirement.



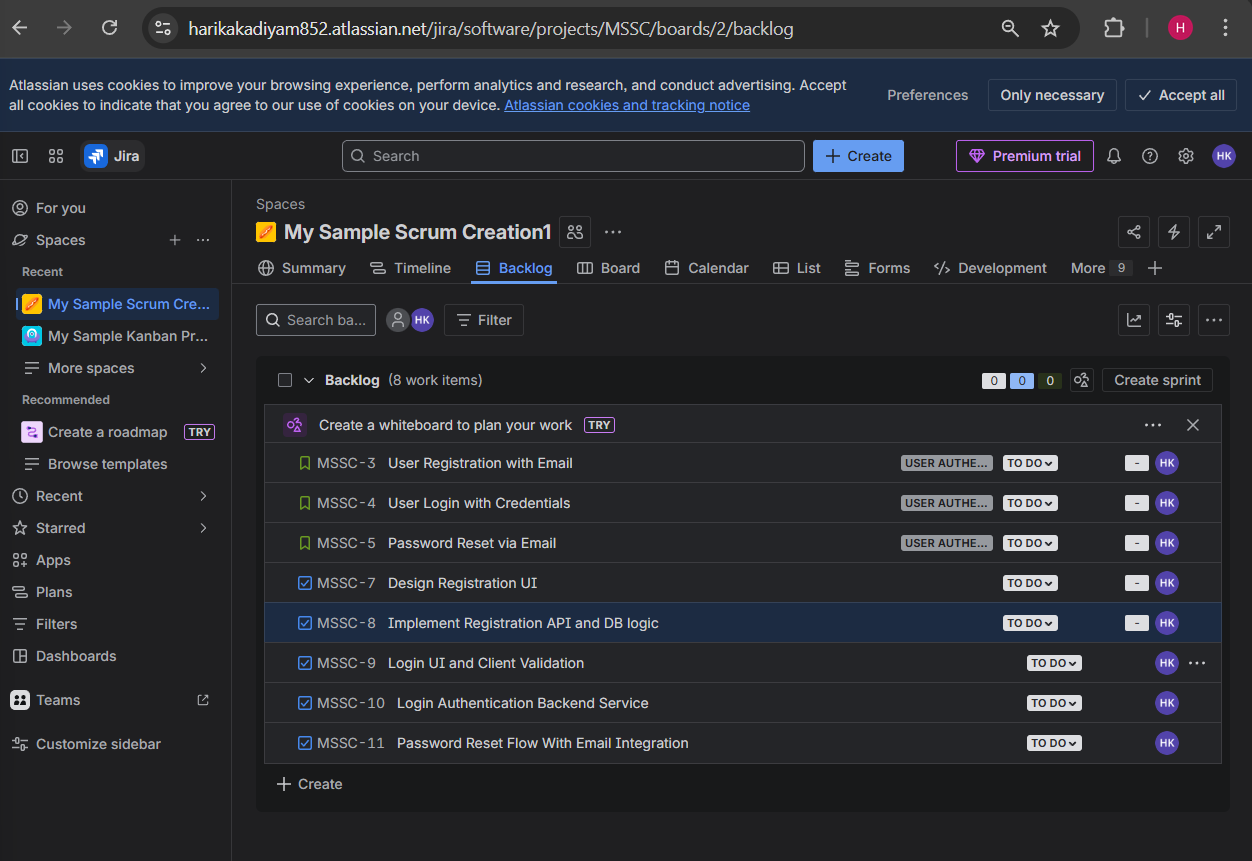
* 3 Stories

Created **3 User Stories** (as below shown) under the Epic.



* 5 Tasks

Created **5 Tasks** mapped to the stories to represent development and testing work.



**Day 2 – Issues, Workflow & Daily Usage**

**Learn**

* Issue types: Epic, Story, Task, Bug

Studied different **issue types** such as Epic, Story, Task, and Bug, and understood their real‑time usage in Agile projects.

* Status flow: To Do → In Progress → Done

Learned the default Jira workflow: **To Do → In Progress → Done**, and how teams track daily progress using this flow.

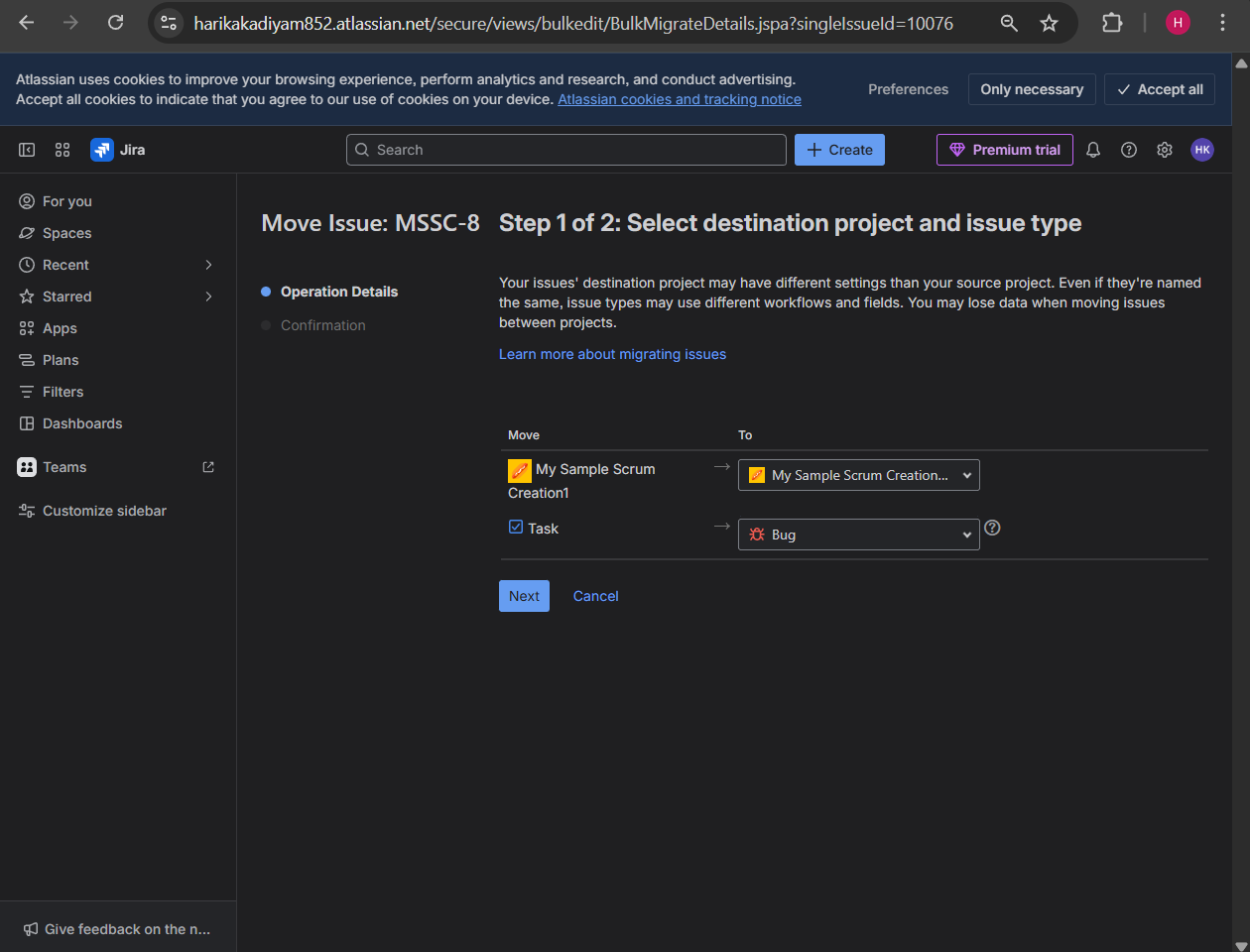
* Priority, labels, assignee, description

Understood the importance of **priority**, **labels**, **assignee**, and **clear descriptions** for effective collaboration.

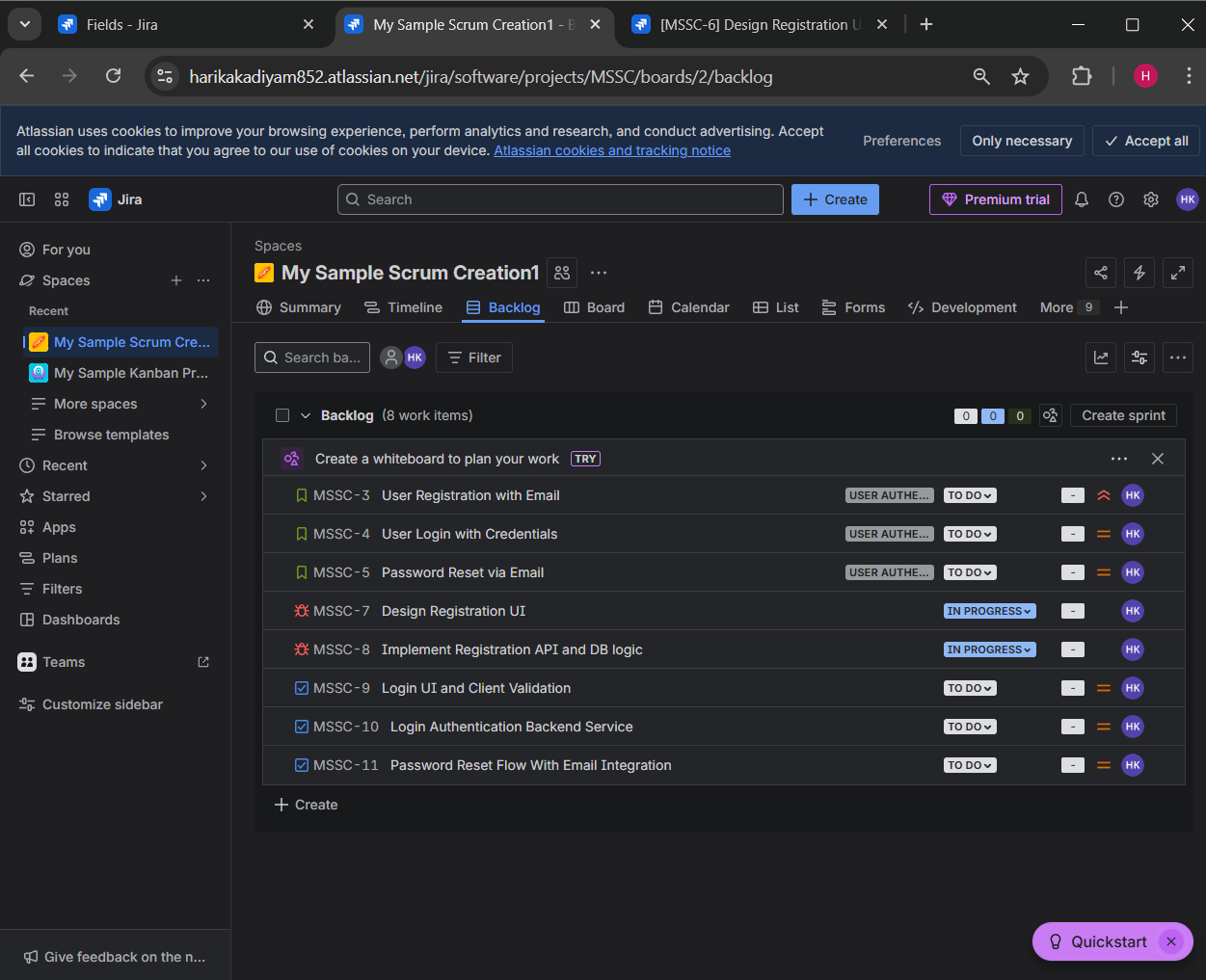
**Practical**

* Converted 2 tasks into **Bugs**

Converted **2 Tasks into Bugs** to simulate defect reporting.

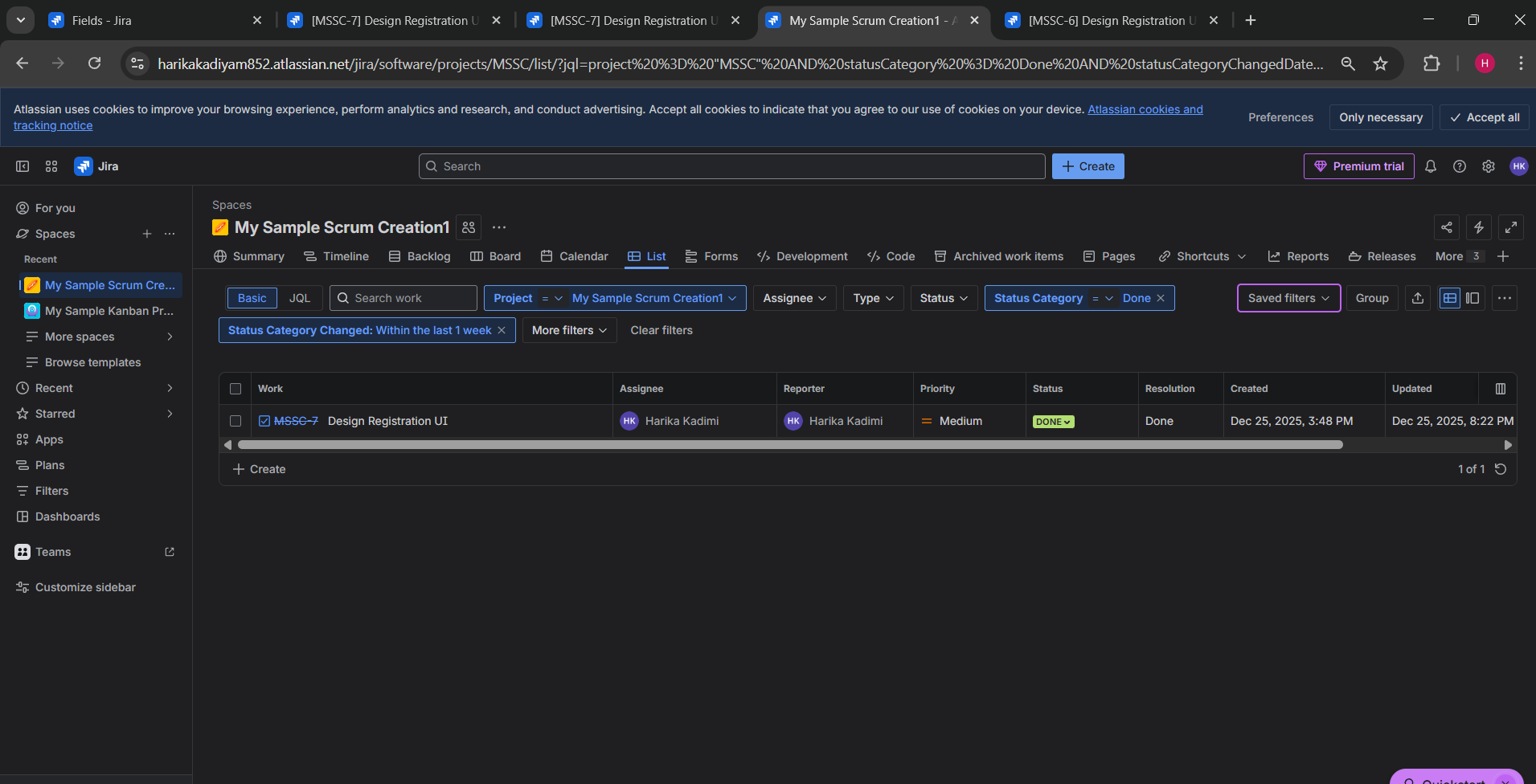


Below are the 2 tasks that are converted to bugs.



* Moved issues across workflow

Moved issues across workflow statuses based on work progress.

