Software Engineering Project (BSCS3001)

Milestone 3 – Scheduling and Design

Submitted by **Team 15**

Niharika Girdhar (21f1001715)

Zahabiyah Ghadiali (21f1004235)

Anirudha Anekal (21f1005685)

Ganesh P (21f1006171)

Laxman Bafna (21f2000485)

Pushpak Ruhil (21f2001180)

Lalit Kumar (21f3001013)

Archit Handa (22f2000744)

IITM BS Degree Program,

Indian Institute of Technology, Madras, Chennai,

Tamil Nadu, India - 600036

Contents

1 Project Schedule

This section showcases the schedule of the overall project, highlighting:

- Schedule of our **sprints/iterations**
- Timings of our **scrum** and **standup** meetings
- Distribution of the **tasks**
- Project Management/Scheduling **tools** being employed

1.1 Sprint Schedule

Below table describes how we have set up our sprints. In order to stay consistent with the coursework, we have aligned our sprints to match the deadlines for the respective 'Project Milestone'. This not only allows us to stay time-bound but also promotes a sense of responsibility among the team members towards the shared objective: *'Seek++'*, the software solution being developed as a part of our SE Project.

Scrum Iterations/Sprints	Start Date	End Date	Status	Notes
Milestone 1	Jun 6, 2024	Jun 17, 2024	Completed ▾	Identify User Requirements
Milestone 2	Jun 18, 2024	Jun 30, 2024	Completed ▾	User Interfaces
Milestone 3	Jul 1, 2024	Jul 12, 2024	Completed ▾	Scheduling and Design
Milestone 4	Jul 13, 2024	Jul 28, 2024	In progress ▾	API Endpoints
Milestone 5	Jul 29, 2024	Aug 7, 2024	Not started ▾	Test Cases, Test Suite of the Project
Milestone 6	Aug 8, 2024	Aug 18, 2024	Not started ▾	Final Submission

***Table 1.1.1:** Scrum Schedule*

1.2 Scrum Meetings

We hold daily discussions about our work in a dedicated WhatsApp group that we have created for this purpose and to seek quick assistance from team mates. To clarify any project-related doubts, we raise the issue in the SE Google Space where the Instructors and TAs can provide us with solutions.

Moreover, Stand-up meetings are scheduled twice a week, on Mondays and Fridays, typically at 10:00 PM, though we occasionally change the timings to accommodate each team member’s comfort and preferences. Usually, during the Monday meetings, we delegate tasks to each member for the week and on Friday meetings, we monitor the progress and make suitable adjustments to plan better for the following weeks and sprints.

Links to Minutes of the Meeting (MoM) for some of our Scrum Meetings:

- Jun 9, 2024 | SE Project Discussion
    - Icebreaker between team members
    - Understood the requirements of the SE Project
    - Identified the tools and tech stack for the project
- Jun 14, 2024 | Standup - SE Project
    - Explored the Learner Journey Map
    - Identified key pain points to further develop user stories till next meeting
- Jun 17, 2024 | Standup - SE Project
    - Collaborated and compiled the list of user stories
    - Categorised them as doable (easy, medium, hard) based on the SMART guidelines to choose the final set of user stories
    - Delegated tasks to complete report for milestone 1 and create wireframes and storyboards for milestone 2

1.3 Task Distribution

Based on the user stories identified in previous milestones and requirements of the following ones, we have come up with the following list of tasks to be completed to bring this SE Project to fruition. Moreover, we have tried to balance out the tasks among each team member leveraging her/his skillset to maintain a collaborative environment and adhere to the sprint timelines.

