Software Engineering Project Report

Sept 2023 Group 02

Honor Codes

I Ankit Kumar Rath with roll no. 21F1003914 declare that I will not use any ideas, writings, code or work that is not my own or my group's with the intention of claiming it my or my group's work. For all the work that I will submit as part of this project, I will not share it outside my group with anybody directly or indirectly or upload it to any of the public forums on the internet.

I acknowledge that failing in any of the above constitutes plagiarism and in that case, the institute will take appropriate disciplinary action.

Sign: Ankit Rath

Date: 13-10-2023

I Chaitanya Kumaria with roll no. 21f1000479 declare that I will not use any ideas, writings, code or work that is not my own or my group's with the intention of claiming it my or my group's work. For all the work that I will submit as part of this project, I will not share it outside my group with anybody directly or indirectly or upload it to any of the public forums on the internet.

I acknowledge that failing in any of the above constitutes plagiarism and in that case, the institute will take appropriate disciplinary action.

Sign: Chaitanya Kumaria

Date: 13-10-2023

I Om Sharma with roll no. 21f1004424 declare that I will not use any ideas, writings, code or work that is not my own or my group's with the intention of claiming it my or my group's work. For all the work that I will submit as part of this project, I will not share it outside my group with anybody directly or indirectly or upload it to any of the public forums on the internet.

I acknowledge that failing in any of the above constitutes plagiarism and in that case, the institute will take appropriate disciplinary action.

Sign: Om Sharma
Date: 13-10-2023

I Yatin Chug with roll no. 21f1007066 declare that I will not use any ideas, writings, code or work that is not my own or my group's with the intention of claiming it my or my group's work. For all the work that I will submit as part of this project, I will not share it outside my group with anybody directly or indirectly or upload it to any of the public forums on the internet.

I acknowledge that failing in any of the above constitutes plagiarism and in that case, the institute will take appropriate disciplinary action.

Sign: Yatin
Date: 13-10-2023

I Aayush Mahapatra with roll no. 21f2000846 declare that I will not use any ideas, writings, code or work that is not my own or my group's with the intention of claiming it my or my group's work. For all the work that I will submit as part of this project, I will not share it outside my group with anybody directly or indirectly or upload it to any of the public forums on the internet.

I acknowledge that failing in any of the above constitutes plagiarism and in that case, the institute will take appropriate disciplinary action.

Sign: Ayush

Date: 13-10-2023

IITM BS Degree Learning Path Recommendation System

A learning path recommendation based on both learning profile and feedback from previous term students can be a valuable tool for students. By taking into account student data from past enrollments, student performance and interests, as well as the feedback of other students who have taken similar courses, a learning path recommendation can help students identify the courses that are most likely to be beneficial to them. Such learning path recommendations can help students stay on track and make progress towards their educational goals. By providing students with a clear roadmap of the courses that they need to take, a learning path recommendation can help students pace themselves and avoid getting lost or sidetracked.

Here are some of the factors that should be considered when making a learning path recommendation:

Enrollment data from previous terms

The student's learning profile, including their past performance, interests, and goals.

The feedback of other students who have taken similar courses.

The student's schedule and other commitments.

The system should ideally have two users - an admin, and a student. An admin can load enrollment data from previous terms. The system should be able to infer patterns and provide recommendations to students. Students can also provide their feedback about past courses. The student can provide their learning profile, interests, goals, schedules and commitments, and the system should provide appropriate recommendations based on all the above inputs.

You are free to think creatively and come up with additional requirements or modify some of these requirements, as long as the overarching purpose of a learning path recommendation system is fulfilled.

Milestone 1- Identify User Requirements

During milestone 1, we thoroughly examined the project requirements and formulated precise, succinct agile-style requirements presented as User Stories. We developed user stories catering to Primary, Secondary, and Tertiary users, ensuring adherence to the SMART principle - Specific, Measurable, Achievable, Relevant, and Time-bound across all narratives.

Primary Users

- **Students**: Users who will receive learning path recommendations based on their profiles, interests, goals, and feedback.

Secondary Users

- **Admins:** Users responsible for managing and inputting data into the system, such as the enrollment data from previous terms.