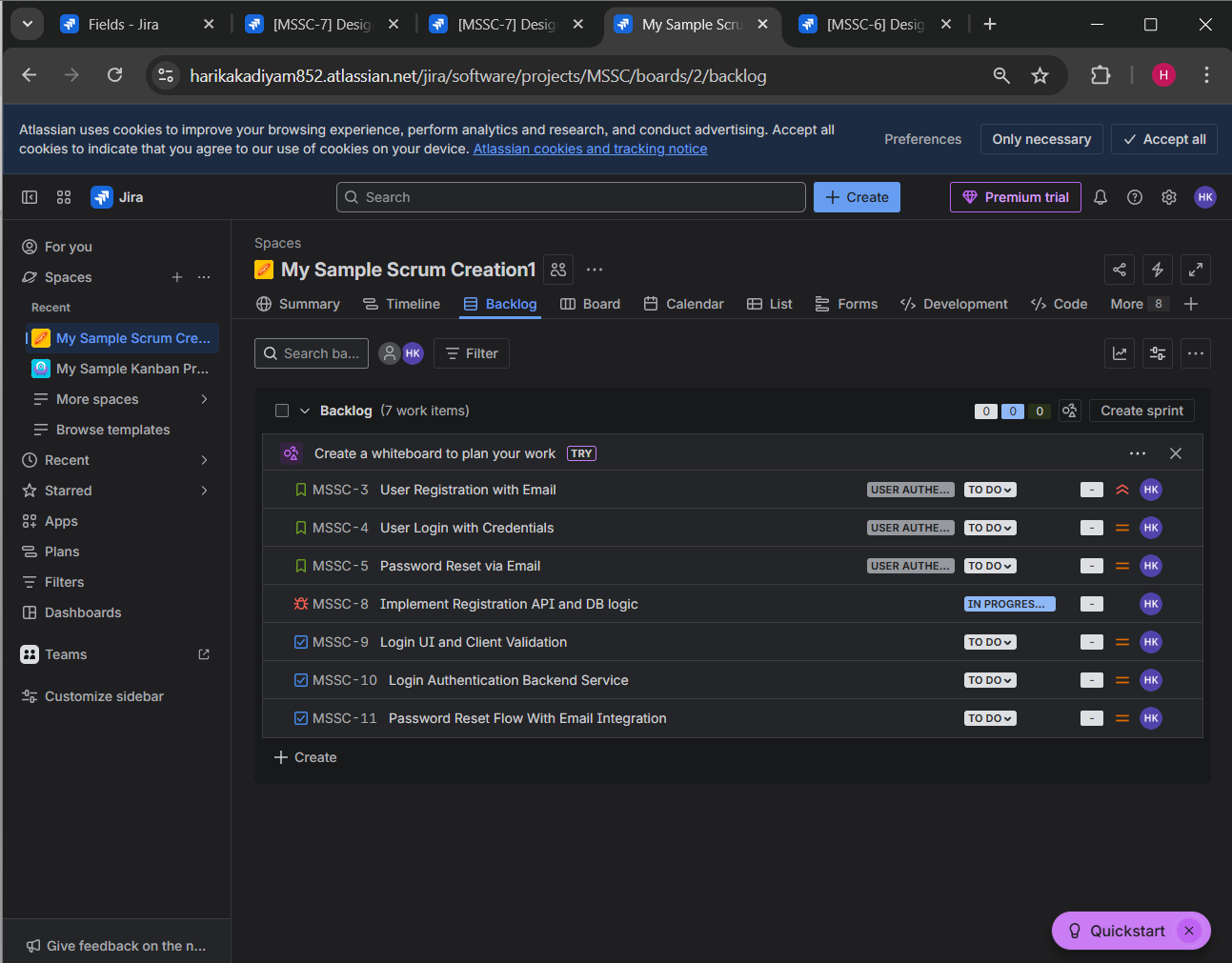


* Linked issues (blocks, relates to)

Linked issues using relationships such as **blocks** and **relates to**.

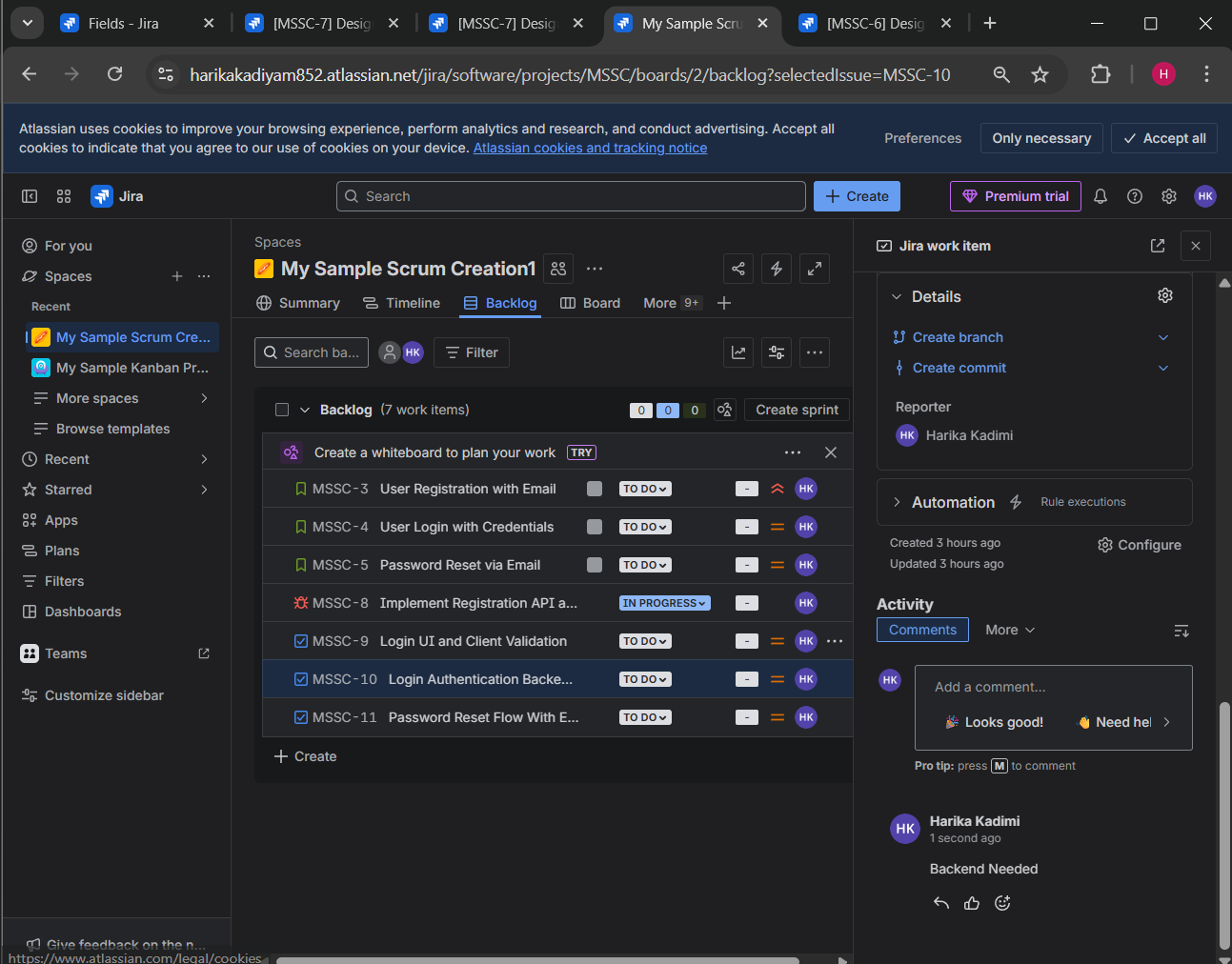
Example: Login UI and client validation is blocked by Login Authentication backend service.

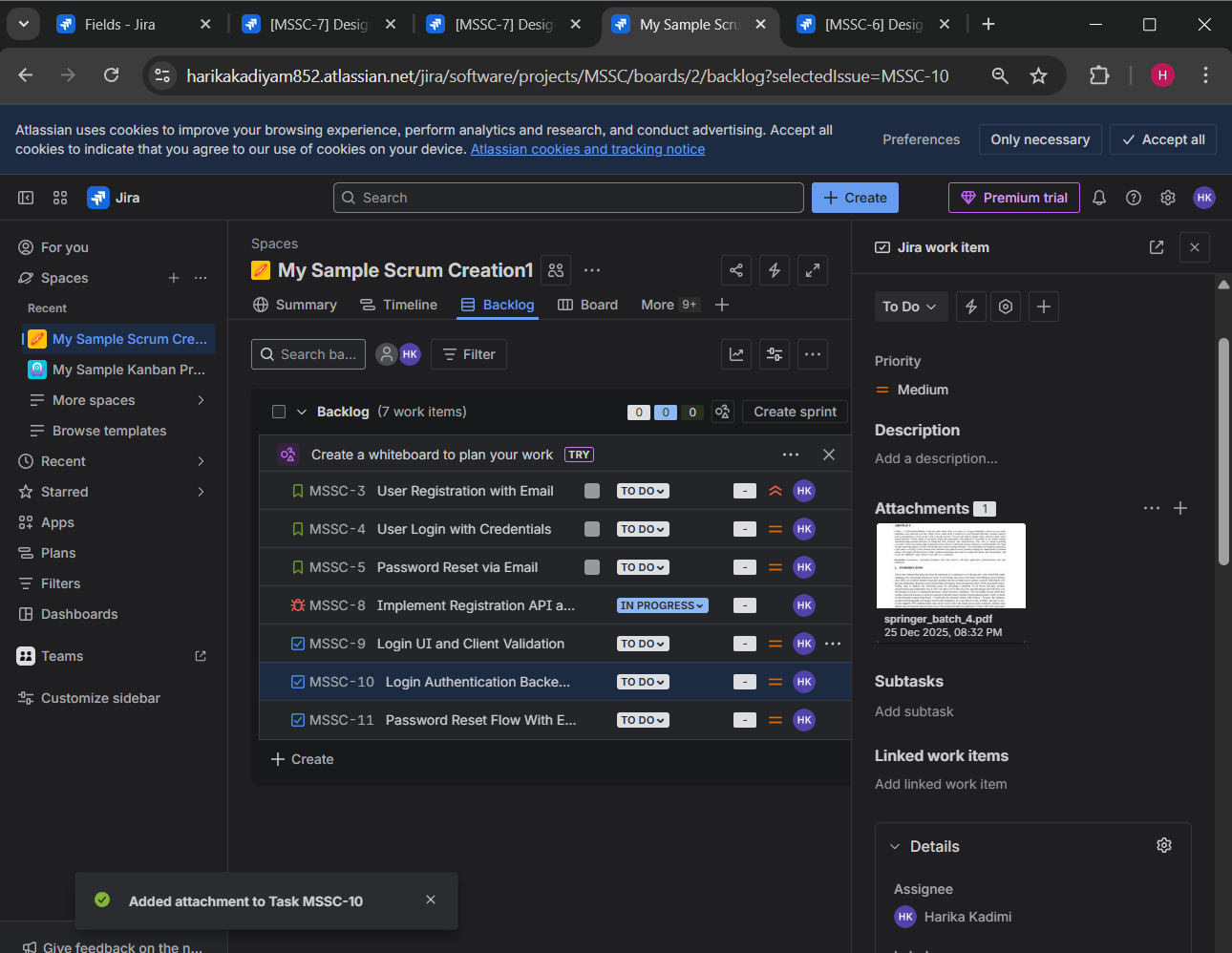
Example: Password Reset via Email is relates to Password Reset Flow with Email Integration.



* Added comments and attachments

Added **comments** for communication and **attachments** (screenshots/logs) for better clarity as below.





**Day 3 – Scrum Board & Sprint Execution**

**Learn**

* Product backlog vs Sprint backlog

Understood the difference between **Product Backlog** (all future work) and **Sprint Backlog** (work committed for the current sprint).

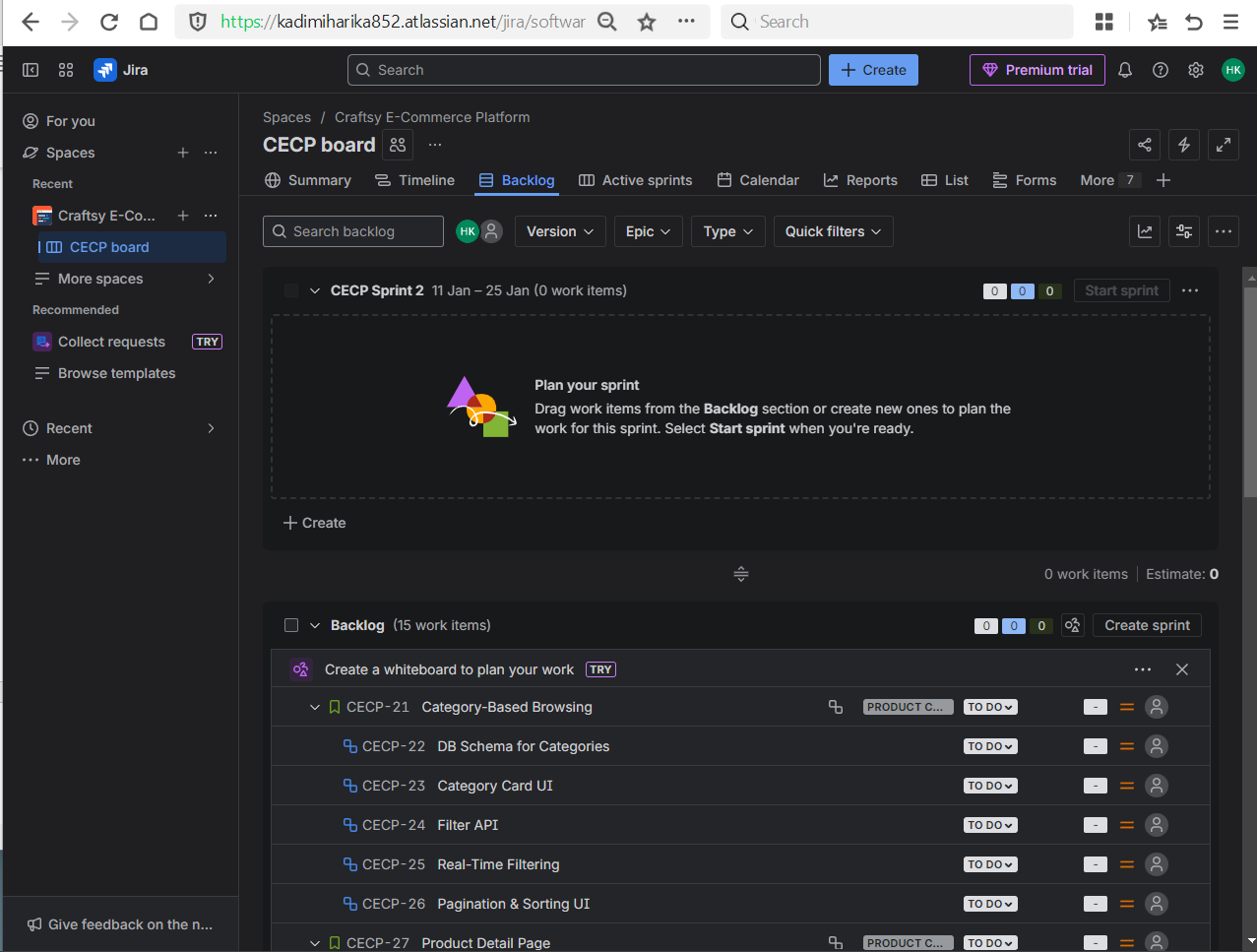
* Sprint planning & execution

Learned how **Sprint Planning**, **Sprint Execution**, and **Sprint Completion** work in real Agile teams.