Milestone	Sub-tasks	Sprint	Assigned to
Milestone 1: Requirements Analysis	Creating Learner Journey Map	1	Archit, Ganesh, Lalit
	Exploring GenAI Functionalities	1	Pushpak, Anirudha, Lalit
	Creating User Stories	1	All
	Preparing Report	1	All
Milestone 2: User Interfaces	Creating Storyboards	2	Niharika, Pushpak, Lalit
	Creating Wireframes	2	Zahabiyah, Archit
	Preparing Report	2	Pushpak
Milestone 3: Scheduling and Design	Scheduling: Tasks, Gantt Chart, Scrum	3	Niharika, Pushpak, Laxman, Zahabiyah
	Design of Components	3	Archit, Ganesh
	Class Diagrams	3	Archit, Ganesh, Anirudha
	Preparing Report	3	Archit
Milestone 4: REST API Endpoints	API Design	4	Anirudha, Archit
	API Description	4	Anirudha, Lalit
	YAML File	4	Zahabiyah, Niharika, Laxman
Milestone 5: Software Testing	Identify and design test scenarios	5	All
	Design test cases and analyse results	5	Anirudha, Zahabiyah
	Preparing report	5	Anirudha, Zahabiyah
Milestone 6: Final Submission	frontend implementation	6	Laxman, Ganesh, Lalit, Zahabiyah
	backend Implementation	6	Anirudha, Archit, Pushpak, Niharika
	Preparing final report	6	All
	Preparing project presentation	6	All

Table 1.3.1: *Task Distribution*

1.4 Project Management Tools

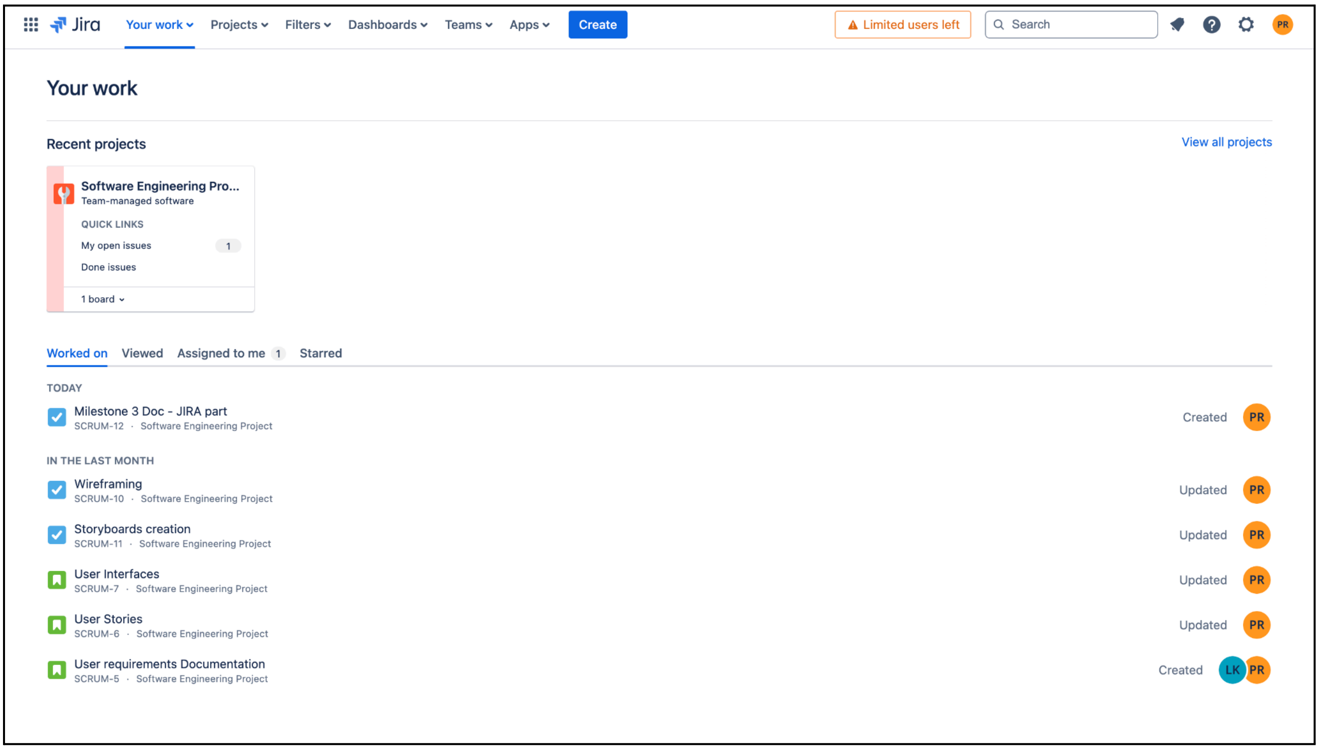
We will be using JIRA as our project management tool (Link to our JIRA platform) and GitHub (Link to our GitHub Repo) for version control and coding collaboration.

