BSCSS3001 Software Engineering

Integrating GenAI into SEEK learning Portal |2024 T2

Project Presentation

Milan | Ritik | Vedanshi | Tamanna | Navdeep | Shubham | Rohan | Tabish

TEAM 8

8

Problem Statement

Effective integration of Generative AI into programming learning environments

Learners in the IITM BS degree program primarily engage with content through the SEEK portal, which offers learning videos, resources, assignments, and quizzes, supporting self-paced learning. With the rapid growth of generative AI (GenAI), this term's software engineering project aims to explore how GenAI can be effectively integrated into learning environments like SEEK.

User Stories

1. **As a** learner,
I want to register/login on the platform,
So that I can access and view the available learning content.

2. **As a** learner,
I want the option to generate a summary for any specific video after watching it,
So that I can reinforce my understanding of the video content.

3. **As a** learner,
I want GenAI to give real-time hints while I code,
so I can improve instantly.

4. **As a** learner,
I want GenAI to analyze my programming quiz attempts and provide detailed feedback
So that I can enhance my programming skills.

User Stories

1. **As a** learner,
I want GenAI to evaluate my subjective assignments for accuracy, grammar, and missed points,
So that I can understand areas for improvement and enhance my answers.

2. **As an** instructor,
I want to utilize GenAI to automatically evaluate and provide feedback on code Submissions
So that I can offer more detailed and helpful feedback to students.

3. **As an** administrator,
I want to add, delete, and update content and assignments,
So that I can ensure learners have access to the most relevant and up-to-date materials.

4. **As an** administrator,
I want to ensure that all GenAI-enhanced features undergo rigorous unit testing,
So that I can verify their functionality and reliability.

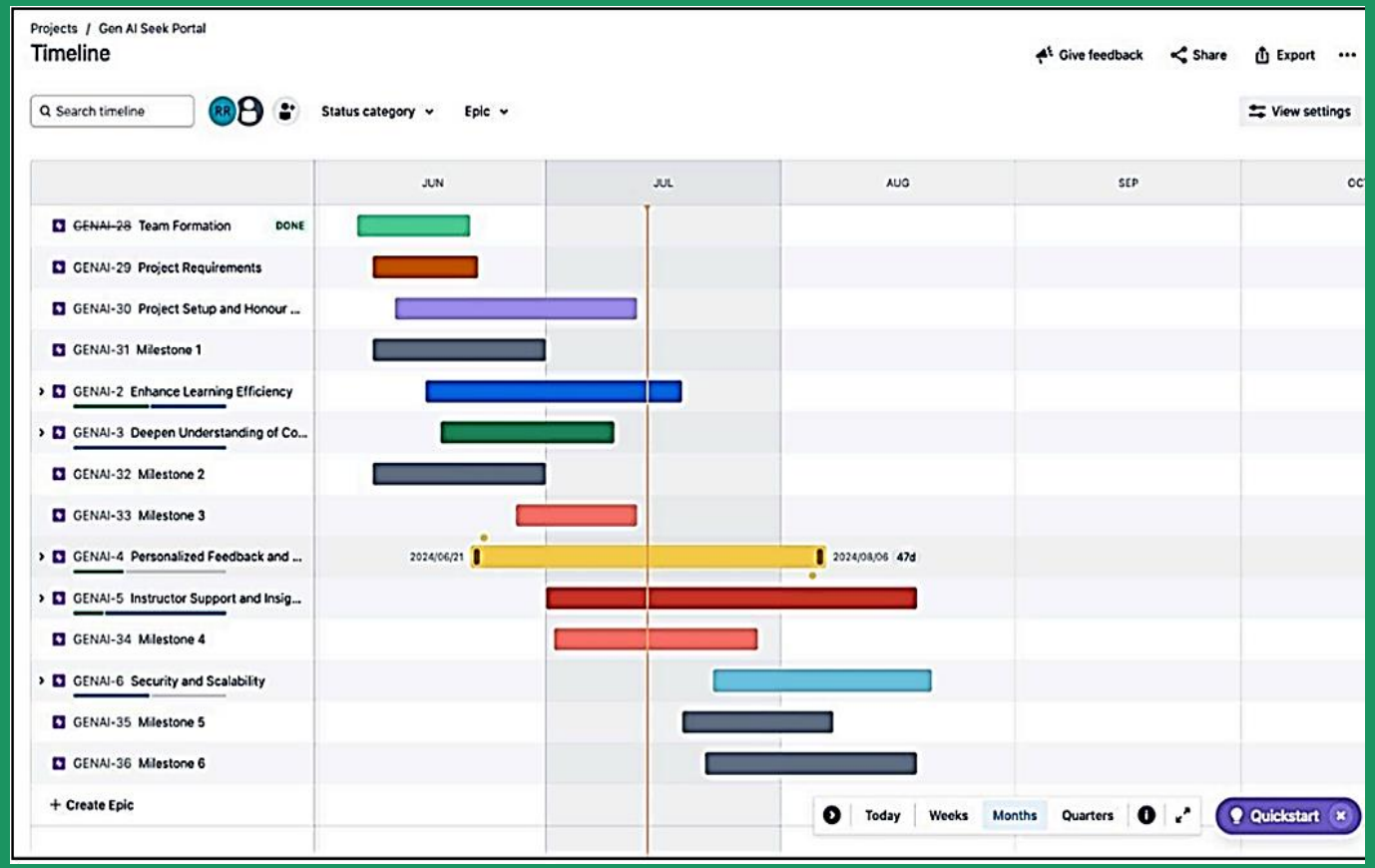
Project Scheduling

We use JIRA/ATLAS for project management, creating and assigning Epics, Activities, and Tasks. We've customized JIRA with workflows, dashboards, and goal planning, aligning the project schedule with the course timeline.