**Practical**

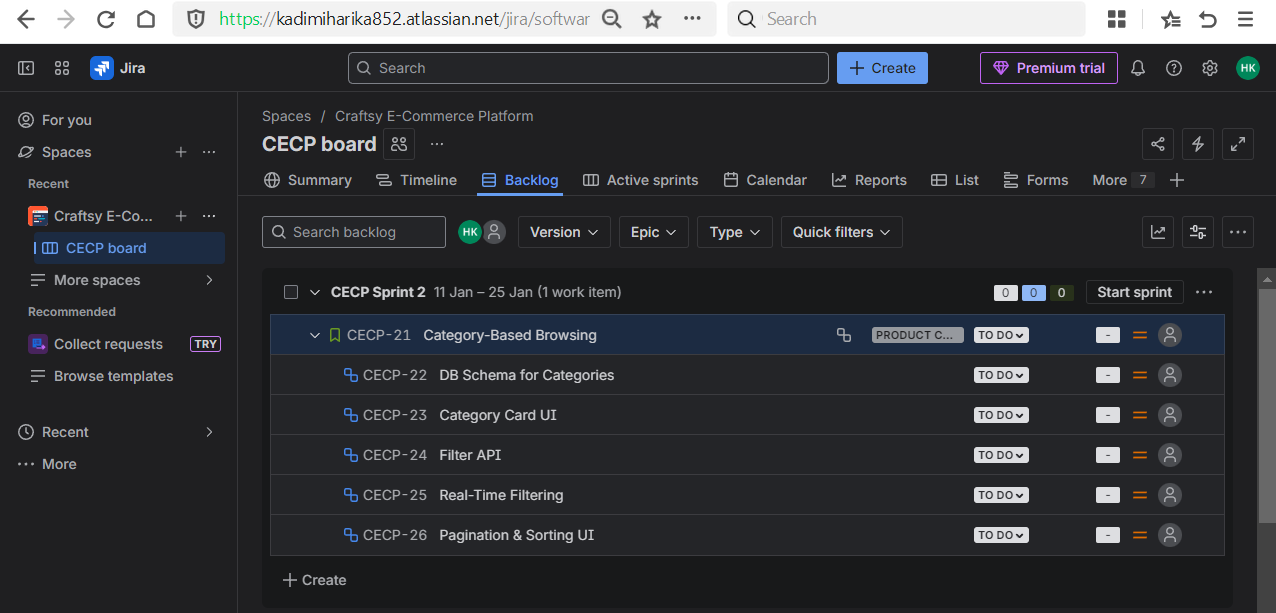
* Created a **Sprint**

Created a sprint from the backlog.



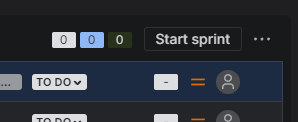
* Added stories to sprint

Added stories and tasks to the sprint based on medium priority.



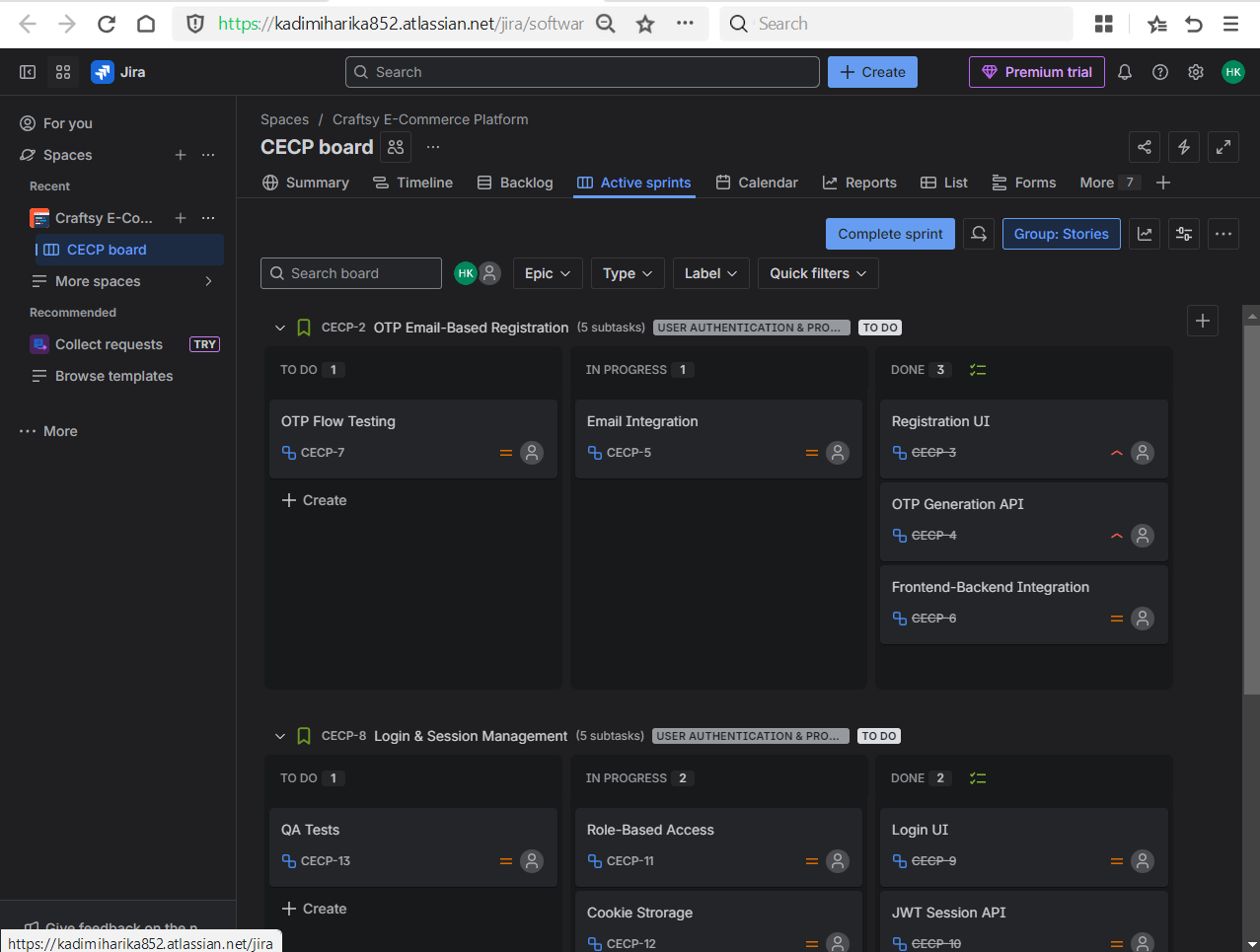
* Started sprint

Started the sprint with a defined duration (1 week).



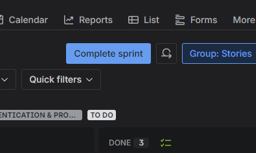
* Moved tasks daily

Moved tasks daily across statuses to simulate daily stand‑up updates.



* Completed sprint

Completed the sprint after all work was marked as Done.



**Day 4 – Kanban + Filters (JQL Basics)**

**Learn**

* Kanban board purpose

Kanban is used for continuous workflow where tasks are not planned in fixed sprints. It is mainly used for support projects, maintenance work, and production issues where work comes continuously.

* Continuous delivery concept

In Kanban, issues move continuously from *To Do* → *In Progress* → *Done* without waiting for sprint completion. This helps teams deliver faster and respond quickly to changes.

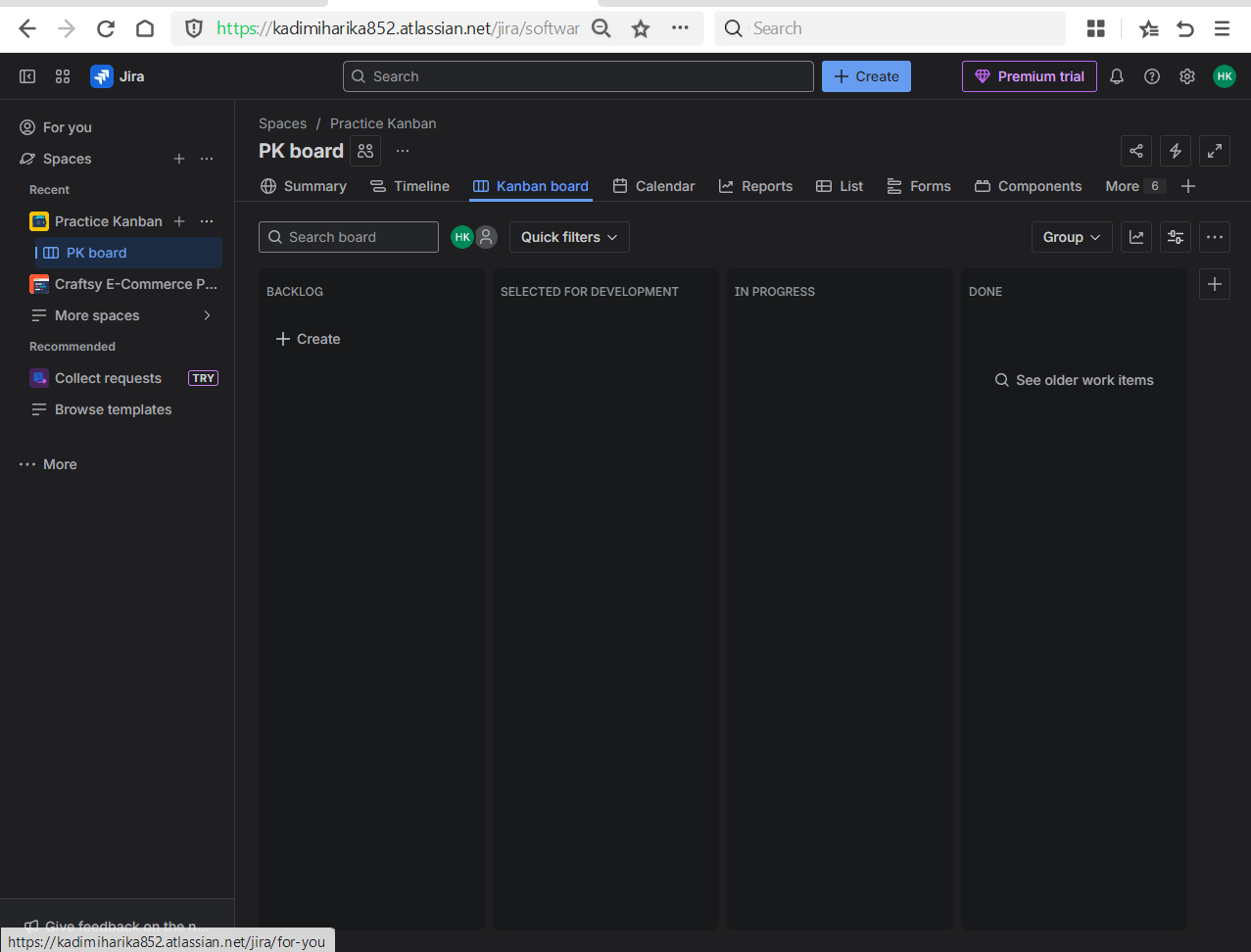
* JQL fundamentals

JQL is used to search issues in Jira based on conditions like assignee, status, priority, project, labels, due date, and text fields.

**Practical**

* Created a **Kanban project**

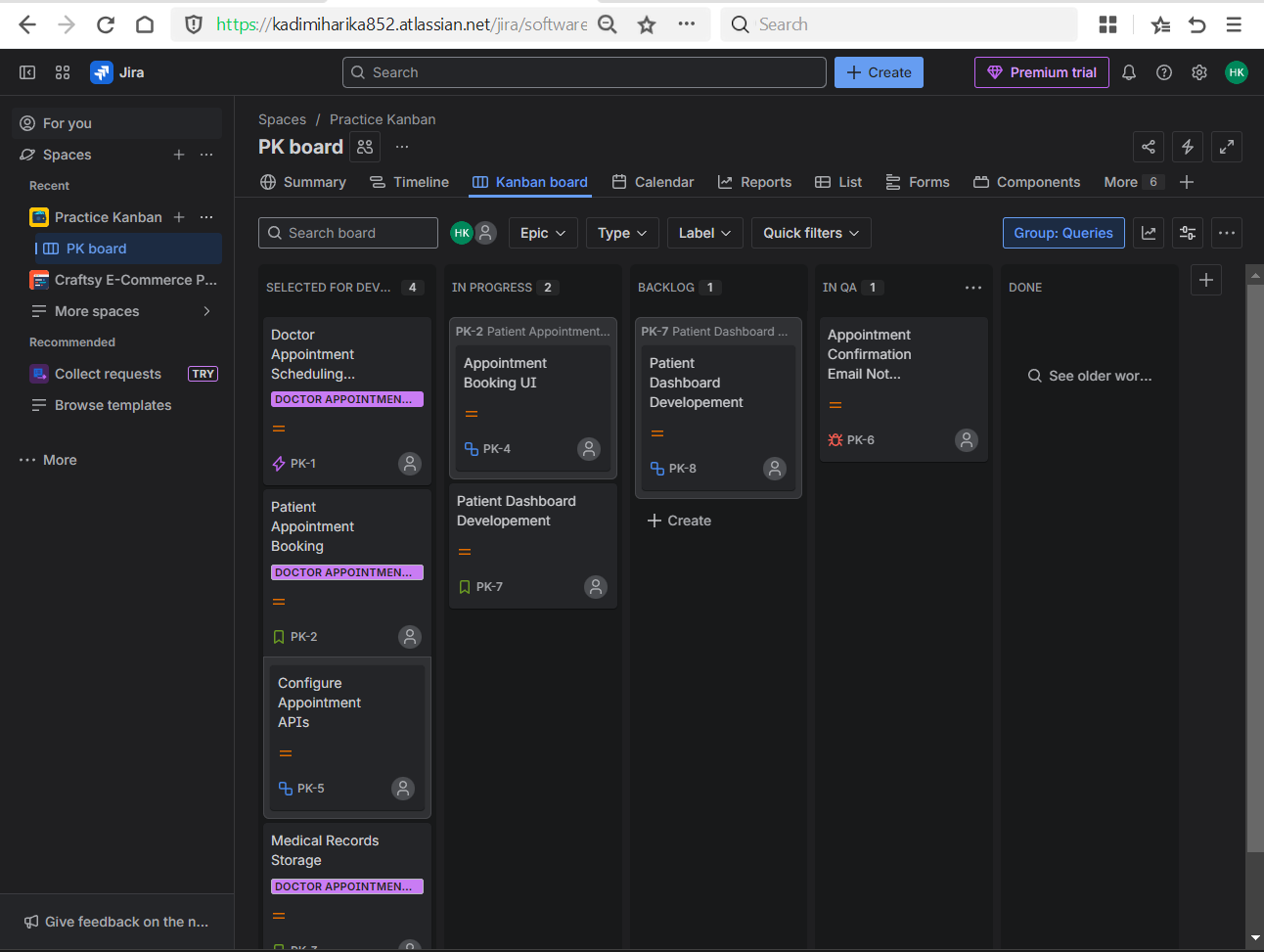
Created a **Kanban project** with a name **Practice Kanban (PK)** in Jira Software.



* Created issues and move them

Created multiple issues such as Stories, Tasks and Bugs.

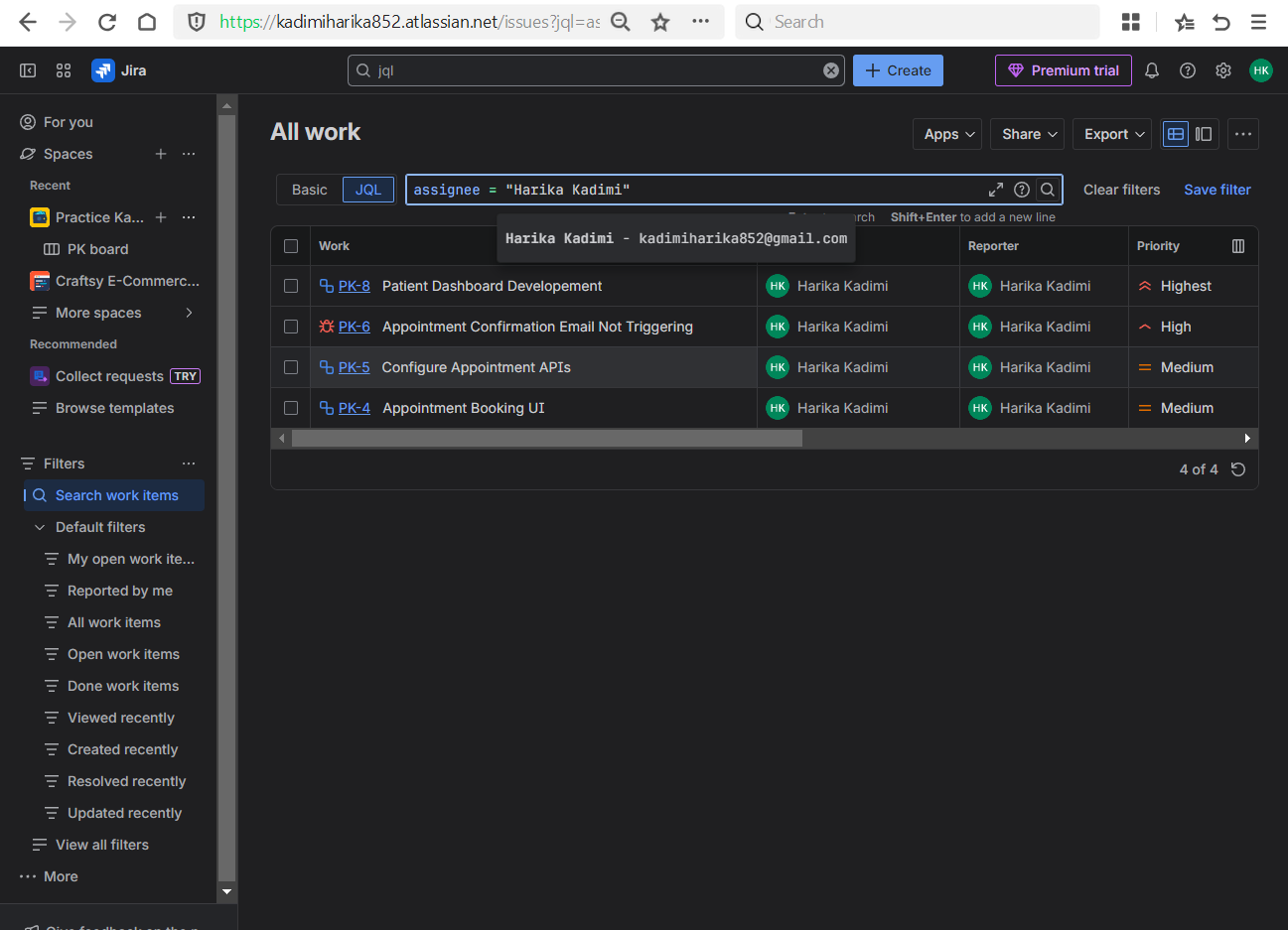
Moved issues across kanban columns to simulate real-time project flow.