Utilising such project management tools help us to:

- Keep track of the sprints to monitor overall progress of the project
- Create issues and raise tickets for each member’s tasks
- Collaborate on code to keep track of the commits on GitHub

In fact, we have integrated our JIRA with GitHub, allowing us to fetch data on code commits, pull requests, and more.

Below are a few screenshots showcasing the JIRA platform used by our team, demonstrating the proof of concept:



Your work

Recent projects

View all projects

Software Engineering Pro...
Team-managed software

QUICK LINKS
My open issues 1
Done issues

1 board

Worked on · Viewed · Assigned to me 1 · Starred

TODAY

Milestone 3 Doc - JIRA part
SCRUM-12 · Software Engineering Project — Created PR

IN THE LAST MONTH

Wireframing
SCRUM-10 · Software Engineering Project — Updated PR

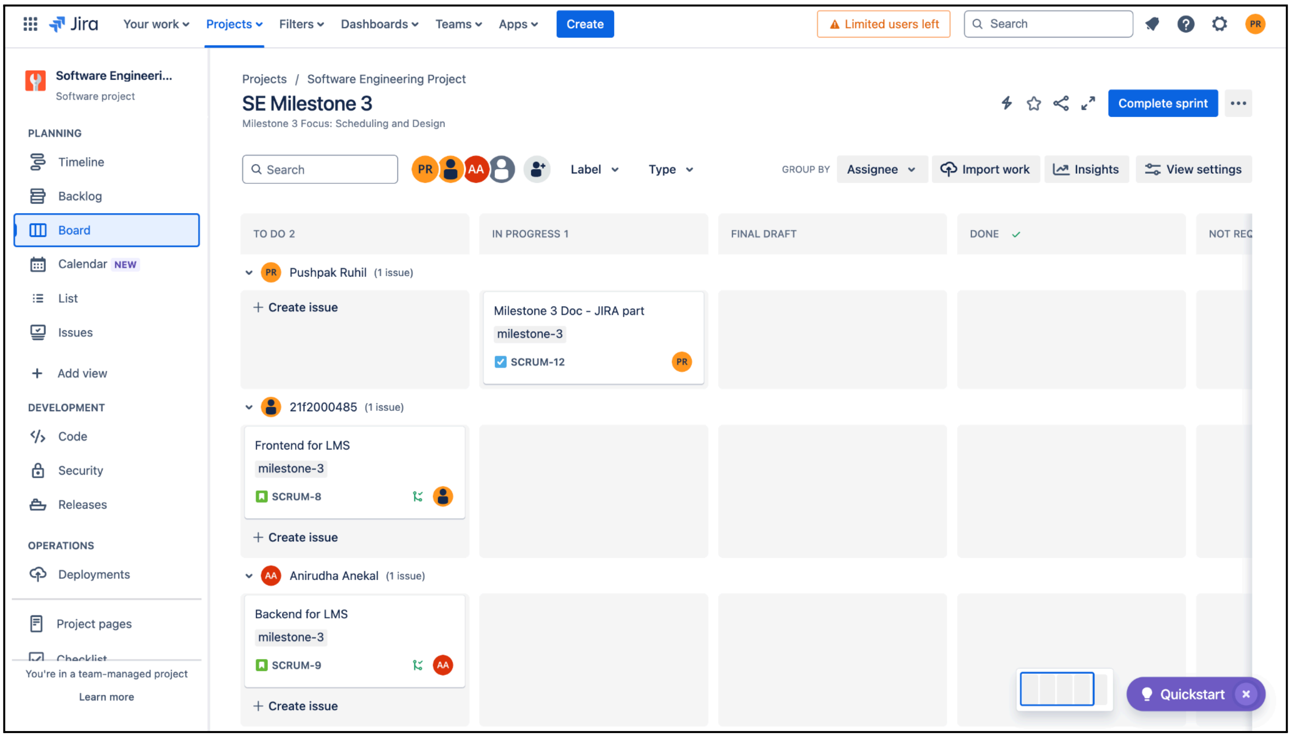
Storyboards creation
SCRUM-11 · Software Engineering Project — Updated PR

