# Week-2: Project Management Tool — JIRA

Using JIRA and SCRUM for MSc Research Project Management: A Modern Alternative to Traditional Tools

Introduction

Effective project management is a cornerstone of successful postgraduate research. Yet, many students and researchers still rely on rigid, timeline-based tools such as **Microsoft Project**, which follow a *Waterfall* approach — linear, sequential, and often unsuitable for the dynamic nature of research.

In contrast, **JIRA**, a product by Atlassian originally designed for software engineering teams, offers a **flexible, agile-based approach** to managing tasks, uncertainties, and evolving priorities. When used with the **SCRUM framework**, JIRA can transform a Master's research project into a structured yet adaptive workflow — ideal for data-driven and iterative fields such as Artificial Intelligence, Computer Science, and Data Analytics.
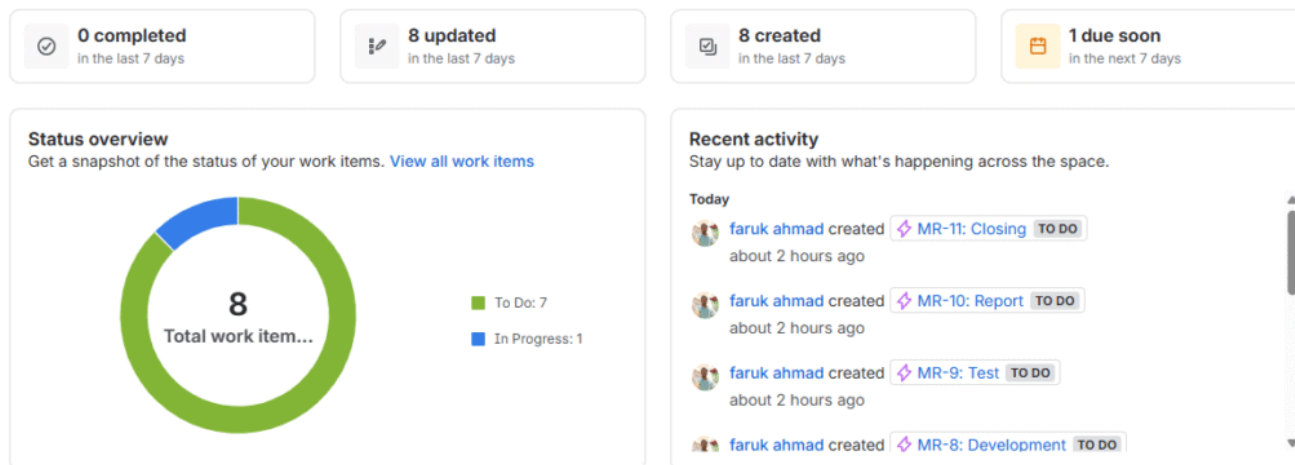
JIRA Dashboard Summary

Why Agile Methodology Suits Research Projects

The Problem with Traditional Planning

Traditional project management assumes that every activity can be pre-defined and scheduled with exact start and end dates. In reality, research — especially empirical or computational research — is **non-linear**. Unexpected results, failed experiments, or new findings often require revising hypotheses or methods.

A rigid Gantt chart may illustrate dependencies well, but it cannot easily adapt to these changes without reworking the entire timeline. This inflexibility can cause unnecessary stress and inefficiency in a research setting.

The Agile Philosophy

Agile project management, on the other hand, thrives in uncertainty. It emphasizes iteration, feedback, transparency, and flexibility. These principles perfectly align with how research evolves — through experimentation, reflection, and

refinement.

- **Iteration** — Dividing work into short, repeatable cycles called sprints.
- **Feedback** — Reflecting after each sprint to adjust goals or priorities.
- **Transparency** — Making progress visible through boards and metrics.
- **Flexibility** — Reordering or redefining tasks without breaking the workflow.