* Used JQL:
  + 1. **Issues assigned to me**

Used to check my own workload.

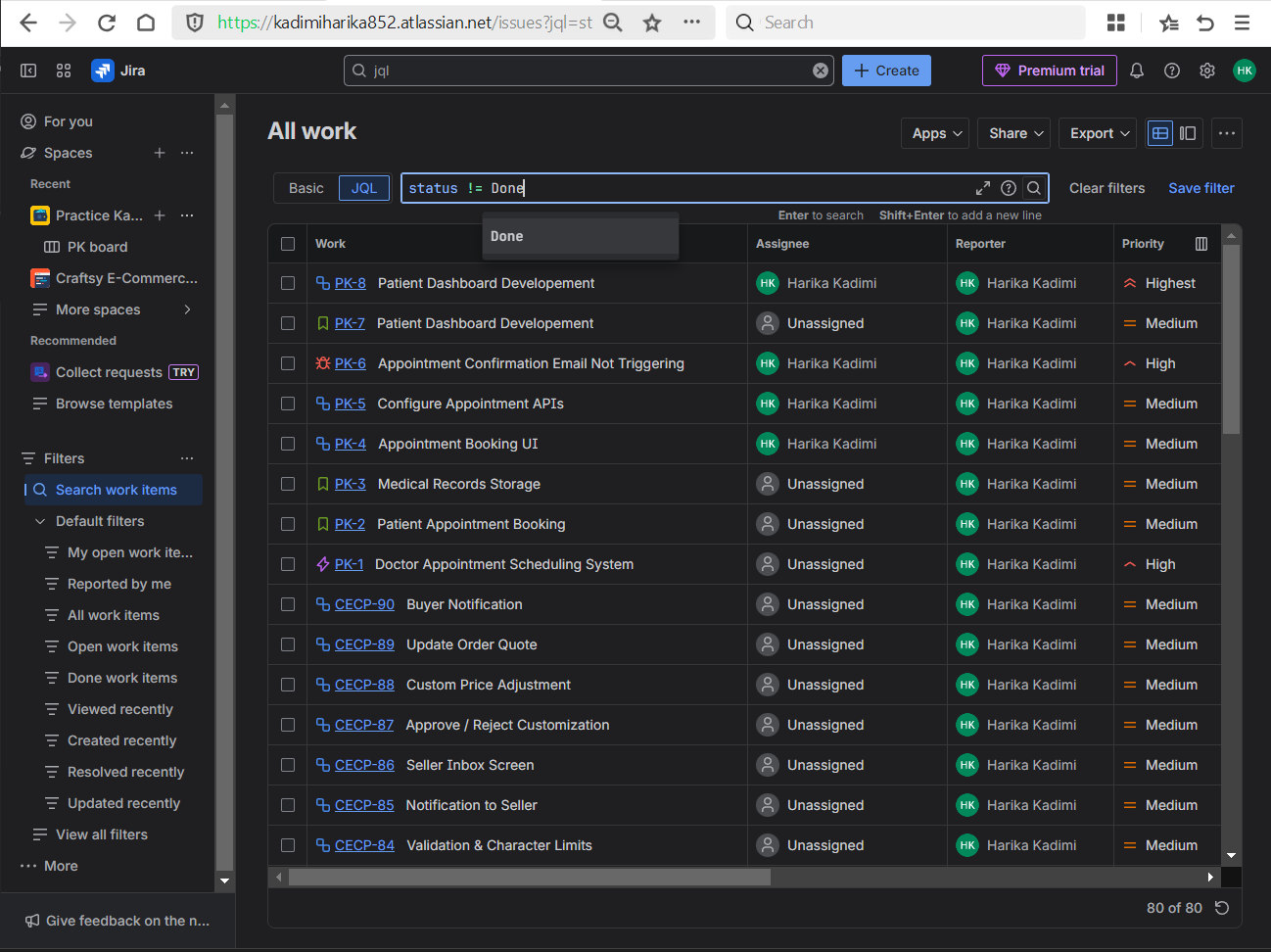
*assignee = currentUser()*



* + 1. **Issues not completed**

Used to track pending work.

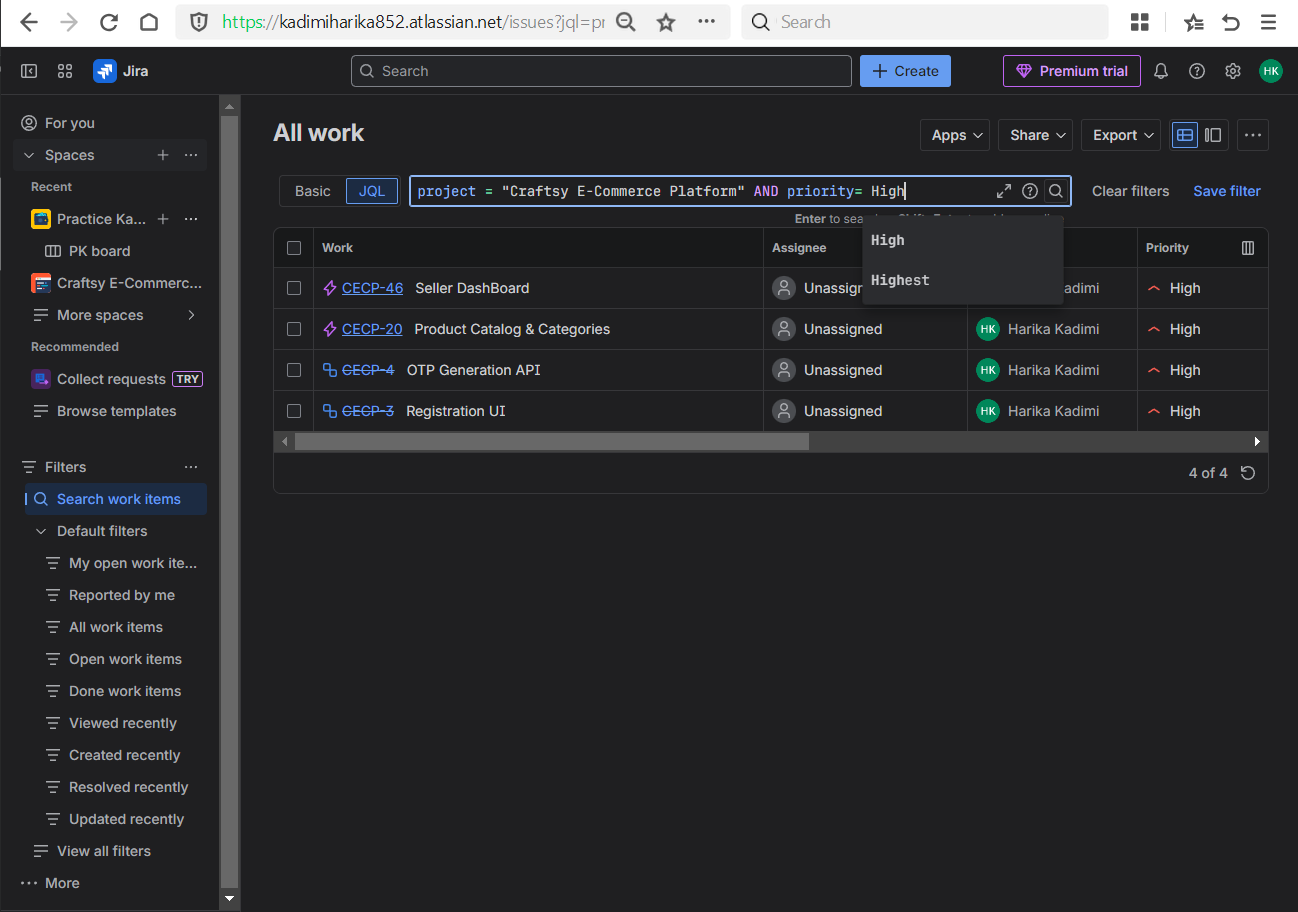
*status != Done*



* + 1. **High priority issues in a specific project**

Used by leads to monitor crtitical tasks.

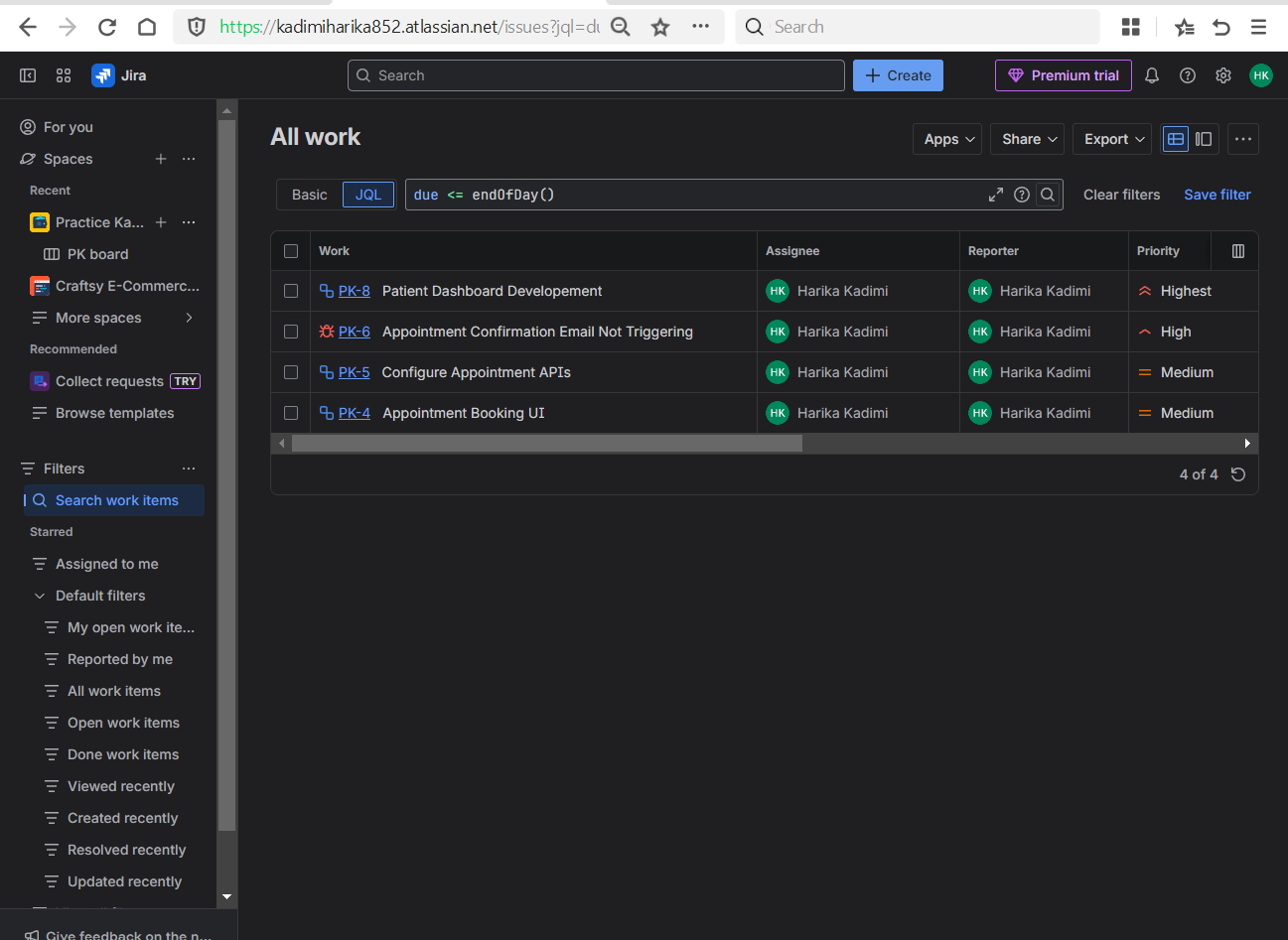
*project = “ProjectName” AND priority = High*