User Interfaces
SCRUM-7 · Software Engineering Project — Updated PR

User Stories
SCRUM-6 · Software Engineering Project — Updated PR

User requirements Documentation
SCRUM-5 · Software Engineering Project — Created LK PR

Figure 1.4.1: *JIRA: Home Page for Members*



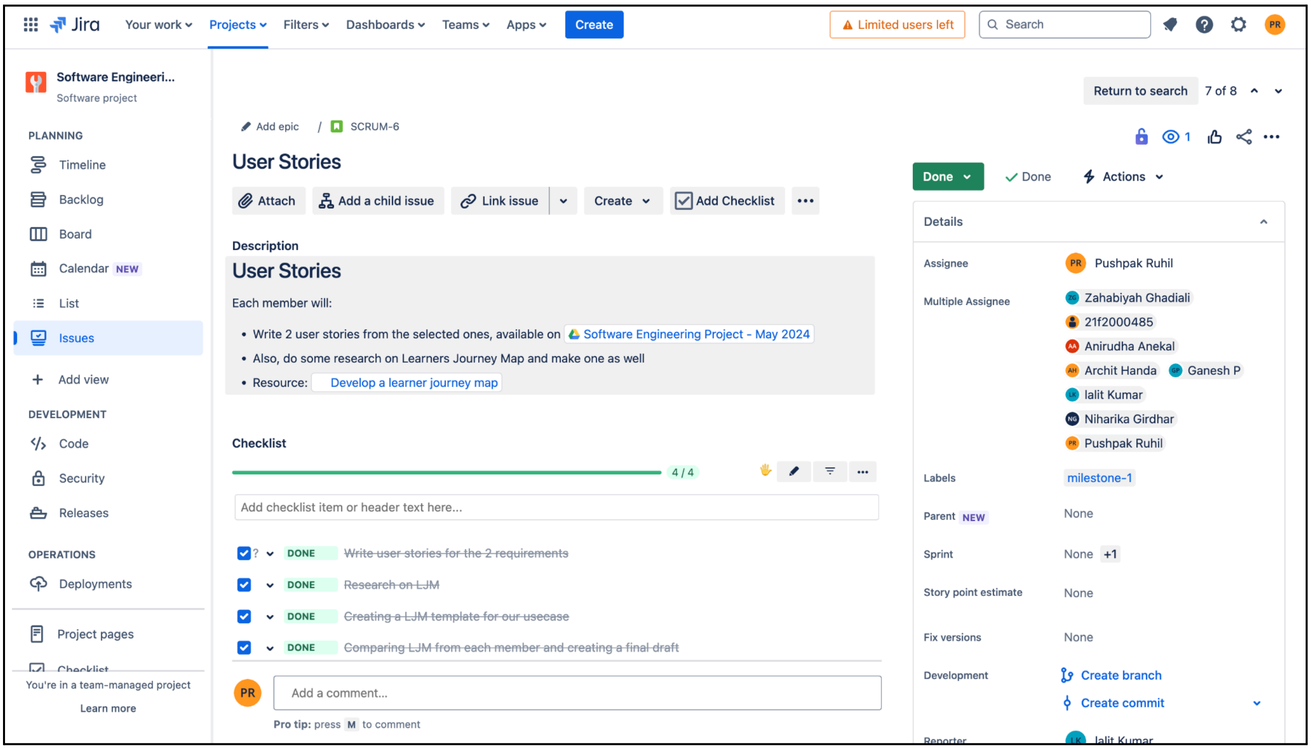
Projects / Software Engineering Project

SE Milestone 3

Milestone 3 Focus: Scheduling and Design

TO DO 2	IN PROGRESS 1	FINAL DRAFT	DONE ✓	NOT REC
PR Pushpak Ruhil (1 issue)				
+ Create issue	Milestone 3 Doc - JIRA part milestone-3 ☑ SCRUM-12 PR			
21f2000485 (1 issue)				
Frontend for LMS milestone-3 SCRUM-8 + Create issue				
AA Anirudha Anekal (1 issue)				
Backend for LMS milestone-3 SCRUM-9 AA + Create issue				

Figure 1.4.2: *JIRA: Current Sprint Board*



Add epic / SCRUM-6

User Stories

Done ✓ Done

Description

User Stories

Each member will:

- Write 2 user stories from the selected ones, available on Software Engineering Project - May 2024
- Also, do some research on Learners Journey Map and make one as well
- Resource: Develop a learner journey map

Checklist

4 / 4

Add checklist item or header text here...