The project schedule is aligned to the course project timeline:

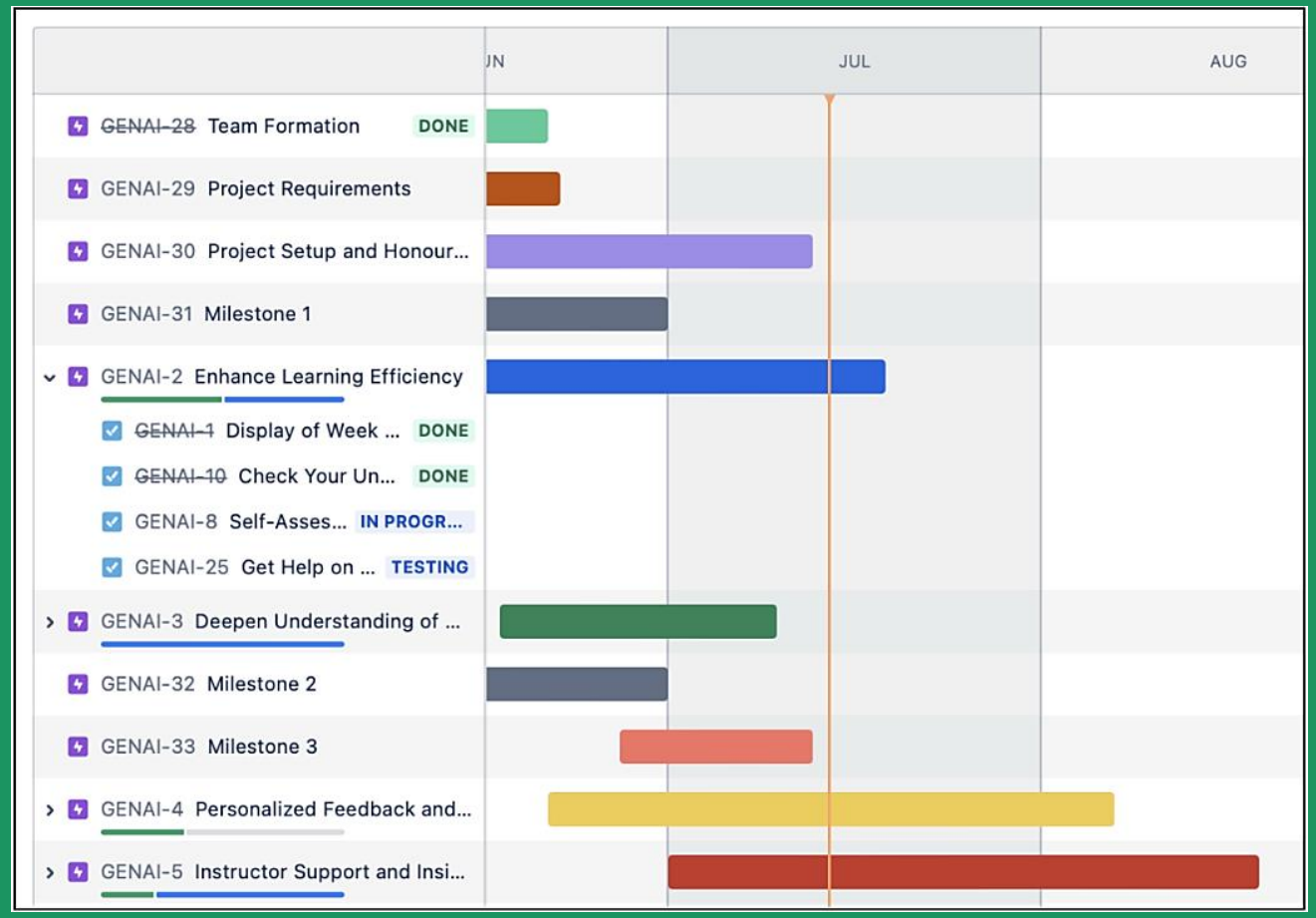
Milestone 1	30th June
Milestone 2	30th June
Milestone 3	12th July
Milestone 4	28th July
Milestone 5	7th August
Milestone 6	18th August