* + 1. **Taks due this day**

Used to track upcoming deadlines

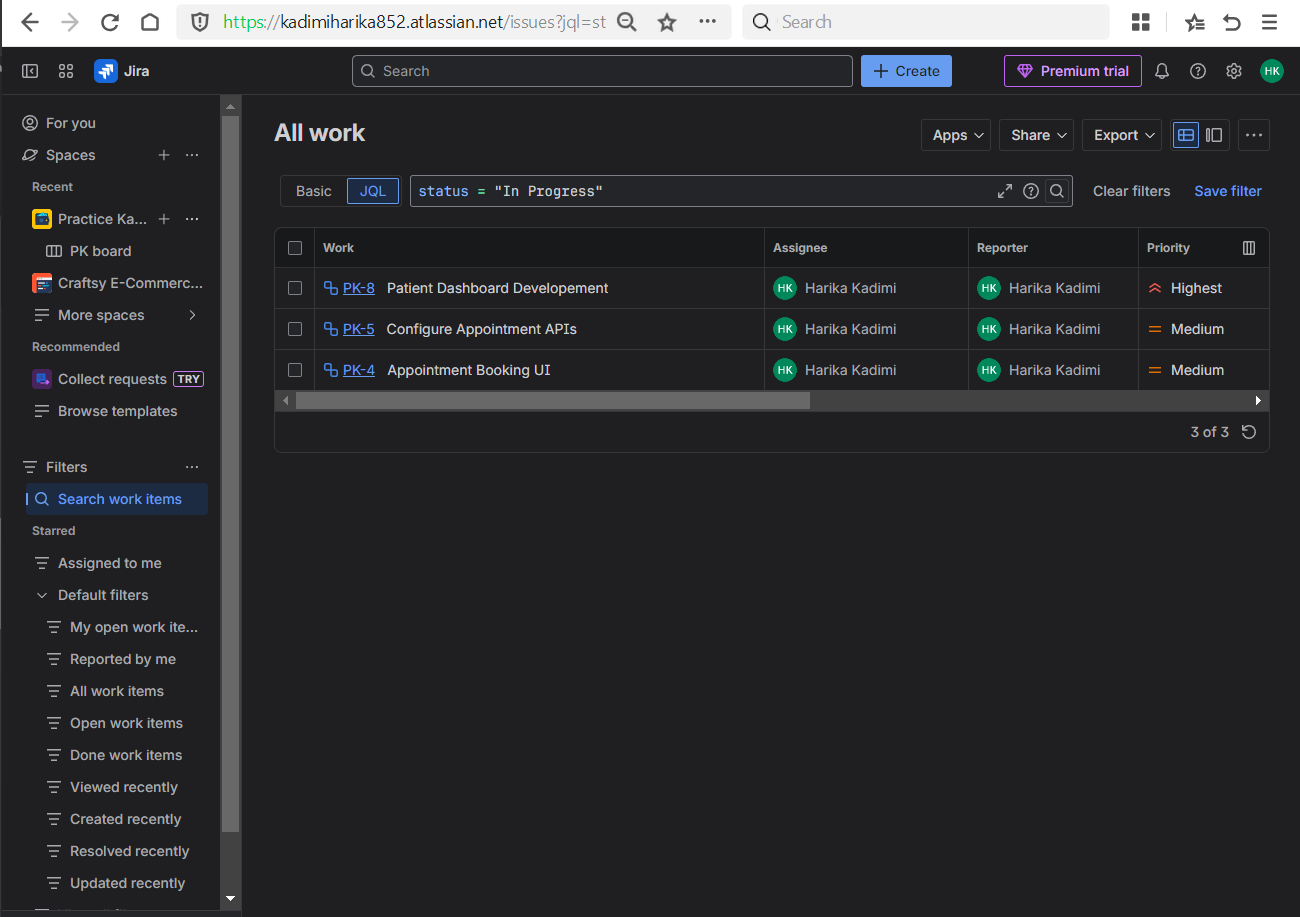
*due <= endOfDay()*



* + 1. **All progressing tasks**

Used to view work that has started and are currently employees working on

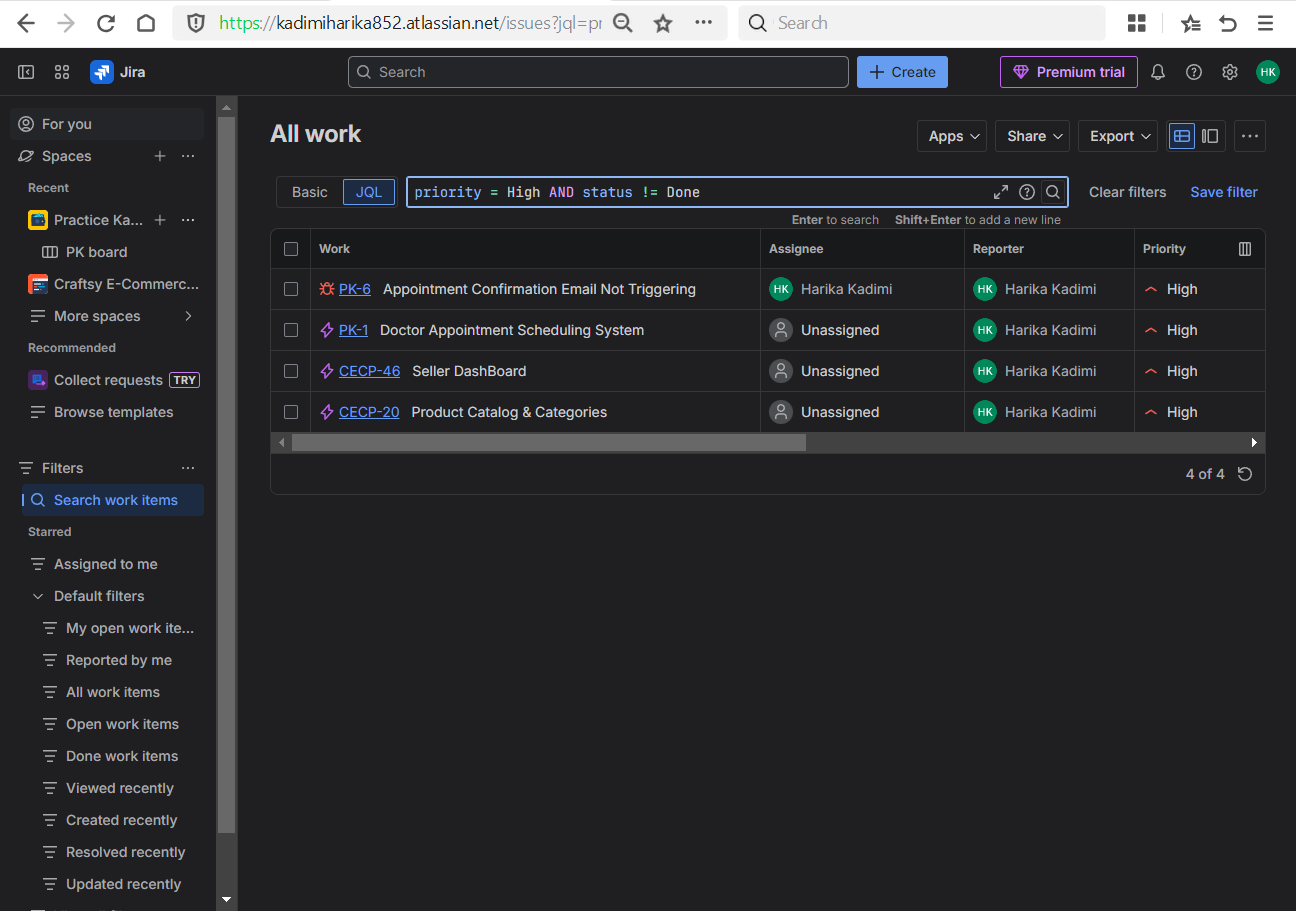
*status = “In Progress”*



* **Combined Conditions**
  + 1. **High priority and not done**

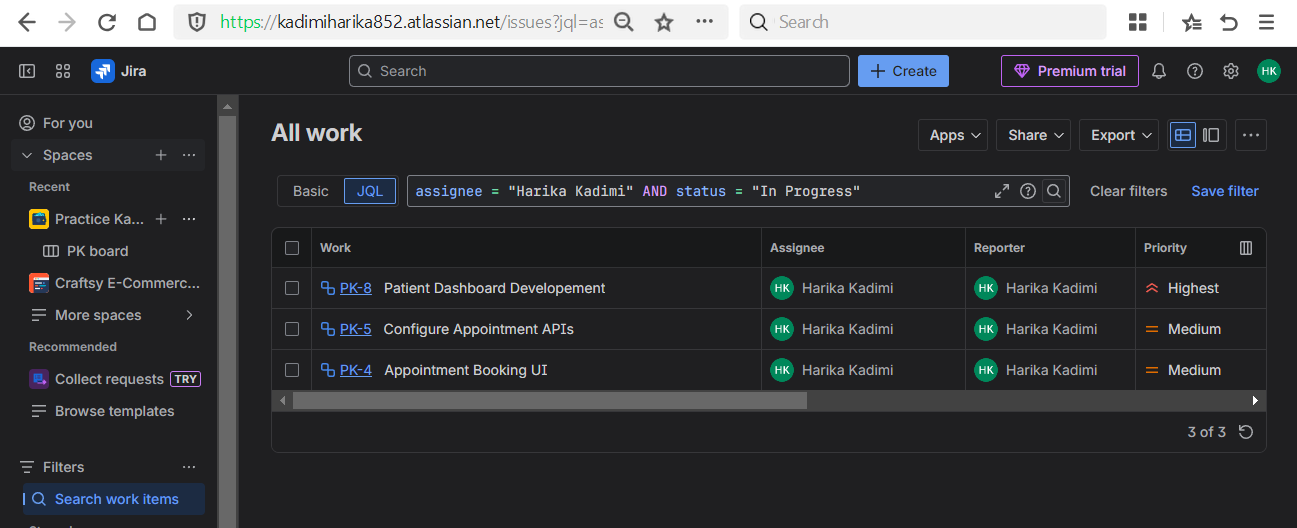
Used to identify urgent unfinished work

*priority = High AND status != Done*



* + 1. **Assigned to Harika and In Progress**

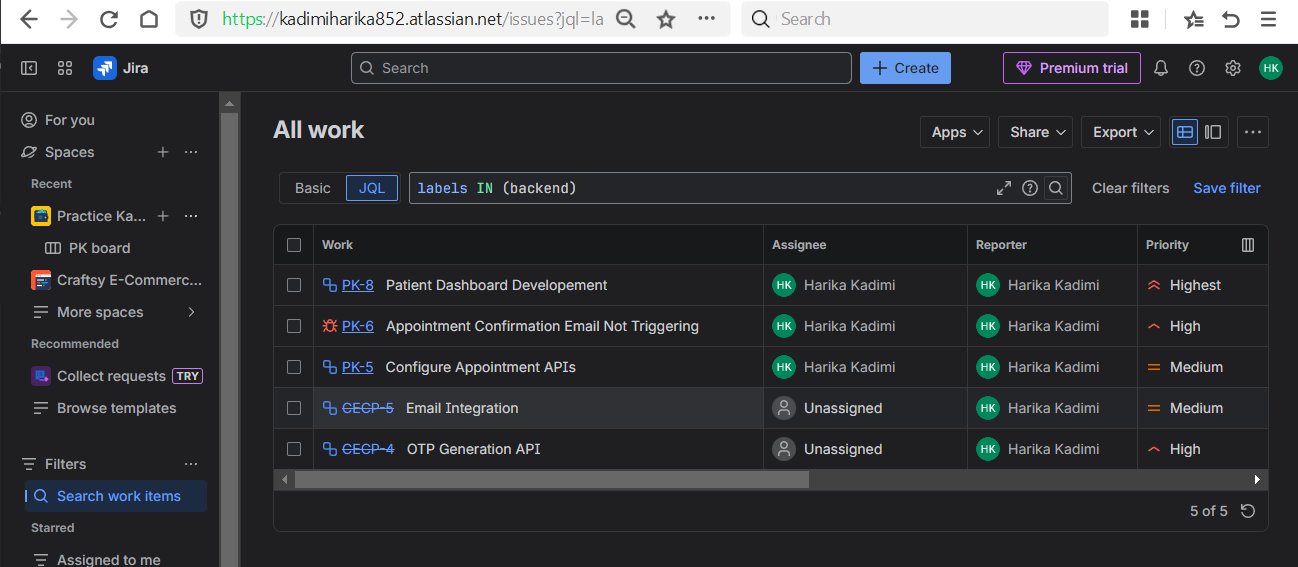
Used to track individual progress.  
*assignee = “Harika Kadimi” AND status = “In Progress”*



* + 1. **Issues with backend label**

Used to filter bakend-related issues

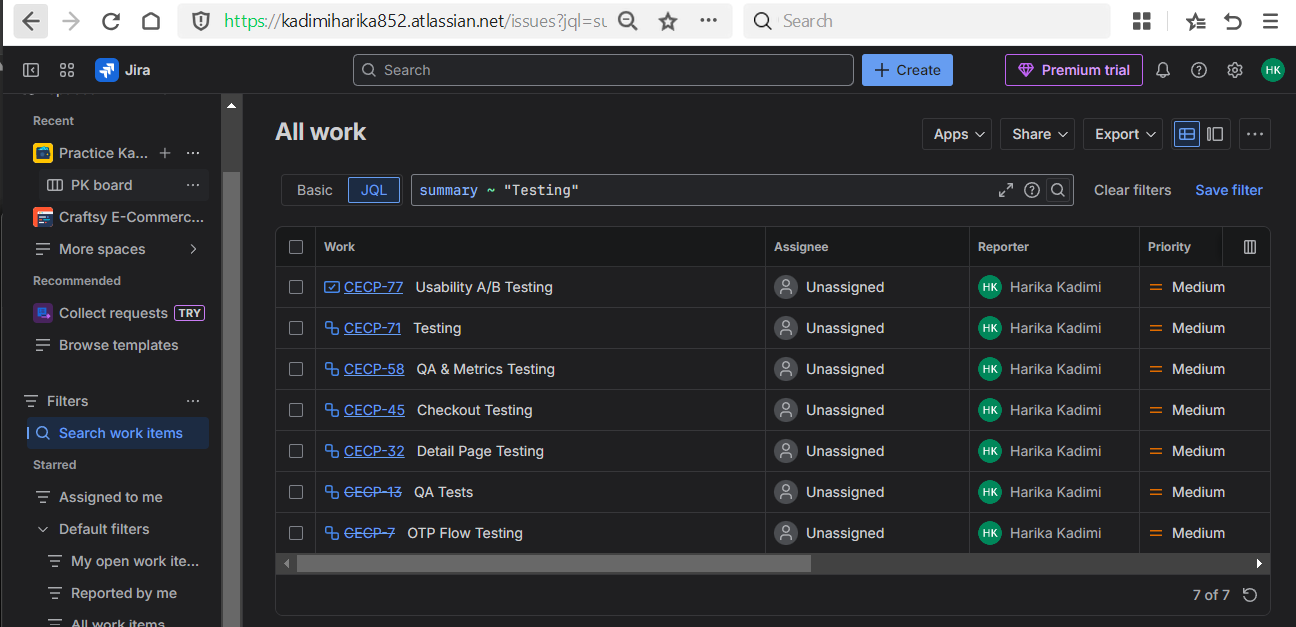
*Labels IN (backend)*



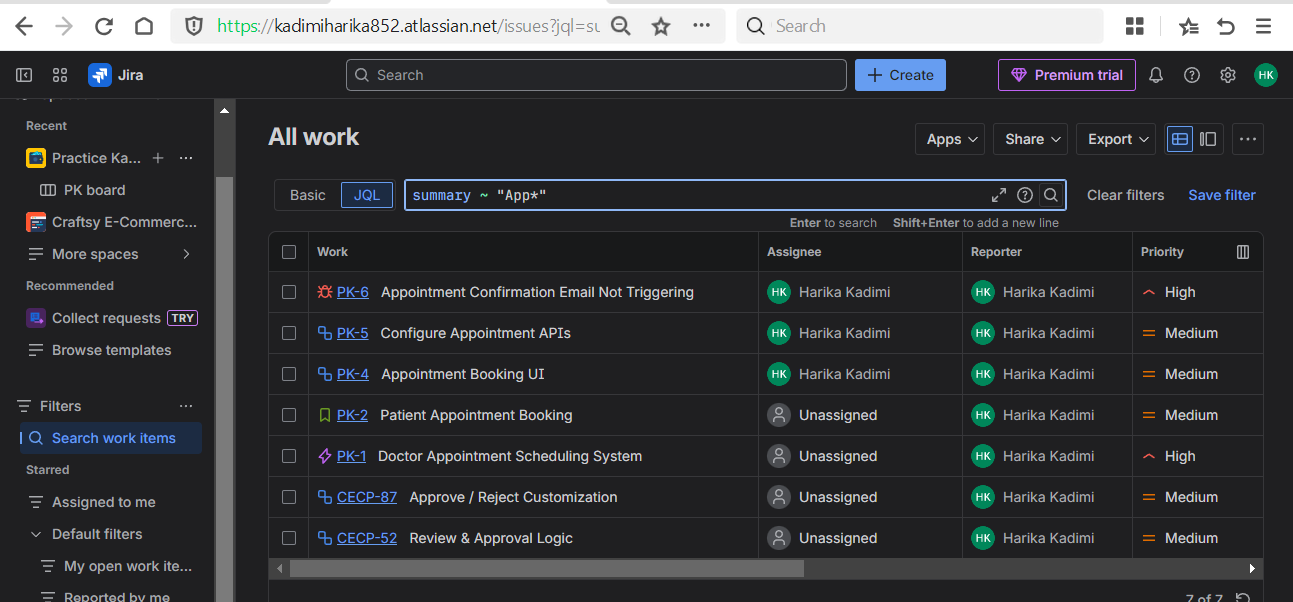
* + 1. **Summary containing keyword**

Used for quick issues search based on the summary containing specified keyword

*Summary ~ “keyword” // testing*



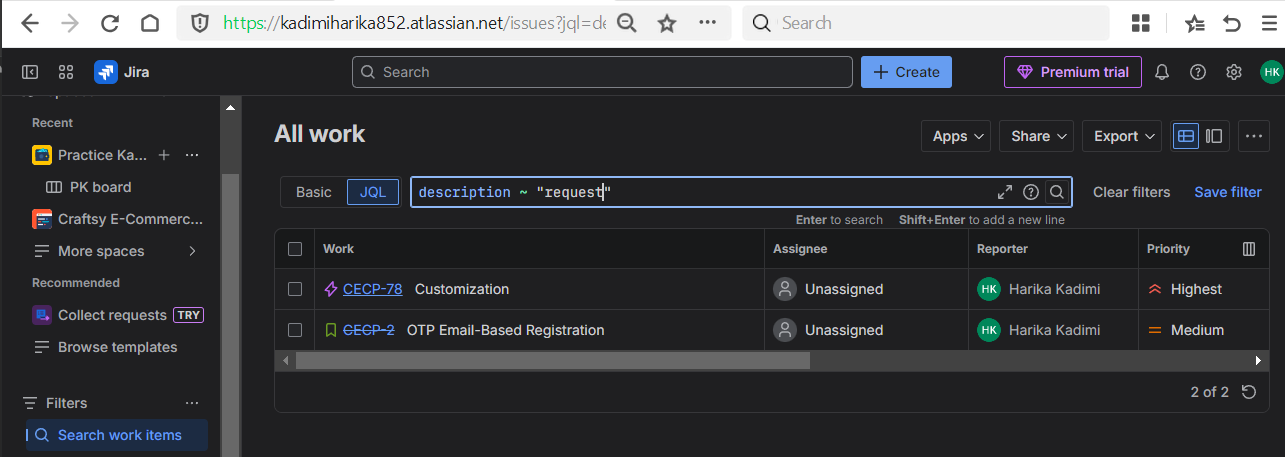
*Summary ~ “keyword” //App*



* + 1. **Description containing keyword**

Used to search the issues that are containing the description with a specified keyword.