Overview of JIRA as a Research Management Tool

JIRA is a cloud-based tool that enables teams to manage tasks, visualize workflows, and track progress using **Agile methodologies** like SCRUM or Kanban. While its origin lies in software development, JIRA's architecture — built around issues, epics, sprints, and boards — is highly customizable. Each feature can be reinterpreted in the context of academic research.
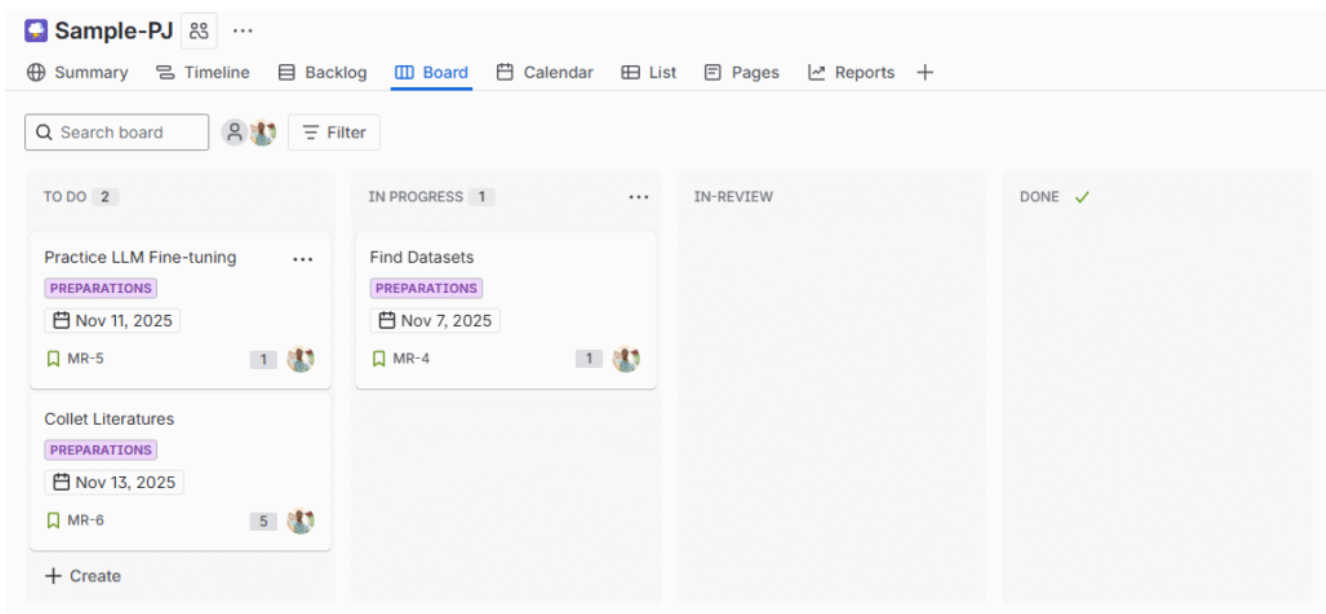
Setting Up JIRA for a Research Project

Step 1: Create a Project and Select the SCRUM Template

Upon creating a new project, select the **SCRUM template**. This automatically generates a **Backlog**, **Board**, and **Sprint structure**. Each issue in JIRA represents a unit of work — for example, "Read papers on Transformer models" or "Train the baseline classifier."