Project Timeline: Gantt Chart – Epics and Sprints



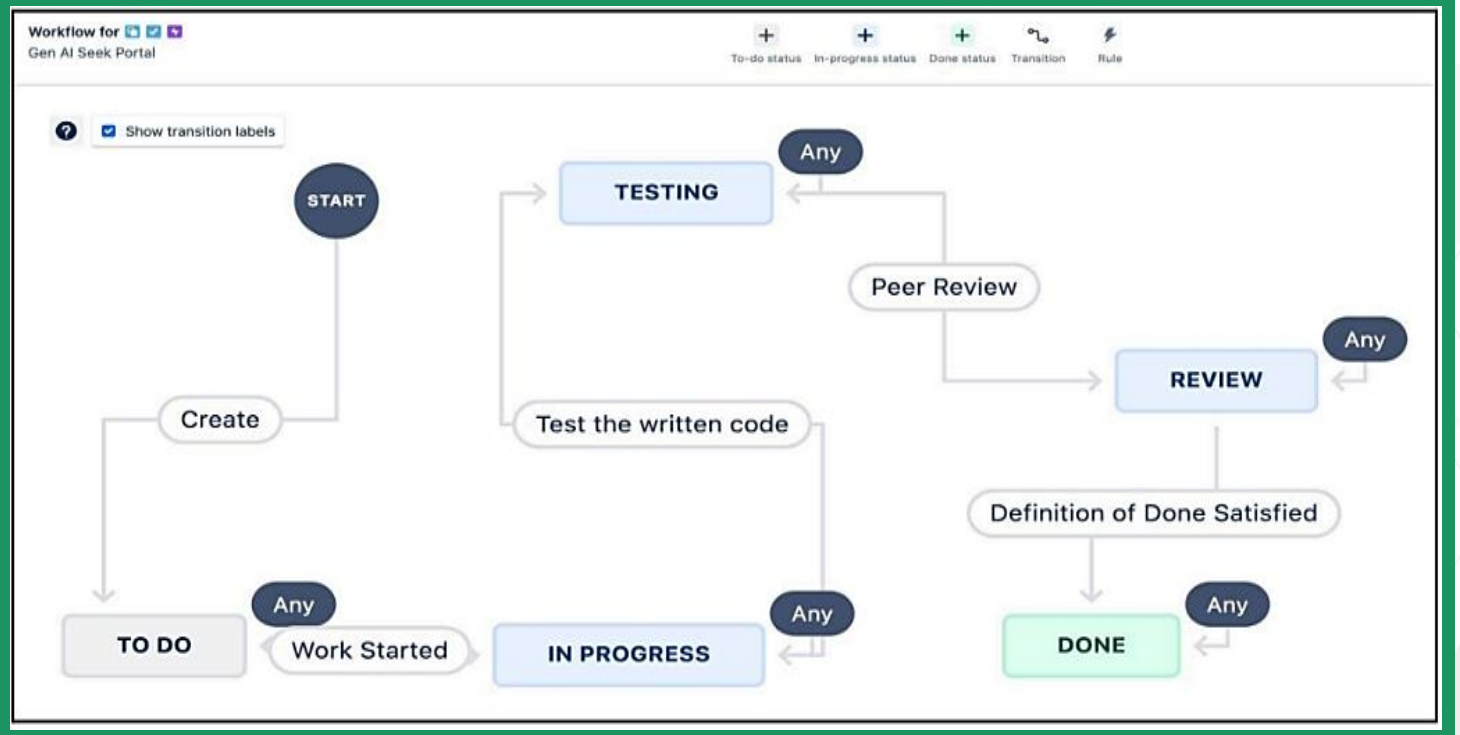
- Jira also has Gantt Charts.
- The Gantt chart is a commonly used project management tool that displays work completed over a period of time.
- Individual tasks were assigned to specific members. Tasks were also assigned backup members based on availability and priority.

Project Timeline: Gantt Chart – Tasks/Issues



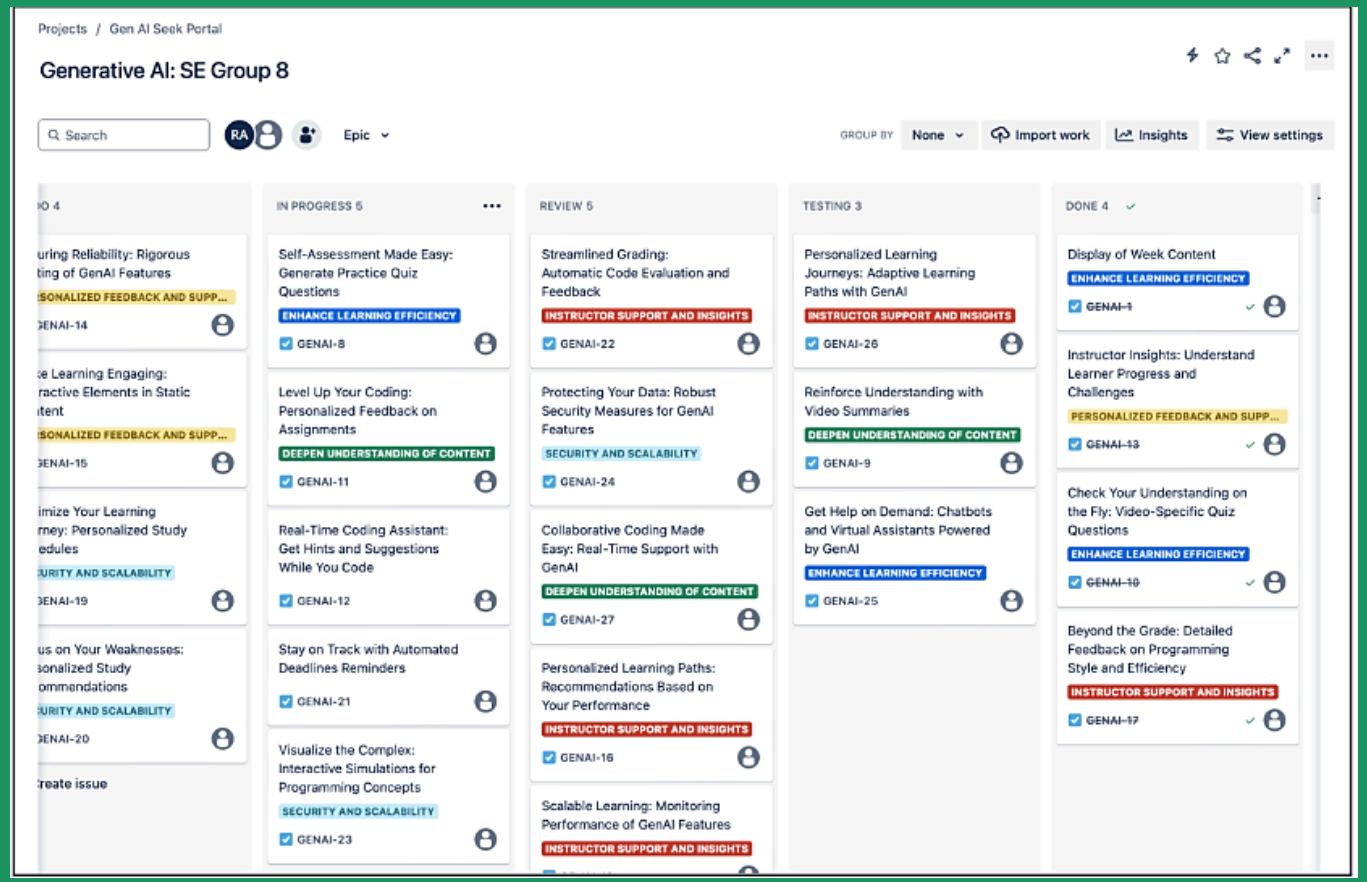
- The tasks allow us to integrate scheduling with start date and end date.
- Here we can also assign tasks to team members for action.
- Each task also has a status that shows as To-Do, In progress or Done.