☑ ? DONE ~~Write user stories for the 2 requirements~~

☑ DONE ~~Research on LJM~~

☑ DONE ~~Creating a LJM template for our usecase~~

☑ DONE ~~Comparing LJM from each member and creating a final draft~~

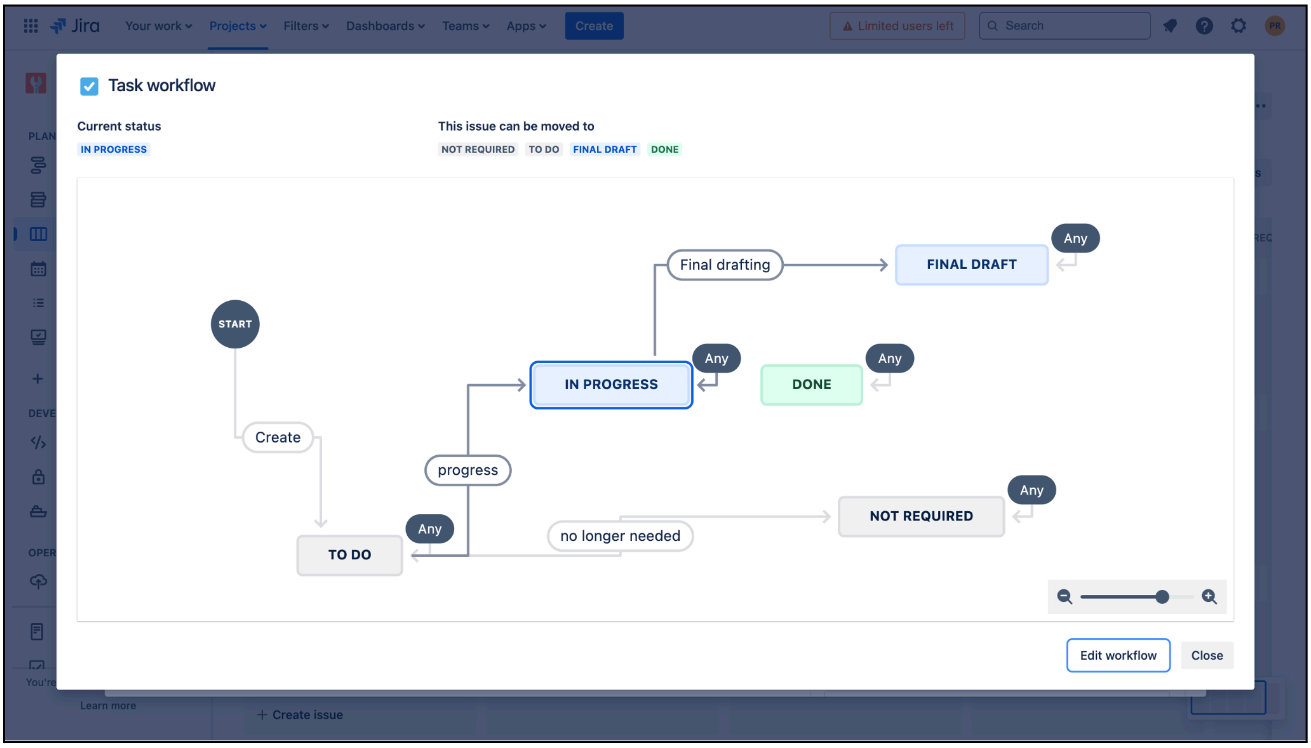
PR Add a comment...