*description ~ “keyword” // request*



* + 1. **All due before a data**

Used to find the issues that are in mentioned date

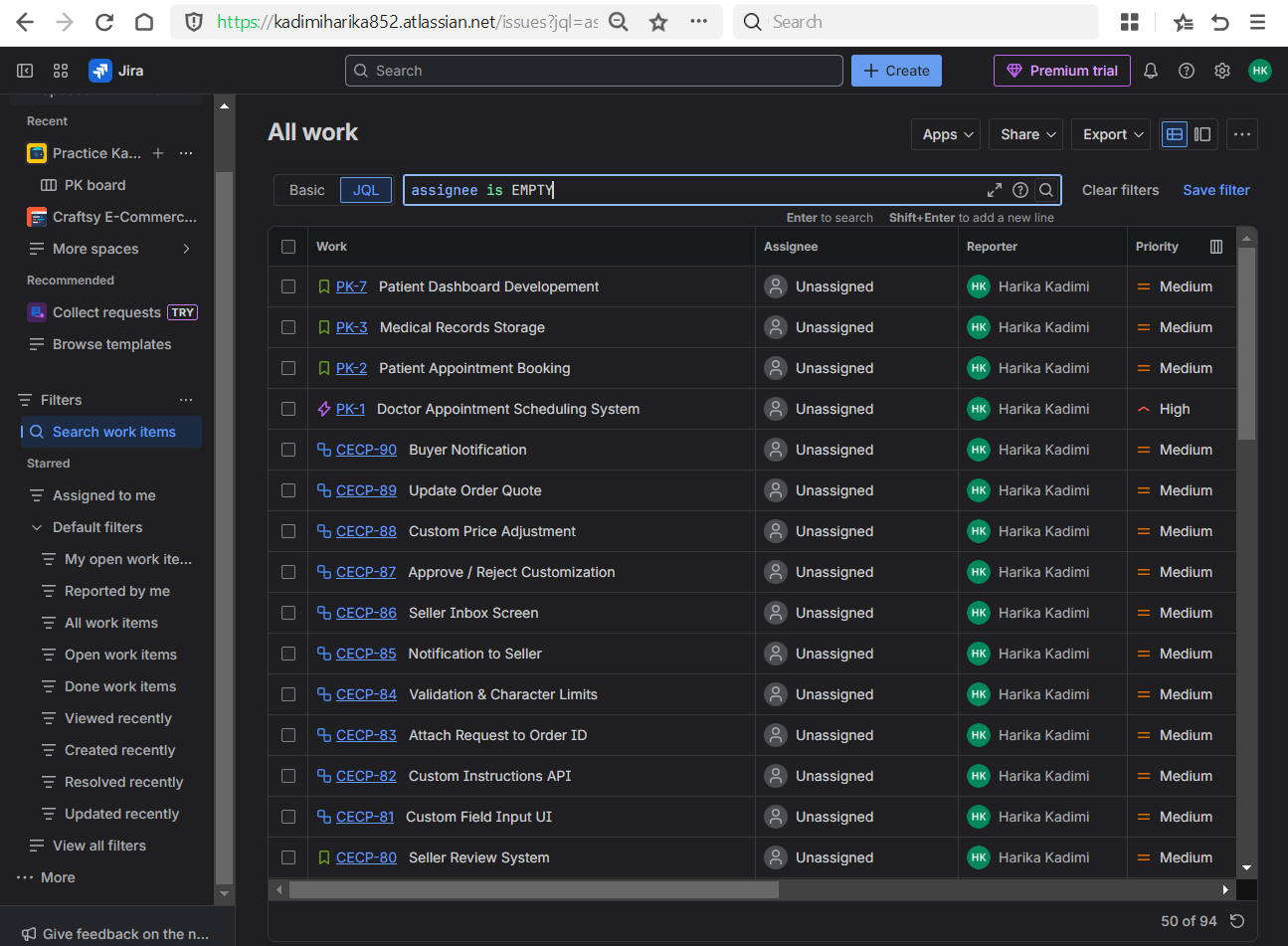
*due <= “Date” // “2025-01-03”*



* + 1. **Issues with no assignee**

Used to find unassigned work.

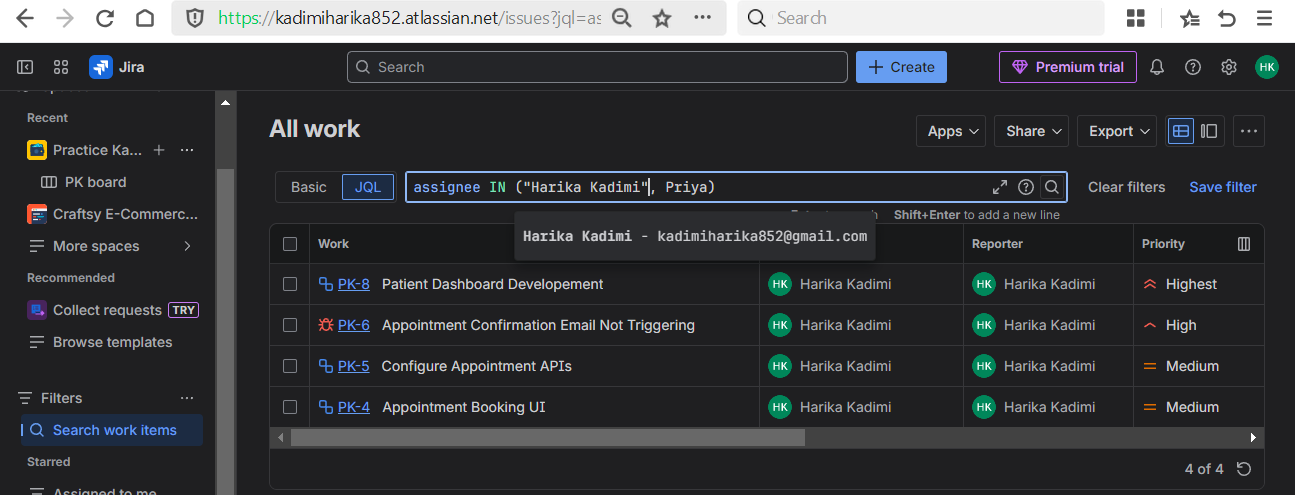
*Assignee IS EMPTY*



* + 1. Issues assigned to multiple users

Used for team-level tracking.

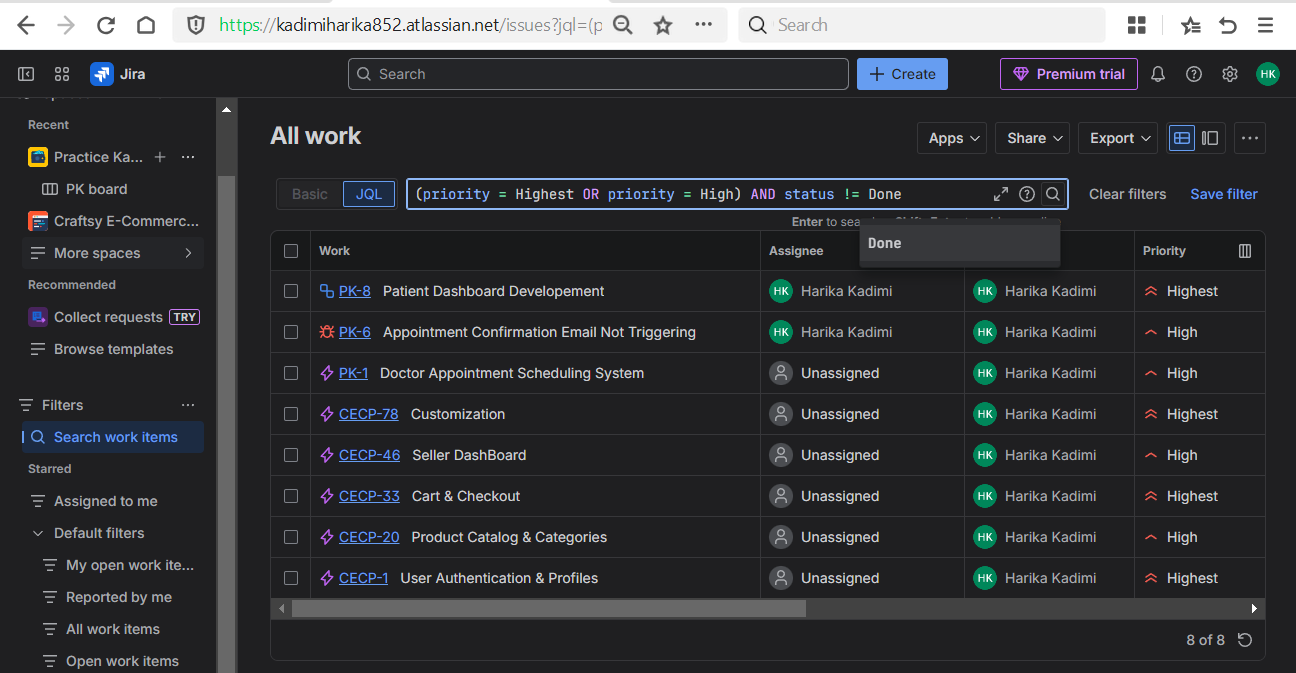
Assignee IN (“Harika Kadimi”, Priya)



* + 1. **Highest or High priority unfinished issues**

Used for risk and escalation tracking.

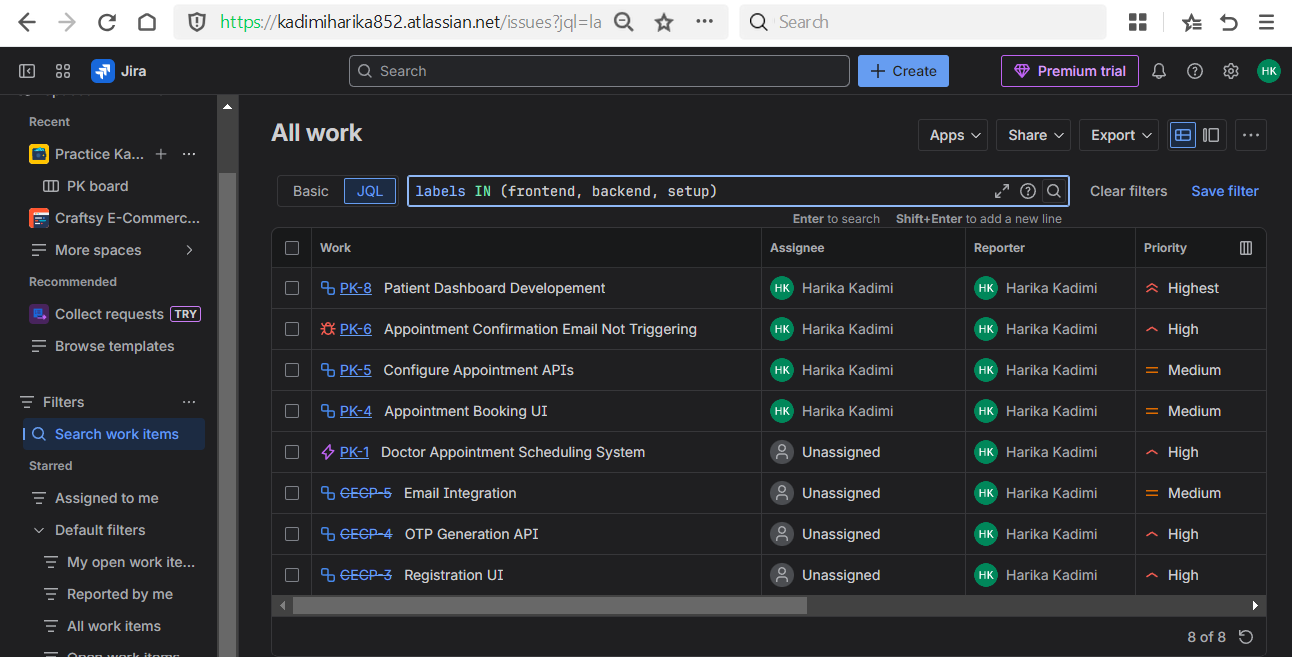
*(prioiry = Highest OR priority = High) AND status != Done*



* + 1. **Filter by multiple lables**

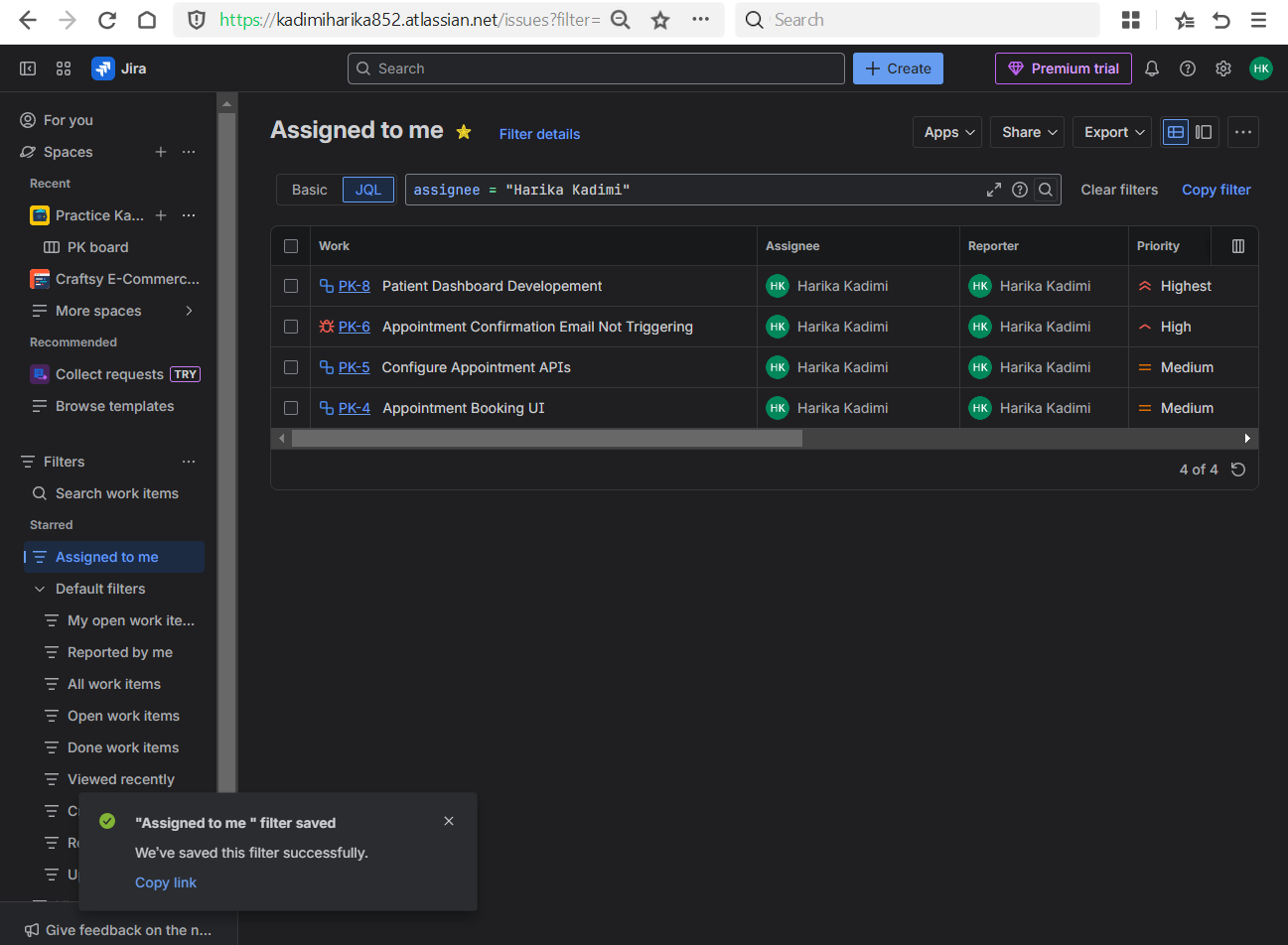
Used to filter the issues quickly using multiple lables.