Project Timeline: Custom Workflow



- We customized task workflows in Jira by setting a sequential status flow.
- Tasks move through statuses and trigger a review step upon completion, changing to Done after review.

Project Scheduling:
KANBAN Board - Overall

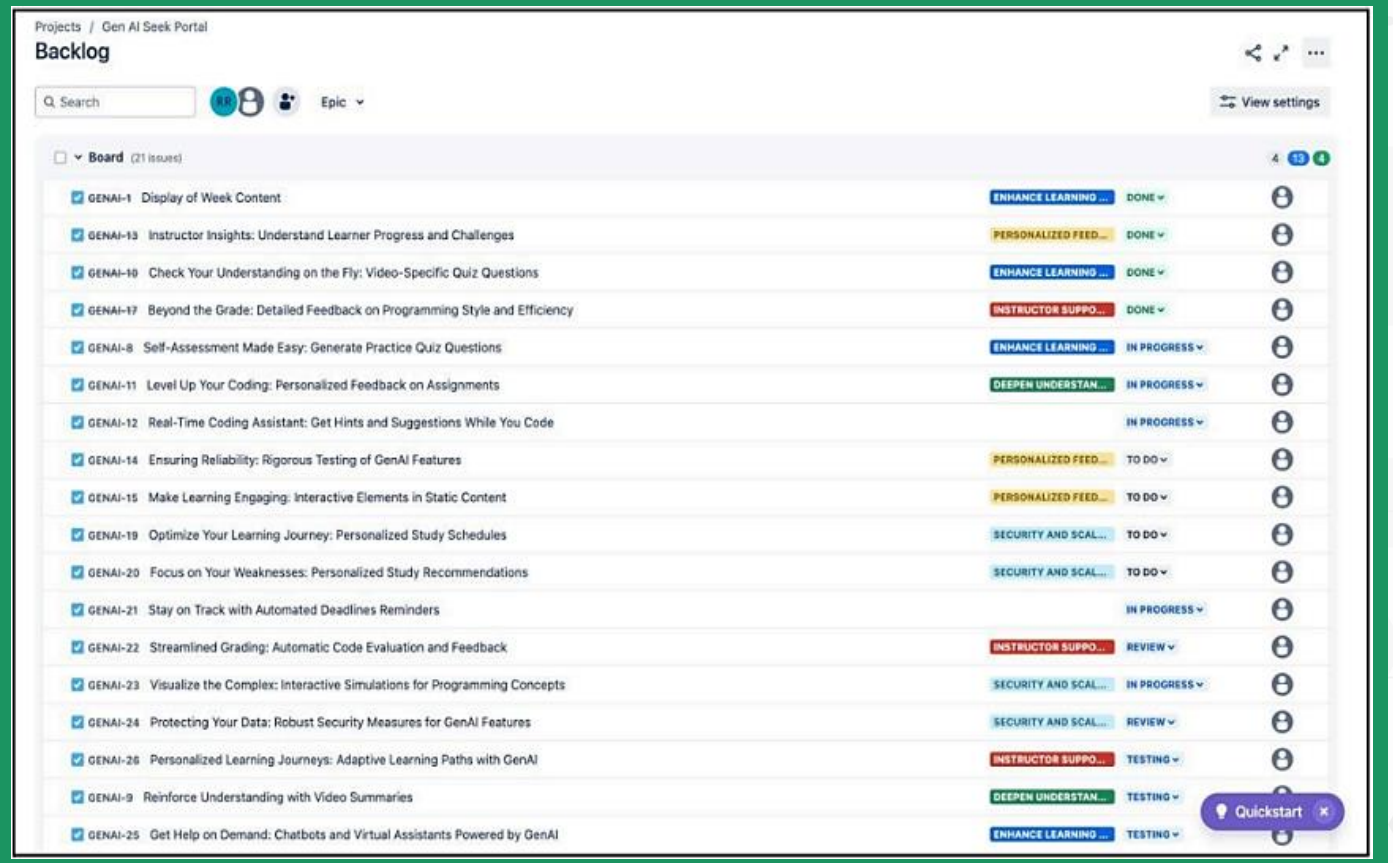


- Kanban is an agile framework that visualizes tasks on a board with statuses:

  To Do, In Progress, Review/Test, Done, and Blocked.

- This helps our team track task progress and requirements.

Project Scheduling: Backlogs



- The backlog is a prioritized list of tasks and features for the project.
- It serves as the source for future work, helping us manage and schedule tasks based on priority and project needs.

API Test Cases

API unit testing ensures that individual API endpoints function correctly by validating their responses against expected outcomes.