Pro tip: press M to comment

Details

Assignee	PR Pushpak Ruhil
Multiple Assignee	Zahabiyah Ghadiali 21f2000485 Anirudha Anekal Archit Handa · Ganesh P lalit Kumar Niharika Girdhar Pushpak Ruhil
Labels	milestone-1
Parent NEW	None
Sprint	None +1
Story point estimate	None
Fix versions	None
Development	Create branch Create commit
Reporter	lalit Kumar

Figure 1.4.3: *JIRA: Issue Ticket Created*



Task workflow

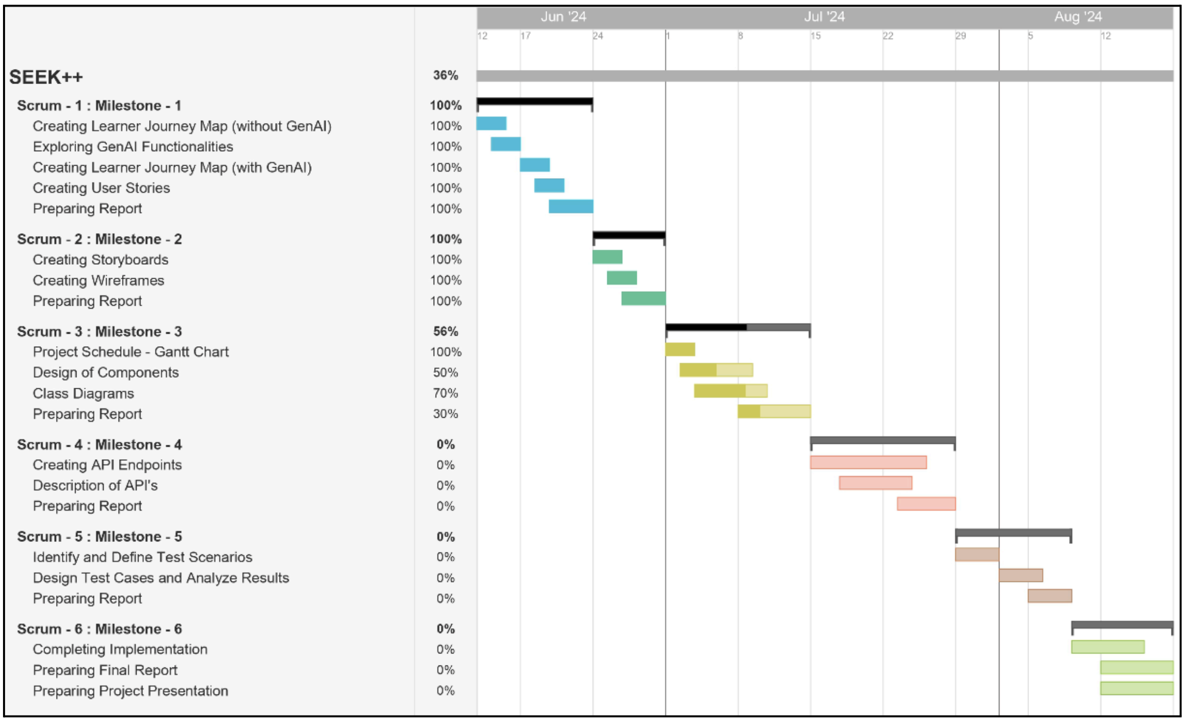
Current status
IN PROGRESS

This issue can be moved to
NOT REQUIRED TO DO FINAL DRAFT DONE

Figure 1.4.4: JIRA: Workflow for Epics, Stories, Tasks and Subtasks

1.5 Gantt Chart

Below is the Gantt Chart highlighting the schedule for each task in the grand scheme of the entire SE Project: '*Seek++*' LMS software solution.