Choosing SCRUM over Kanban is recommended for MSc research because SCRUM enforces a rhythm of progress through time-boxed sprints and measurable deliverables.
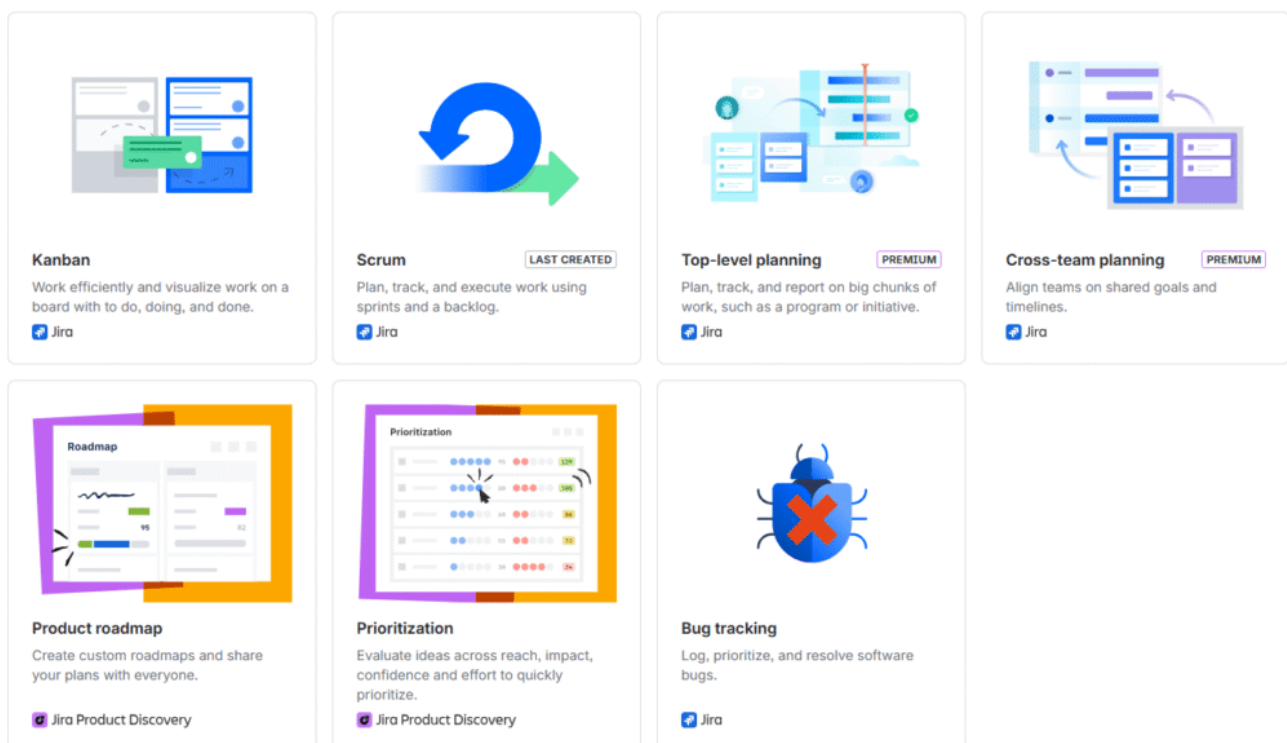

Project Template

Step 2: Define Epics — The Major Components of Research

An Epic in JIRA represents a broad objective or milestone. This hierarchical structure provides a clear top-down view of the project scope while allowing detailed breakdowns underneath. A sample backlog with Epics might look like this-

Epic ID	Epic Name	Description
EPIC-1	Literature Review	Reviewing existing work and identifying research gaps.
EPIC-2	Data Preparation	Collecting and preprocessing datasets.
EPIC-3	Model Development	Implementing and training models.
EPIC-4	Evaluation	Assessing model performance and validating results.
EPIC-5	Report Writing	Drafting dissertation chapters and visualizing findings.
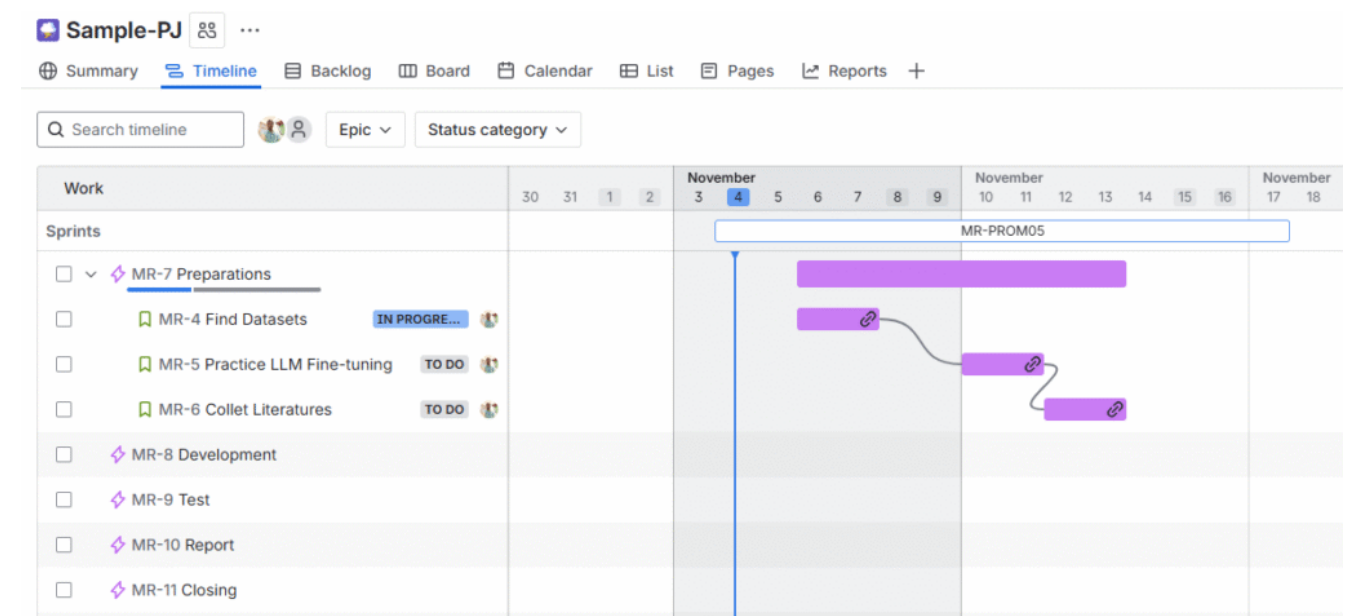
Project Timeline