For each user story, new API endpoints are created

User Login API

Input

```
{

"username": "johndoe",

"password": "securePassword123"

}
```

- **API being tested:** /user_login
- **Expected Output**:
  - Status Code: 302
  - Response Body: Redirect to user dashboard
- **Actual Output:**
  - Status Code: 302
  - Response Body: Redirect to /user_dashboard
- **Result:** Success

Admin Login API

Input

```
{
"ad_username": "janesmith",
"passwrd": "adminSecurePassword456"
}
```

- **API being tested:** /admin_login
- **Expected Output**:
  - Status Code: 302
  - Response Body: Redirect to admin dashboard
- **Actual Output:**
  - Status Code: 302
  - Response Body: Redirect to /admin
- **Result:** Success

Create Course API

Input

```
{

"cat_name": "Introduction to Programming",

"cat_details": "A beginner course on programming.",

"cat_img": "<binary data>"

}
```

- **API being tested:** /admin/create_course
- **Expected Output**:
  - Status Code: 302
  - Response Body: Redirect to course list
- **Actual Output:**
  - Status Code: 302
  - Response Body: Redirect to /admin
- **Result:** Success

Update Course API

Input

```
{

 "cat_name": "Introduction to Programming",

 "cat_details": "A beginner course on programming.",

 "cat_img": "<binary data>"

}
```

- **API being tested:** /admin/<cat_id>/update
- **Inputs**: Path Parameter: {cat_id} = 1
- **Expected Output**:
  - Status Code: 302
  - Response Body: Redirect to course list
- **Actual Output:**
  - Status Code: 302
  - Response Body: Redirect to /admin
- **Result:** Success

Create Content API

Input

```
"content_type": "assignment",

"assignment_question": "What is the complexity of binary search?",

"option_a": "O(n)",

"option_b": "O(log n)",

"option_c": "O(n log n)",

"option_d": "O(1)",

"correct_answer": "O(log n)"
```

- **API being tested:**
  /admin/<cat_id>/create_content
- **Inputs**: Path Parameter: {cat_id} = 1
- **Expected Output**:
  - Status Code: 302
  - Response Body: Redirect to content list
- **Actual Output:**
  - Status Code: 302
  - Response Body: Redirect to /admin/<cat_id>/contents
- **Result:** Success

Update Content API

Input

```
{
 "content_type": "assignment",
 "assignment_question": "Updated question?",
 "option_a": "Updated option",
 "correct_answer": "Updated option"
}
```

- **API being tested:** /admin/<cat_id>/<content_id>/update
- **Inputs**: Path Parameters: {cat_id} = 1, {content_id} = 1
- **Expected Output**:
  - Status Code: 302
  - Response Body: Redirect to content list
- **Actual Output:**
  - Status Code: 302
  - Response Body: Redirect to /admin/<cat_id>/contents
- **Result:** Success

Prompt with video API

Input

```
{

 "prompt": "Explain the concept of polymorphism."

}
```

Expected Output (Response body)

```
{

 "Answer": "Polymorphism allows objects to be treated as instances of their
      parent class rather than their actual class."

}
```

- **API being tested:** /process_prompt
- **Expected Output**:
  - Status Code: 302
- **Actual Output:**
  - Status Code: 302
  - Response Body: As Expected
- **Result:** Success

Prompt with web API

Input

```
{

 "prompt": "Explain the concept of polymorphism."

}
```

Expected Output (Response body)

```
{

 "Answer": "Polymorphism allows objects to be treated as instances of their
      parent class rather than their actual class."

}
```

- **API being tested:** /process_web_prompt
- **Expected Output**:
  - Status Code: 302
- **Actual Output:**
  - Status Code: 302
  - Response Body: As Expected
- **Result:** Success

Coding Answer API

Input

```
{

 "coding_question": "Write square root function"

}
```

Expected Output (Response body)

```
"Answer": "def square_root(x): if x < 0: return "Square root of a negative
     number is not real." else: return x ** 0.5 # Example usage:result =
     square_root(16) print("The square root is:", result)"
```

- **API being tested:** /get_coding_answer
- **Expected Output**:
  - Status Code: 302
- **Actual Output:**
  - Status Code: 302
  - Response Body: As Expected
- **Result:** Success

Coding Answer hint API

Input

```
{
 "coding_question": "Write square root function",
 "additional_input": "a=b^2",
}
```

Expected Output (Response body)

```
'Answer": "Consider using the exponentiation operator ** with 0.5 to compute
     the square root of a number."
```

- **API being tested:** /get_coding_hint
- **Expected Output**:
  - Status Code: 302
- **Actual Output:**
  - Status Code: 302
  - Response Body: As Expected
- **Result:** Success

Check User Code API

Input

```
"code": "def square_root(x):

      return x ** 0.5

",

"input": "16,

"expected_output": 4",
```

Expected Output (Response body)

```
{

"is_correct": "True"

}
```

- **API being tested:** /check_code
- **Expected Output**:
  - Status Code: 302
- **Actual Output:**
  - Status Code: 302
  - Response Body: As Expected
- **Result:** Success

Subjective Evaluation API

Input

```
{
 "question": "What is phenomenon",
 "answer": "a fact or situation that is observed to exist or happen, especially one whose cause or explanation is in question.
}
```

Expected Output (Response body)

```
{
 "cohesiveness_feedback": "The answer is concise and directly addresses the question, providing a clear definition of 'phenomenon.",
 "grammar_feedback": "The answer is grammatically correct, but the spacing between words could be improved for better readability.",
 "plagisrism_feedback": "The given content seems to be 95% AI generated"
}
```