SEEK++	**36%**
Scrum - 1 : Milestone - 1	**100%**
Creating Learner Journey Map (without GenAI)	100%
Exploring GenAI Functionalities	100%
Creating Learner Journey Map (with GenAI)	100%
Creating User Stories	100%
Preparing Report	100%
Scrum - 2 : Milestone - 2	**100%**
Creating Storyboards	100%
Creating Wireframes	100%
Preparing Report	100%
Scrum - 3 : Milestone - 3	**56%**
Project Schedule - Gantt Chart	100%
Design of Components	50%
Class Diagrams	70%
Preparing Report	30%
Scrum - 4 : Milestone - 4	**0%**
Creating API Endpoints	0%
Description of API's	0%
Preparing Report	0%
Scrum - 5 : Milestone - 5	**0%**
Identify and Define Test Scenarios	0%
Design Test Cases and Analyze Results	0%
Preparing Report	0%
Scrum - 6 : Milestone - 6	**0%**
Completing Implementation	0%
Preparing Final Report	0%
Preparing Project Presentation	0%

Figure 1.5.1: Gantt Chart

2 Design Components

We plan to develop four main components based on the user stories highlighted in milestones 1 and 2. These are User, Course, Assessment Management, and GenAI. Following are the details for each individual component.

2.1 User Component

User Component houses classes that will manage the onboarding of the student and/or the instructor onto the 'Seek++' portal. It will allow users to access other components based on the roles and respective permissions granted post-authentication.

Following are the major classes in this component:

1. **User**: Maintains the login credentials for a user
2. **Student**: Maintains the list of enrolled courses that the student can access
3. **Instructor**: Maintains the list of courses taught by the instructor along with the ability to create and check graded assignments

2.2 Course Component

Course Component houses classes that will allow instructors to manage the content for courses and develop specific modules, lessons, and assignments. The students will be later accessing these resources as a part of their learner journey.

Following are the major classes in this component:

1. **Course**: Maintains details for a course as well as a list of modules that will be taught throughout the duration of the course; ***e.g. BSCS3001 - Software Engineering***
2. **Module**: Maintains the list of lessons that have been developed for each concept of the course along with the assignments for the same; ***e.g. Week 1 - Software Development Lifecycle***
3. **Lesson**: Maintains the video lectures and slide decks that the instructor has prepared for each sub-concept; ***e.g. Lec 1.1 - Deconstructing the Software Development Process***

2.3 Assessment Management Component

Assessment Management Component houses classes that will allow instructors to create common graded and practice assignments. Moreover, it will also allow the students to not only make submissions but also raise requests to generate more assignments on demand.

Following are the major classes in this component:

1. **Assignment**: Maintains details for the assignment that the instructor has created and allows her/him to grade the submissions

2. **Submission**: Maintains the submission attempts a student makes to complete the assignments as well as the grades (s)he receives after deadline

2.4 GenAI Component

GenAI Component is the main powerhouse of 'Seek++'. It houses classes that will allow integration of Generative AI into the conventional LMS software. We have designed it to encapsulate all the necessary sub-components that serve as a solution to the problems identified in earlier milestones and will eventually aid the student in their learner journey.

Following are the major classes in this component:

1. **AssignmentGenerator**: Handles the on-demand request to generate practice assignments both theoretical and programming; it will help design the questions, provide solutions, and give tips to students to understand which concepts are to be applied in what situation; it further encompasses the **PAGenerator** and **PrPAGenerator** subclasses that allow developing theoretical and programming practice assignments respectively

2. **CodeHelp**: Specifically designed to provide students with hints and code-snippets that will help them to think about the problem in a more directed and focussed manner while attempting the coding assignments

3. **SpeechToText/SpeechToCode**: Special classes that promote inclusive education by allowing students with limited motor skills to attempt the assignments and code with ease

4. **TestCaseGenerator**: Helps instructors to create custom test cases for the programming assignments based on the public and private test requirements

5. **Summarizer**: Allows students to summarize the contents for a lesson (video and/or slide deck) in a concise format making it extremely beneficial for quick revision while negating the need for taking manual notes; encompasses subclasses **VideoSummarizer** and **SlideSummarizer** to handle the specialised cases separately

6. **ChatBot**: Instantly answers quick questions about course content, assignment deadlines, exam dates and more as per the information fed by the instructor eliminating the need to post simple query on the discussion forums and await its delayed reply

7. **Translator**: Converts the summaries as well as the ChatBot's response into the regional language of the student for her/him to better understand and connect with the concepts

8. **PainPointDetector**: Identifies the weak learning points and gap areas of the student based on her/his performance trend and further devises and suggest a study/revision plan to cover the identified gaps in a more planned and effective manner

3 Class Diagram

As per the components and classes discussed in the Section 2, we have come up with the following UML Class Diagram. This helps us visualize the hierarchy and connections between each individual component better. Moreover, while developing, we can look back at this diagram whenever stuck to check whether we have developed the required components and connections or not.