*Labels IN (frontend, backend, setup)*



* Saved filters

Used to save the filters for quick access of specific issues.



**Day 5 – Dashboards & Reports**

**Learn**

* Why managers use dashboards

Dashboards are used by managers, team leads, and stakeholders to get a quick overview of project progress without checking individual issues.

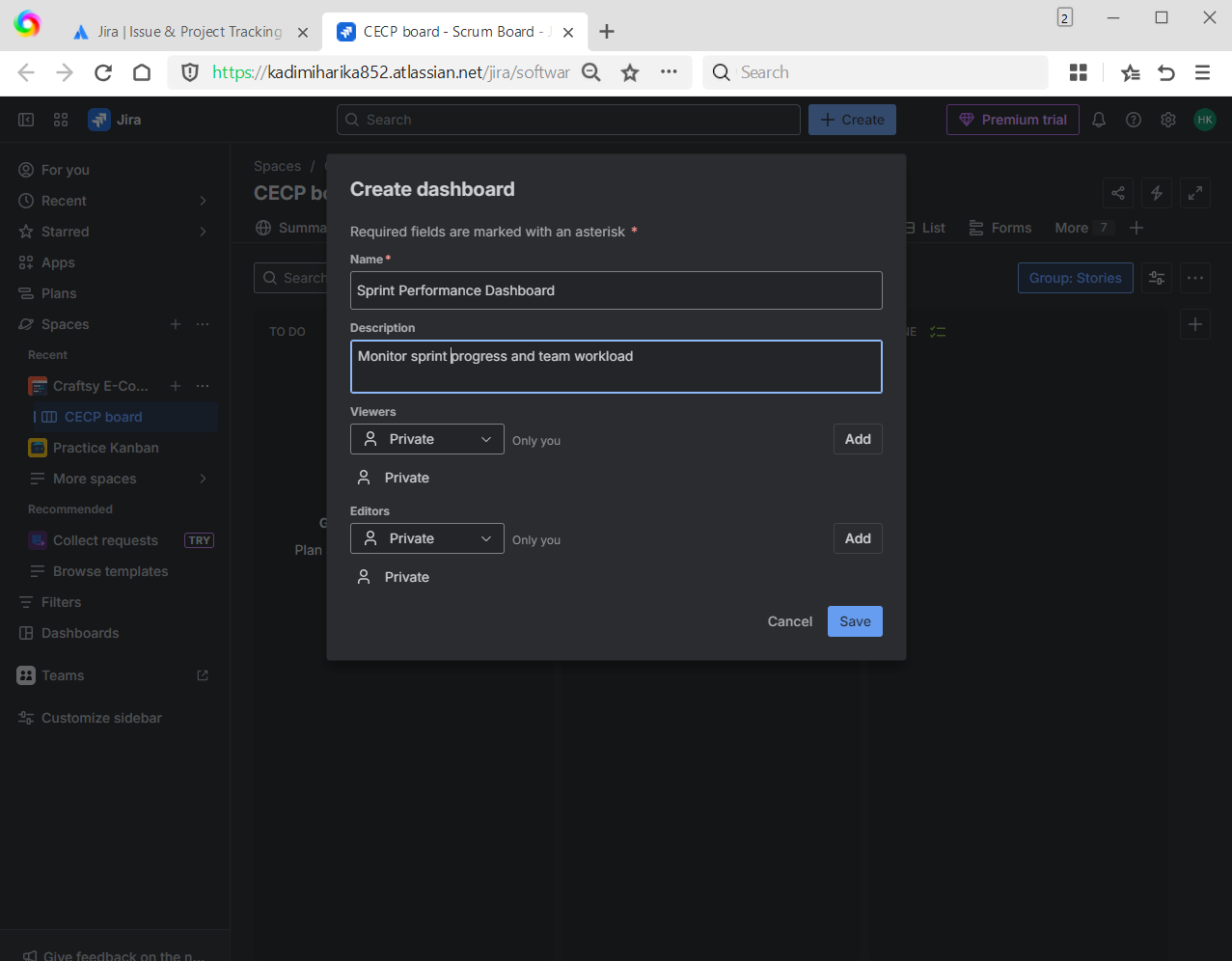
* Agile reports importance

Learned how reports help teams understand sprint performance, work completion rate, and identify delays or bottlenecks.

**Practical**

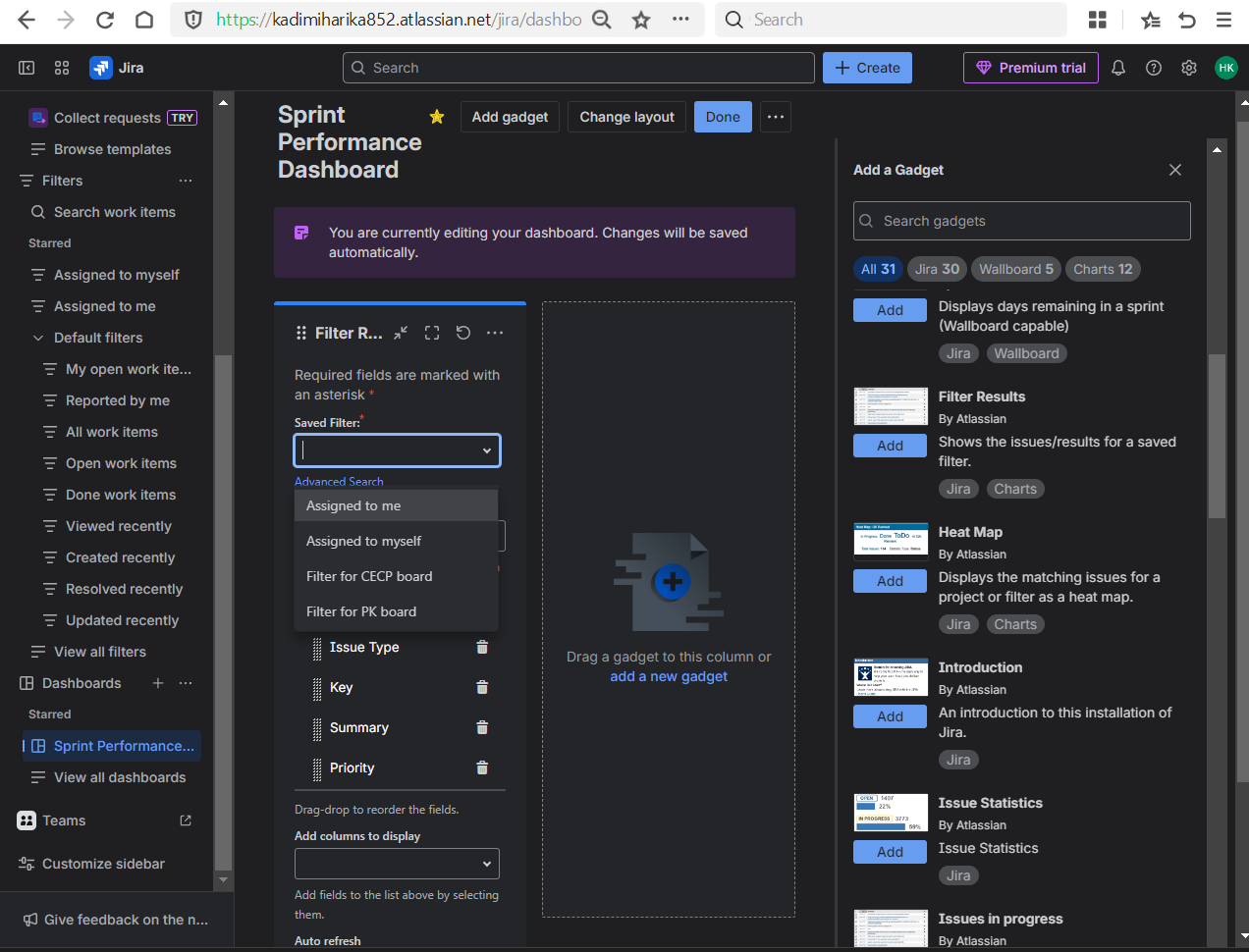
* Created a dashboard

Created a **custom dashboard** named **Monitor sprint progress and team workload** for the project.

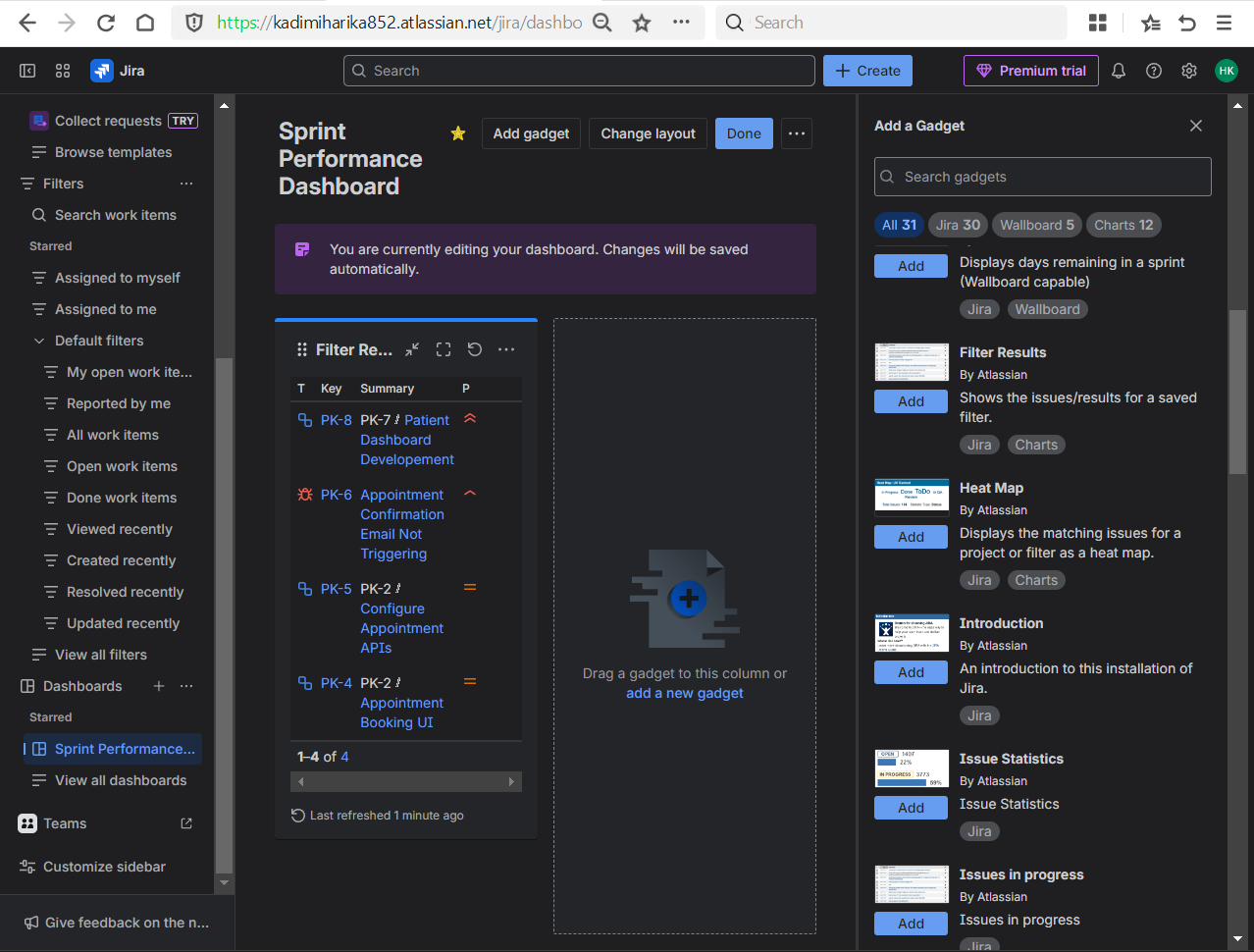


* Added gadgets:
  + Filter Results - to display issues based on saved JQL filters.