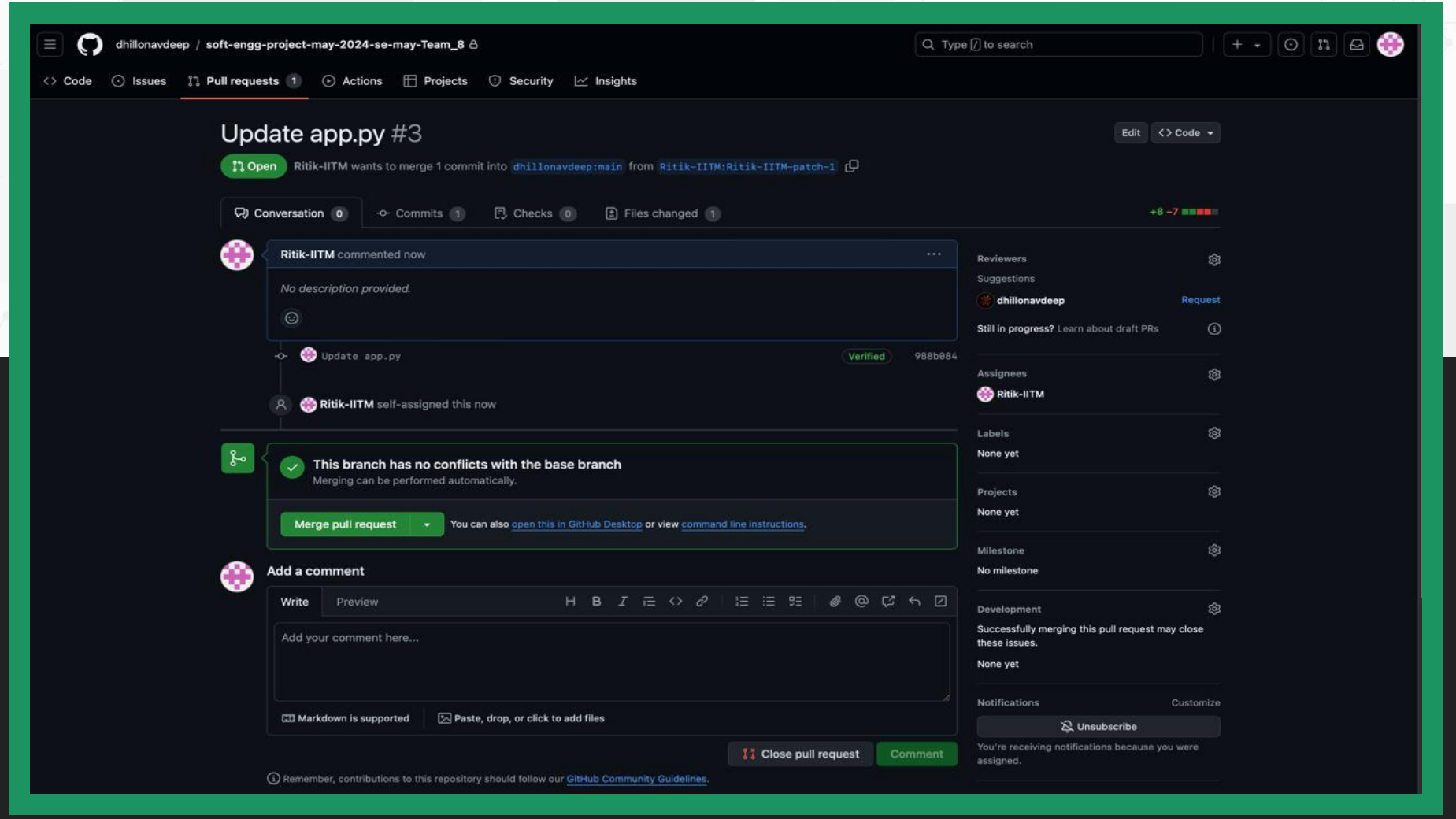
- **API being tested:** /evaluate_subjective
- **Expected Output**:
  - Status Code: 302
- **Actual Output:**
  - Status Code: 302
  - Response Body: As Expected (with minor changes)
- **Result:** Success

Code Review & Testing

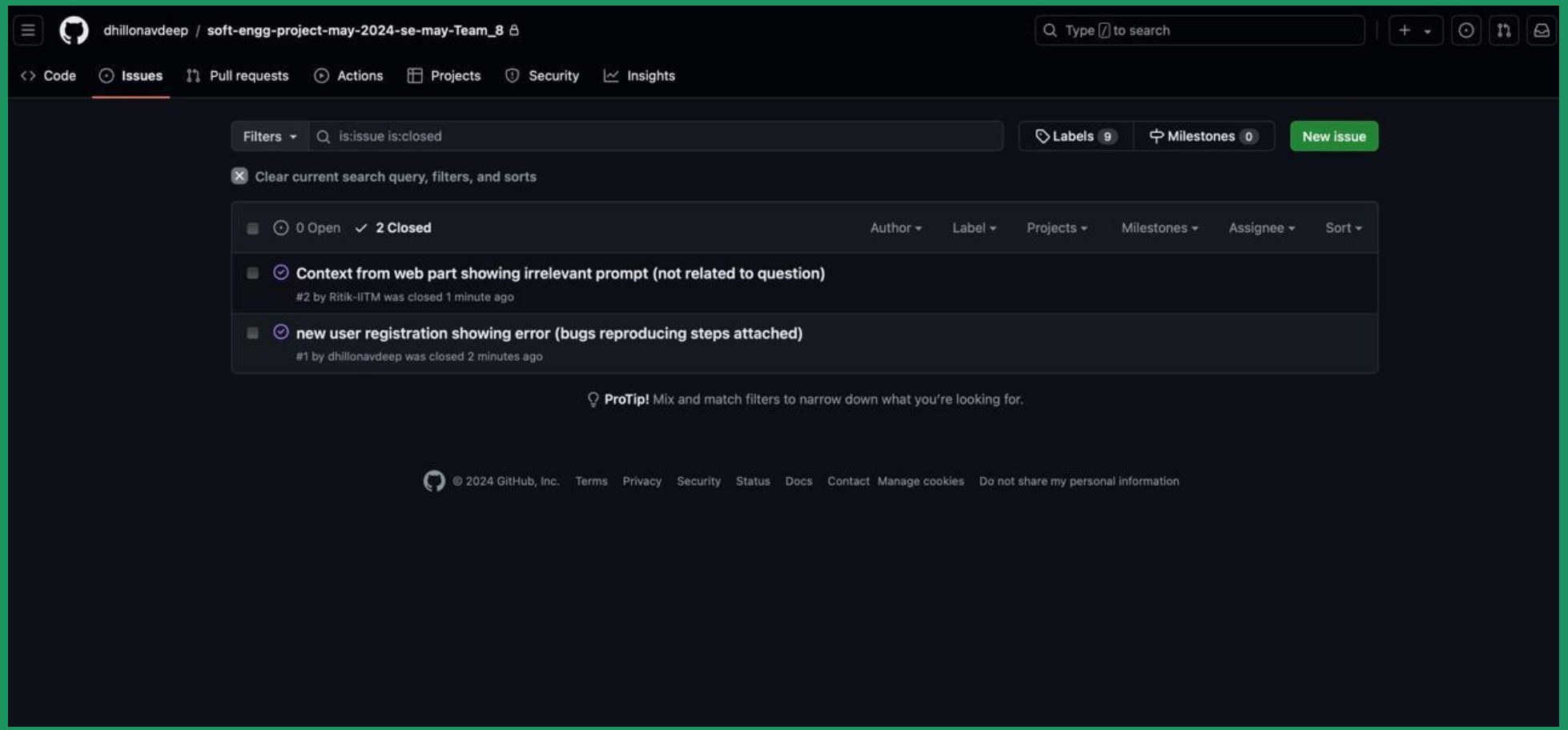
Development Sprint Reviews: We held weekly reviews with developers, the product manager, and the scrum master to assess progress, resolve issues, and plan the next sprint.