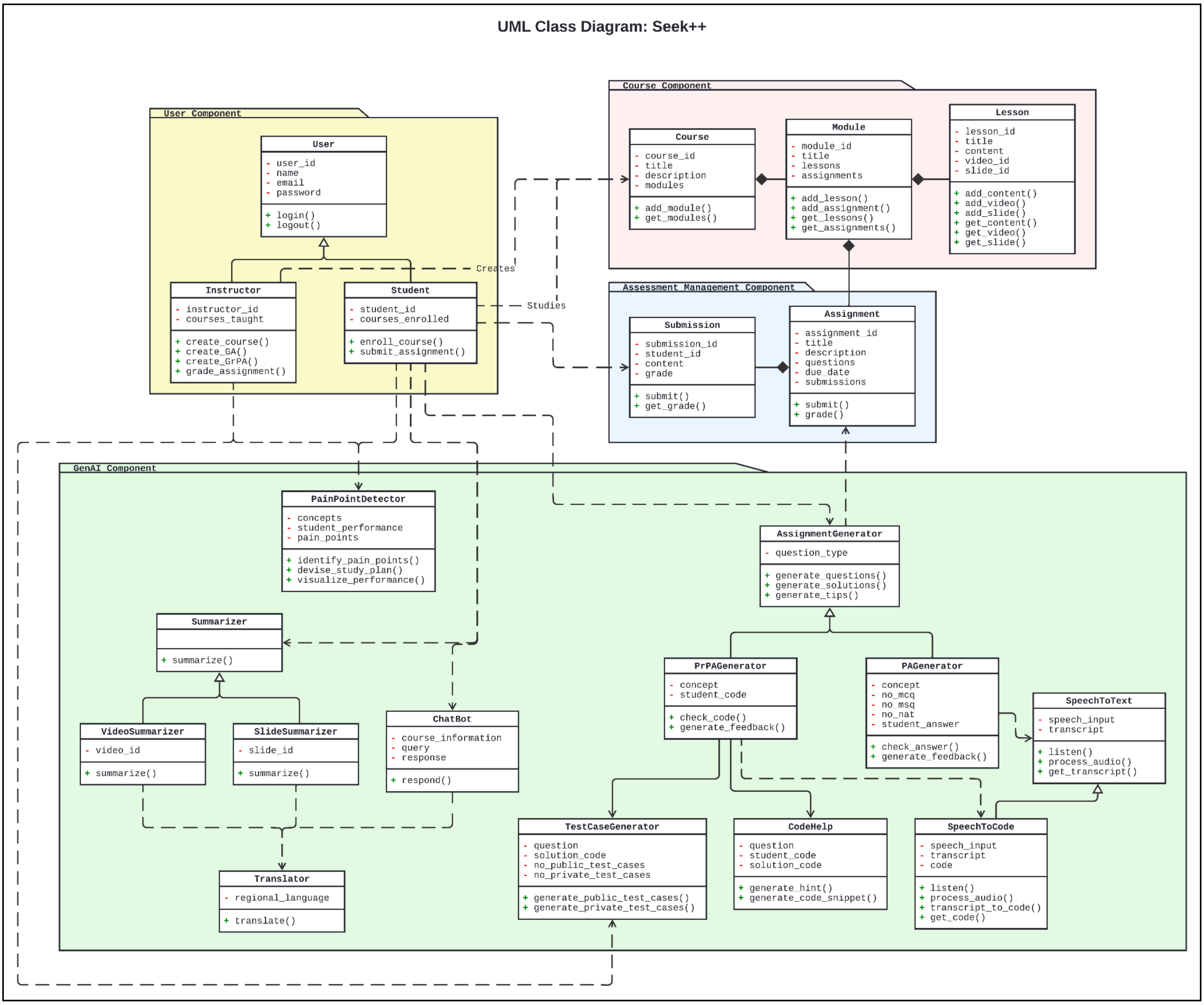


Figure 3.1: *UML Class Diagram for 'Seek++'*