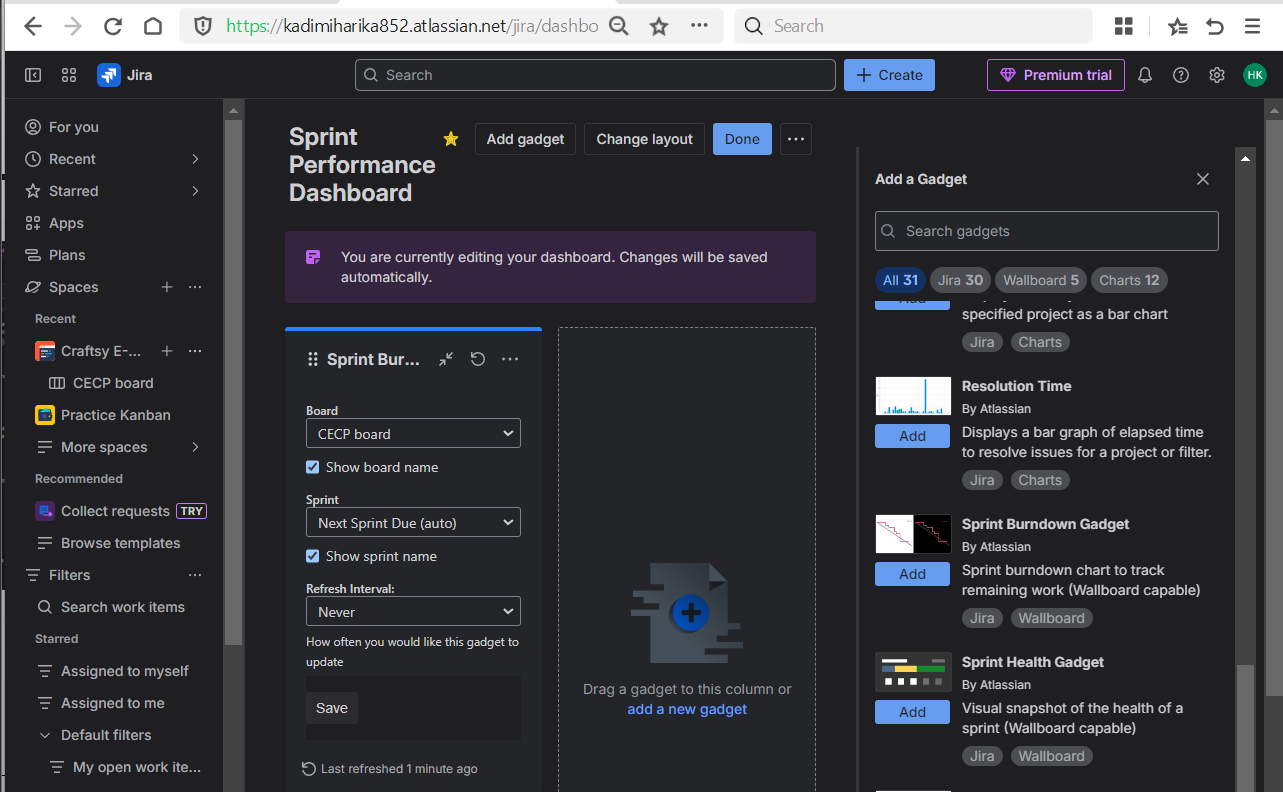
Example : assigned to myself

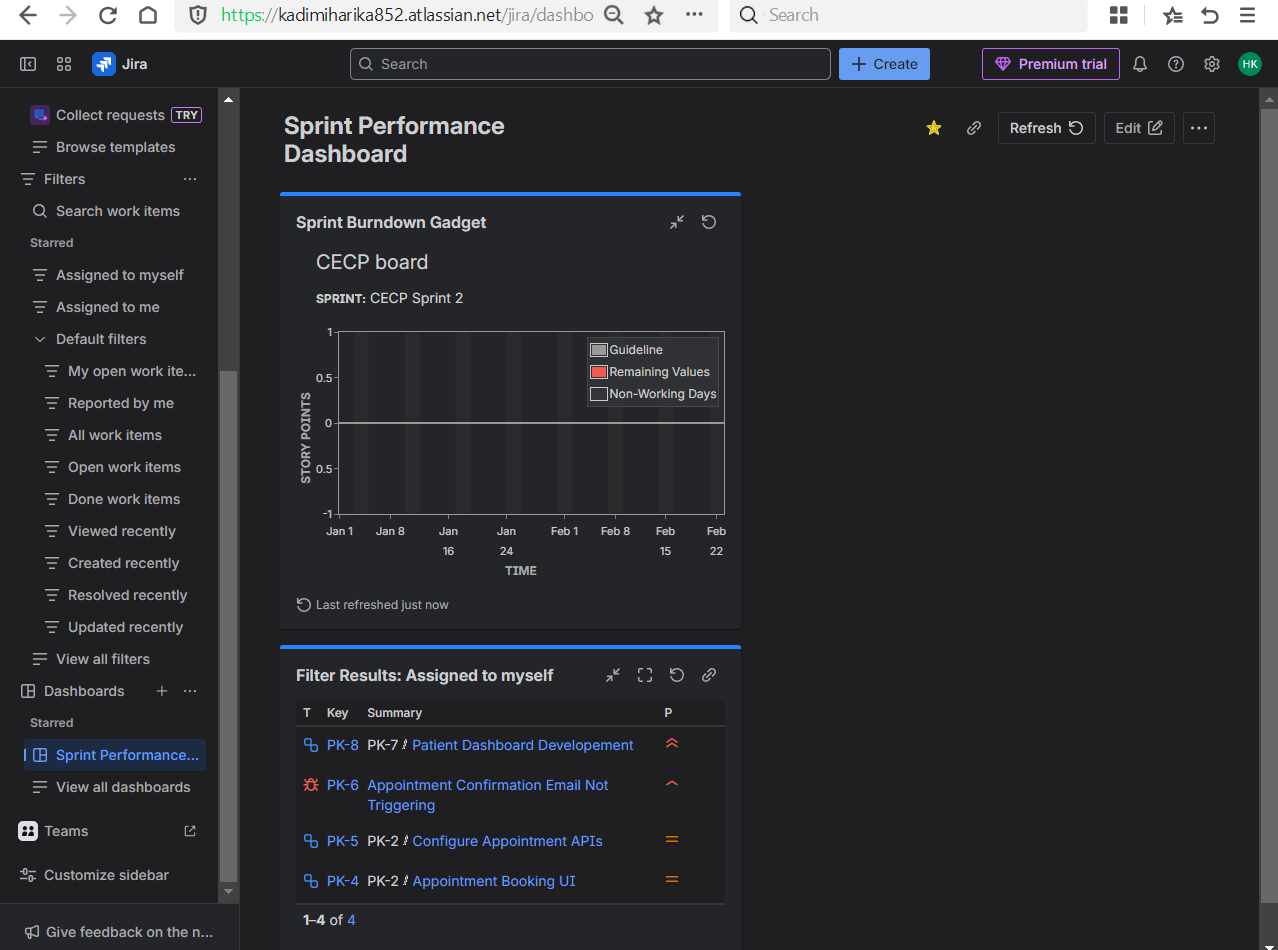


* After saving the filter the result as shown as below.



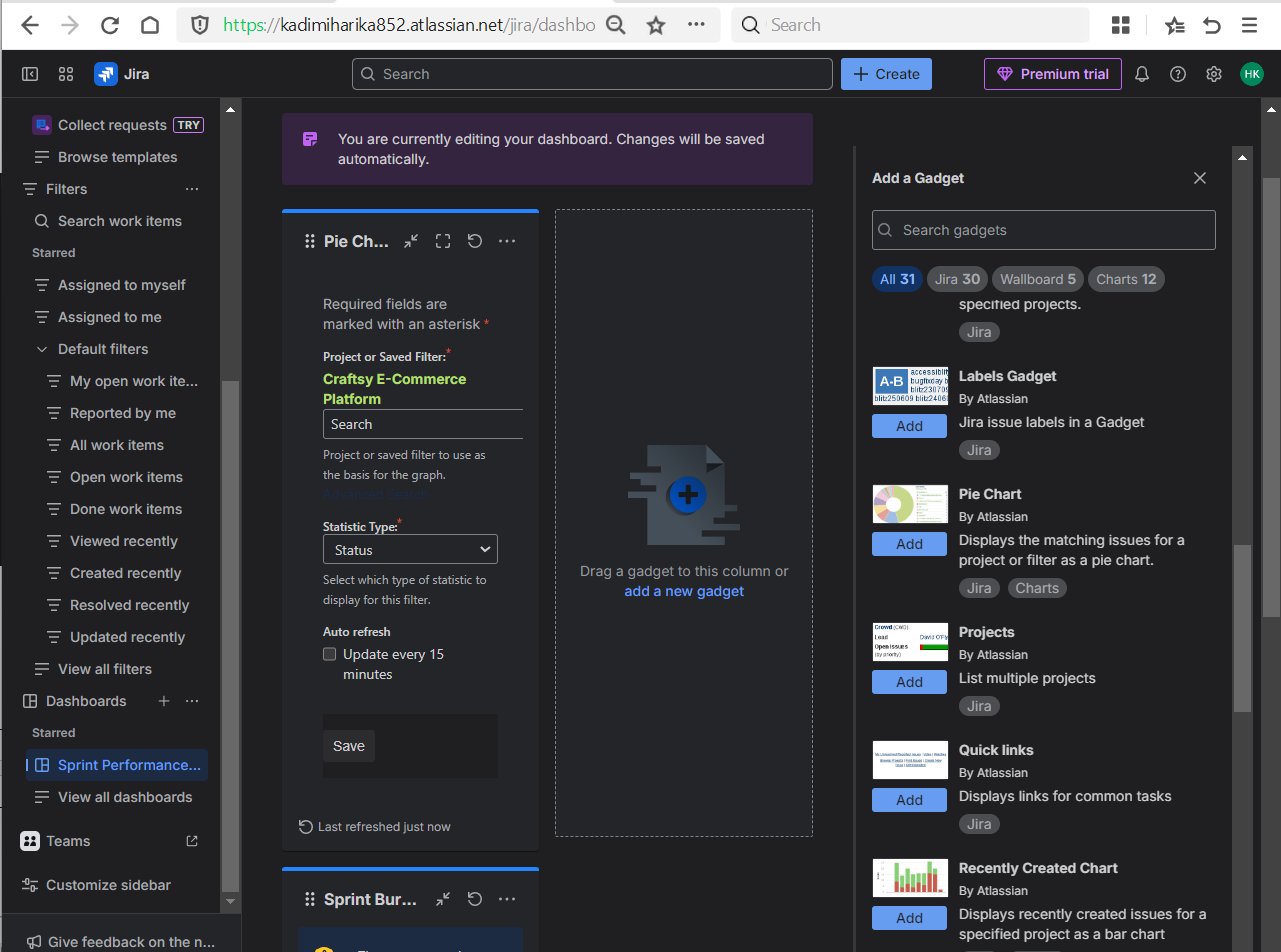
* + Sprint Burndown Chart - to track remaining work during a sprint.

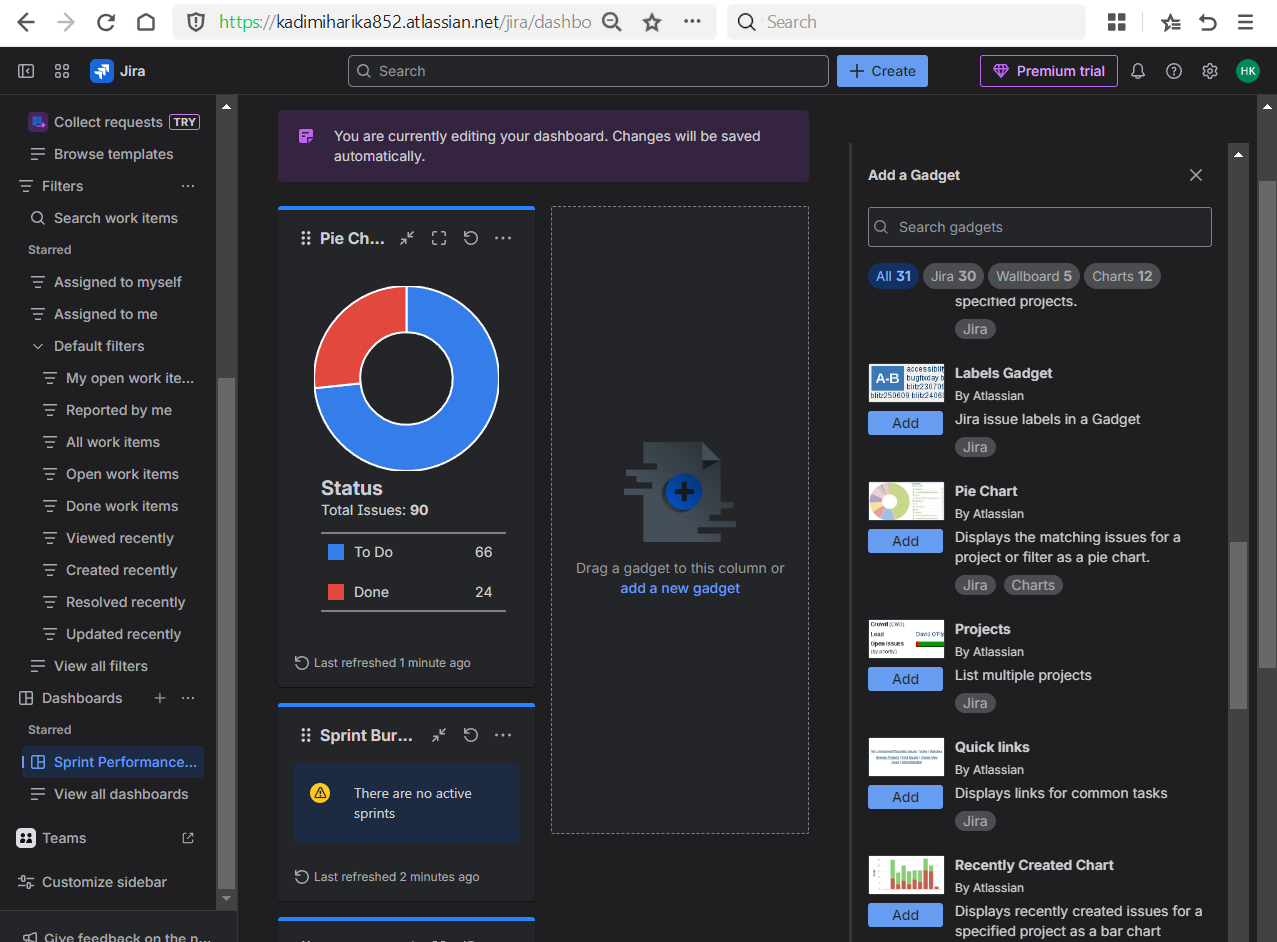




Actually the sprint data is not updated enough to show real progress

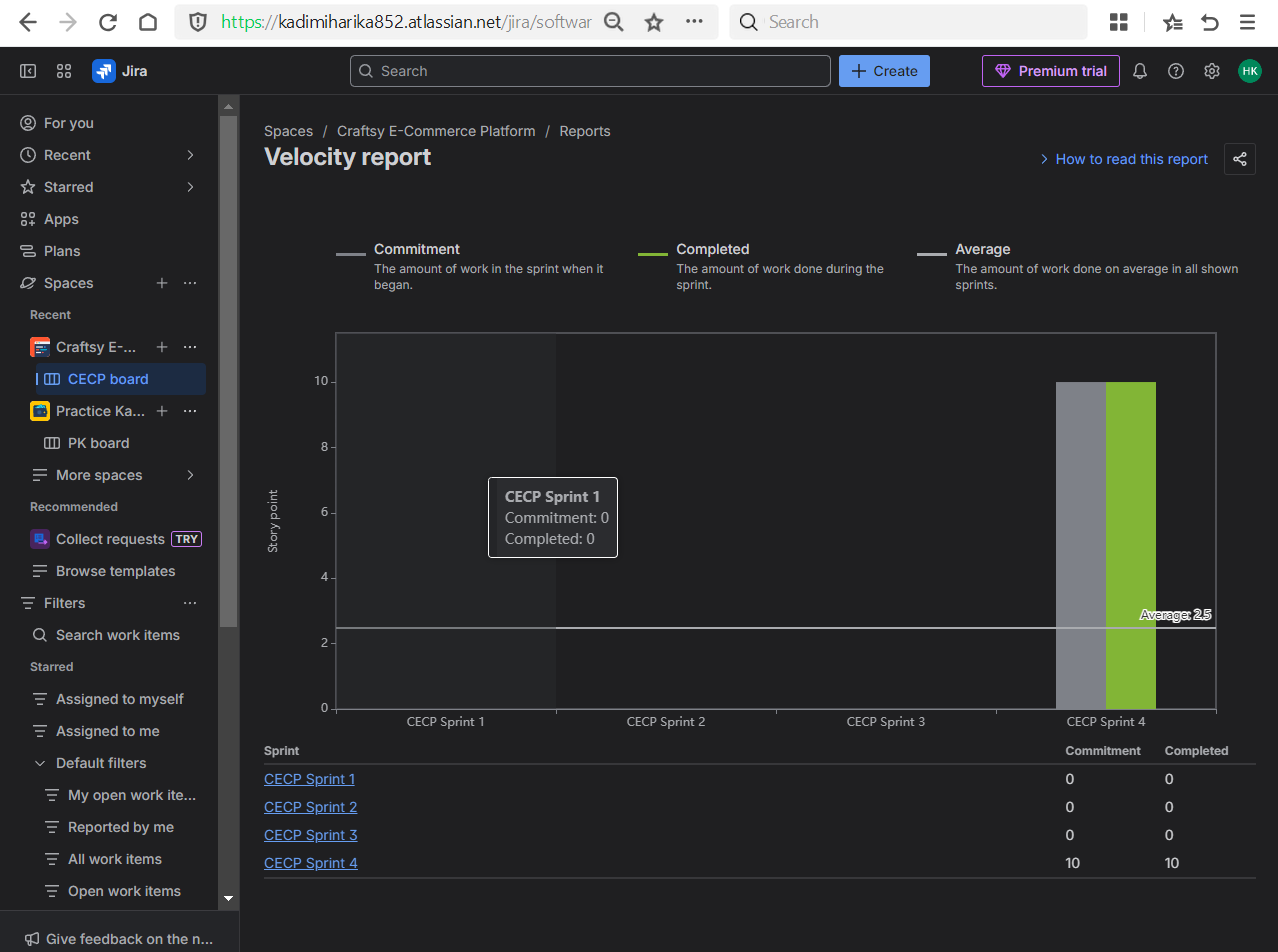
* + Pie Chart - to show issue distribution by status or priority.





Showing that I have completed 24 issues and 66 issues have to work on.

* Viewed:
  + Velocity report to understand team capacity across sprints.



* + Burndown chart to analyze daily progress.



**Day 6 – Custom Fields + Real Project Simulation**

**Learn**

* Custom fields (what companies customize)

Custom fields are used to capture additional information required by organizations that is not available by default in Jira.

* Real bug tracking flow

Understood the complete **bug lifecycle** followed in real projects from reporting to closure.

New -> Assigned -> Open -> Fixed -> Testing -> Verified -> Closed.

**Practical**

* Added a custom field (Severity)

Created a custom field named **Severity** to classify bugs based on impact.

* Created bugs with:
  + Steps to reproduce
  + Expected vs actual result

Practiced realistic bug movement across workflow statuses.

* Simulated:
  + Bug → In Progress → Fixed → Done