Code Demos and Reviews: Secondary developers and testers reviewed and integrated the code..

Code Review on Git: Pull Requests



Code Review on Git:
Issue Raised



Tools and Technologies Used

We used JIRA/ATLAS for project management, Kanban boards for task tracking, Git for version control, and IDEs for coding. API testing tools were employed for unit testing to ensure reliability.

Tools and Technologies Used

#	Tools/Technology	Usage
1	Adobe	Designing and editing content
2	Flask	Python web framework
3	Flask-SQLAlchemy	ORM integration
4	Flask-RESTful	API framework
5	Git/Git Hub	Version Control system
6	Google Docs	Collaborative Documents
7	Google Mail	Communication and sharing resources
8	Google Meet	Meetings,managing team
9	Google Sheet	Issue traking for end user testing
10	Google Slides	Used for making presentation
11	HTML/CSS,JS	Base technologies for web development
12	JIRA	PM tool for tracking tasks,issues etc
13	Kanban	A board system used for managing tasks
14	MS Powerpoint	Presentation and announcement slides
15	MS Word	Documentation and announcement
16	ollama	Language models
17	PyTest	python testing framework
18	SQLite	Application database
19	Vscode	IDE for developing code

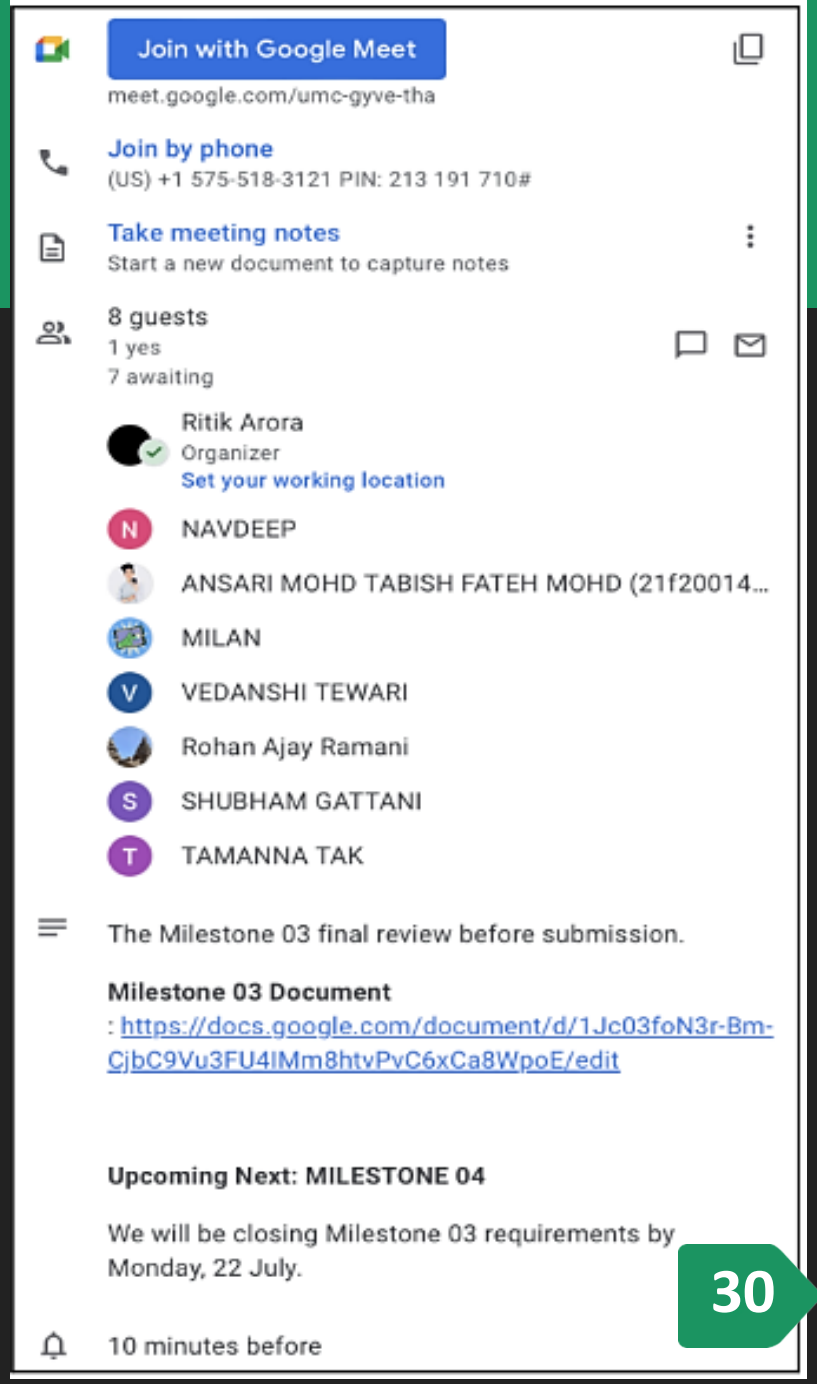
How we worked

Scrum meetings bring together the scrum master, product manager/owner, and scrum teams to exchange updates, address ongoing issues, review progress, gather feedback, and plan and assign upcoming tasks.

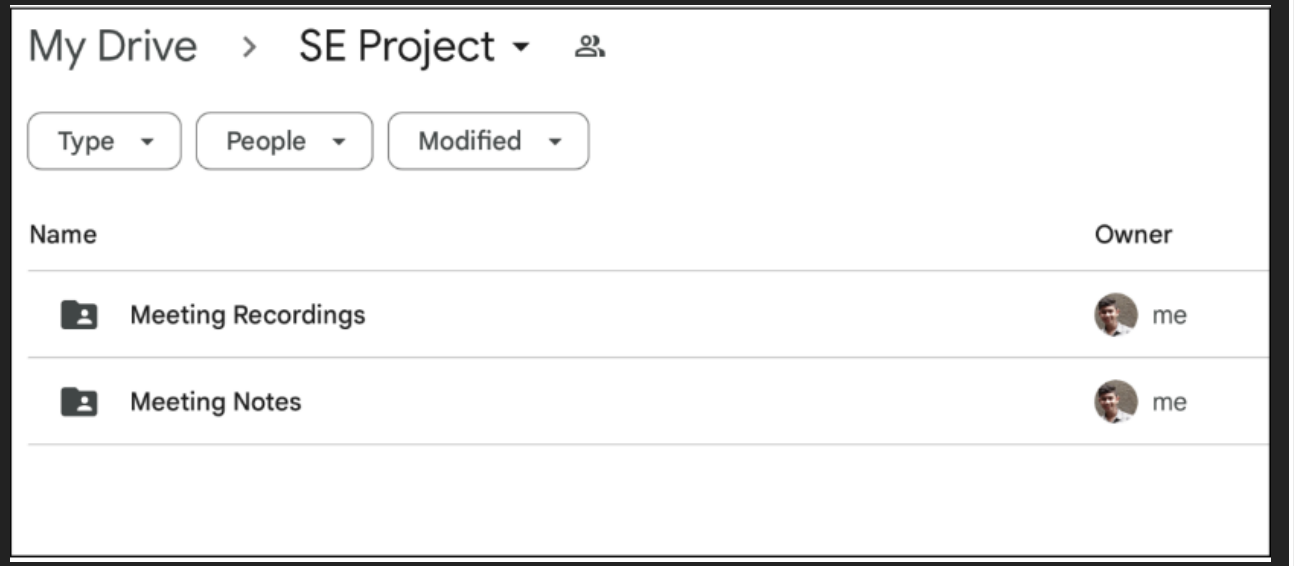
We've highlighted some key practices, tools, and deliverables that emerge from these meetings.

Meeting Planning

Step 1. Availability Calendar	Meetings scheduled based on availability. We have created an availability calendar to manage planned absences and common events (quizzes, festivals).