Step 3: Break Down into Stories and Tasks

Each Epic is decomposed into stories or tasks that can be completed within a sprint. For example, under Model Development, tasks might include implementing the baseline model, evaluating performance, and optimizing parameters.

- Implement baseline model (Logistic Regression)
- Evaluate with F1-score metric

- Optimize hyperparameters using GridSearchCV

Each task includes details such as description, assignee, labels, attachments, and deadlines — allowing precise tracking and accountability for every research action.
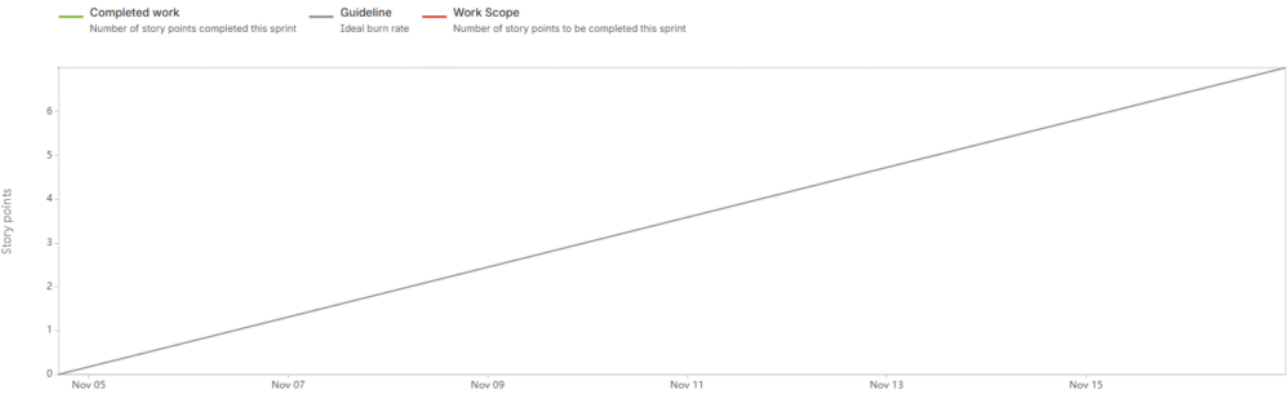
Step 4: Plan and Execute Sprints

A sprint is a short, fixed period (usually one or two weeks) where a defined set of tasks is completed. Each sprint might focus on a distinct research phase such as literature review, data collection, or model testing.

- **Sprint 1:** Gather and summarize academic papers
- **Sprint 2:** Collect datasets and perform preprocessing
- **Sprint 3:** Implement baseline models
- **Sprint 4:** Perform evaluation and result interpretation
- **Sprint 5:** Write and refine dissertation chapters

Step 5: Track Progress with Boards and Charts

JIRA's SCRUM board visually represents all active tasks in columns such as **To Do → In Progress → Under Review → Completed**. Moving tasks across the board provides visibility and motivation. Additionally, burndown charts and velocity reports offer quantitative insights into progress and productivity.