Step 2. Communication and Resource Sharing	Apart from our personal WhatsApp and Gmail accounts, we are also using a common WhatsApp Group to consolidate and streamline communications.
Step 3. Attendance Management	Attendance management and recording help track participation, maintain accountability, and ensure everyone stays informed by providing access to meeting records..
Step 4. Publish Team Calendar	The all-hands meet is scheduled two days in a week at 10:00 to 11:00PM. Scrum teams & milestone delivery teams schedule additional meets.
Step 5. Agenda & Meeting Report	The meeting invites include the agenda and prep work required. After meetings, a formal meeting report is published to all members.



Meeting Report



Meeting Notes

1. Review of last meeting and Milestone 1 story selection

The meeting started with a quick review of the last meeting and actions closed for Milestone 1. The team reviewed the stories from Milestone 1 for discussion and selection for further development. These stories were selected based on the fulfillment of project requirements, ability to demonstrate our understanding and capability to implement, and based on effective feasibility.

2. Shortlisted User Stories

Six stories were shortlisted. For each development story we have an assigned pair of developers from our scrum team based on personal availability and interest.

Date	4 July 2024 22:00 to 23:00
Agenda	Milestone 3 Discussion and Progress Updates
Meeting Link	https://meet.google.com/iob-jcai-yoe

Attendance

1	**Milan**	☑
2	**Ritik Arora**	☑
3	**Vedanshi Tiwari**	☑
4	**Tamanna Tak**	☑
5	**Navdeep**	☑
6	**Shubham Gattani**	☑
7	**Rohan Ajay Ramani**	☑
8	**Tabish**	☑

Table of Content

Thank You