Date - November 4th, 2025 - November 17th, 2025

Sprint goal - - Creating Research Plan/Timeline - Preparing on/about LLM fine-tuning - Dataset Review

Burnup chart

Step 6: Integrate with Supporting Tools

JIRA integrates with popular tools that support academic workflows:

- **GitHub / GitLab** — Link code commits to tasks.
- **Confluence** — Maintain progress logs and meeting notes.
- **Google Drive / OneDrive** — Attach datasets and drafts.

Automations can also be configured — for example, automatically moving a task to "Review" when a GitHub pull request is merged.

Advantages of JIRA in an Academic Research Context

JIRA provides a structured yet adaptable way to handle the complexity of postgraduate research. Below are some of the most beneficial features for academic users:

Feature	Description	Benefit to Research

Backlog & Sprint Planning	Organizes research objectives dynamically.	Ensures continuous progress without rigid scheduling.
Epics and Stories	Hierarchical structuring of goals and subtasks.	Makes large projects manageable and traceable.
Burndown Charts	Displays remaining work against time.	Provides quantifiable progress tracking.
Comments & Mentions	Facilitates supervisor collaboration.	Reduces email dependency.
Custom Workflows	Statuses like "Data Collected" or "Under Review."	Reflects real research phases.
Reports & Dashboards	Generates visual summaries.	Useful for dissertation progress reports.

Comparison: JIRA vs. Microsoft Project

While both JIRA and Microsoft Project are capable project management platforms, their methodologies differ significantly. MS Project follows a traditional Waterfall model, while JIRA embraces iterative Agile practices, making it more suitable for evolving academic research.

Criteria	JIRA (SCRUM-based)	MS Project (Waterfall-based)
Approach	Iterative and adaptive	Linear and sequential

Flexibility	Reprioritize tasks mid-project	Requires re-baselining schedules
Collaboration	Real-time comments and tagging	Primarily individual use
Visualization	Interactive boards and metrics	Static Gantt charts
Learning Curve	Moderate and intuitive	Steeper, requires PM experience
Integration	GitHub, Confluence, Cloud tools	Limited integrations
Suitability for Research	Ideal for evolving research	Best for fixed, predictable projects

---

Example: MSc Research Managed via JIRA

An MSc student researching "AI-Based E-commerce Recommendation Systems" could structure JIRA as follows:

- **Epic 1:** Literature Review — Collect papers on collaborative filtering and NLP techniques.
- **Epic 2:** Data Collection — Scrape and clean review datasets.
- **Epic 3:** Model Implementation — Train and evaluate models.
- **Epic 4:** Reporting — Draft methodology and discussion chapters.

Each Epic is divided into sprints, and progress is visualized through burndown charts. Confluence integration can be used for documentation and supervisor feedback.

Challenges and Best Practices

Common Challenges

While JIRA provides powerful functionality, new users may face challenges such as over-configuration, solo usage without SCRUM ceremonies, and time-tracking fatigue. Awareness of these limitations helps maintain simplicity and focus.

Best Practices

- Start simple — one board, 3-4 columns, and a few sprints.
- Hold weekly retrospectives to refine workflow and document lessons learned.
- Use supervisor dashboards for accountability.
- Leverage labels and filters for quick milestone tracking.

Conclusion

JIRA, when combined with the SCRUM methodology, provides a **modern, adaptive, and data-driven approach** to managing academic research projects. It bridges the gap between structured documentation and flexible exploration — an essential balance in postgraduate research.

Unlike MS Project, which enforces predefined linearity, JIRA supports **iterative experimentation**, **transparent progress tracking**, and **real-time adaptability**. For MSc students navigating complex, evolving research tasks, it serves not

merely as a project tracker but as a **comprehensive research companion** — aligning daily work with long-term academic goals.

References

1. https://www.atlassian.com/software/jira
2. https://www.atlassian.com/software/jira/guides/getting-started/introduction