Tertiary Users

- **Instructors:** Users who will indirectly benefit from the system's recommendations, leading to an enhanced classroom experience.
- **Planning Teams:** Planning Team can take estimates from the Recommender Systems to allocate resources (assuming students most likely take the Suggestions provided from knowledge of his/her learning path)

User Stories

As a student

1. I want to log in to the system using my student email address, so that I can access my personalized learning recommendations and account.
2. I want to track my educational progress and achievements throughout my learning path, so that I can stay motivated and have a clear understanding of my academic journey.
3. I want to register for the upcoming term through the Course Registration page, so that I can view all available courses and plan my academic term.
4. I want to provide details about my learning profile, including the number of hours per day I can dedicate and the number of subjects I can complete per week, as well as my interests, goals, schedules, and commitments. This information will help the system recommend courses that align with my academic objectives and availability.
5. I want to receive personalized learning path recommendations based on my past performance and feedback from other students, so that I can make informed decisions about my course selections, leading to a more fulfilling learning experience.
6. I want to be able to provide feedback about the courses I have taken, so that the system can incorporate my input "anonymously" and continually improve its recommendations for future students.
7. I want to receive notifications or reminders for important academic deadlines, such as assignment due dates, so that I can stay organized and avoid missing important events.
8. I want to view a user-friendly dashboard where I can see the recommendations generated by the learning system and an overview of my academic performance so far.

As an admin

1. I want to access the system using my institute email address, so that I can access my account.
2. I want to upload enrollment data from previous terms, including information about courses, enrollment numbers, and student performance data, so that the system can

analyze historical patterns and provide more accurate learning path recommendations to students.

3. I want to view and analyze overall student performance data, so that I can identify trends and make data-driven decisions to enhance the effectiveness of the learning path recommendation system.
4. I want to be able to update course information and curriculum details, so that the recommendations are always based on the most current and relevant data.

As an instructor

1. I want to view anonymous feedback from students about past courses, so that I can understand areas of improvement and adjust my teaching methods accordingly.
2. I want access to aggregated data on student preferences and performance, so that I can tailor my courses to align with student interests and enhance overall engagement and learning outcomes.
3. I want to receive automated alerts for students who are struggling academically so that I can provide appropriate support.

As a Planning team

1. I want to use course recommendations to estimate organizational/support team requirements, so that we can plan ahead to make students learn better with adequate resources.

Milestone 2 - User Interfaces:

In milestone 2, we leveraged the user stories to generate storyboards and abstract wireframes using Figme.. Storyboards serve as a holistic depiction of the project, facilitating Agile software development by furnishing developers with a lucid comprehension of pending tasks. Crafted using Exacildraw, our storyboard effectively communicates the fundamental functionality of the project in an engaging and interactive fashion.

🔗 Figma

🔗 Storyboard

Milestone 3 - Scheduling and Design:

1. Project Scheduling Overview

Project scheduling is an important aspect of a software development project. Pivotal Tracker was used as the management tool for managing the project, assigning tasks and scheduling. A team meeting was held to plan and strategize the project roadmap and each member's responsibilities.

Agile Framework

This section explains the fundamental components of the agile framework that we have adopted for our project. We cover:

- **Sprint Length**: We use a one-week sprint length.
- **Multiple Feature Requests in a Sprint**: Our sprints can include multiple feature requests depending on the request complexity.
- **Labels**: Sprint-number, design, frontend, and backend.
- **Checklists**: Checklists are used to track and manage tasks within feature requests.

Sprint Management

This section delves into how we manage sprints within the project. Key points include:

- **Sprint Planning**: Eacn sprint kicks off with an explanation of the sprint requirements.
- **Team Involvement**: The team collaboratively selects and prioritizes feature requests.
- **Sprint Length Significance**: 1 week sprints allow us to have time bound iterations and a week-wise roadmap for the entire software development phase. It can accommodate multiple feature requests depending on their complexity.

Feature Requests

We describe how user stories and features are systematically converted into feature requests:

- **Identification**: User stories are converted to feature requests.

- **Definition**: Defining and breaking down features into actionable tasks.
- **Checklists**: To track progress within feature requests.
- **Activity:** To reflect progress, team members must share the progress in the feature request with relevant content and resources.
- **Status:** Each feature request has multiple status like unstarted, in progress, finished, accepted and rejected. It is important to keep them updated as per the current state of the progress.
- **Class Diagram:** A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

Labels and Categorization

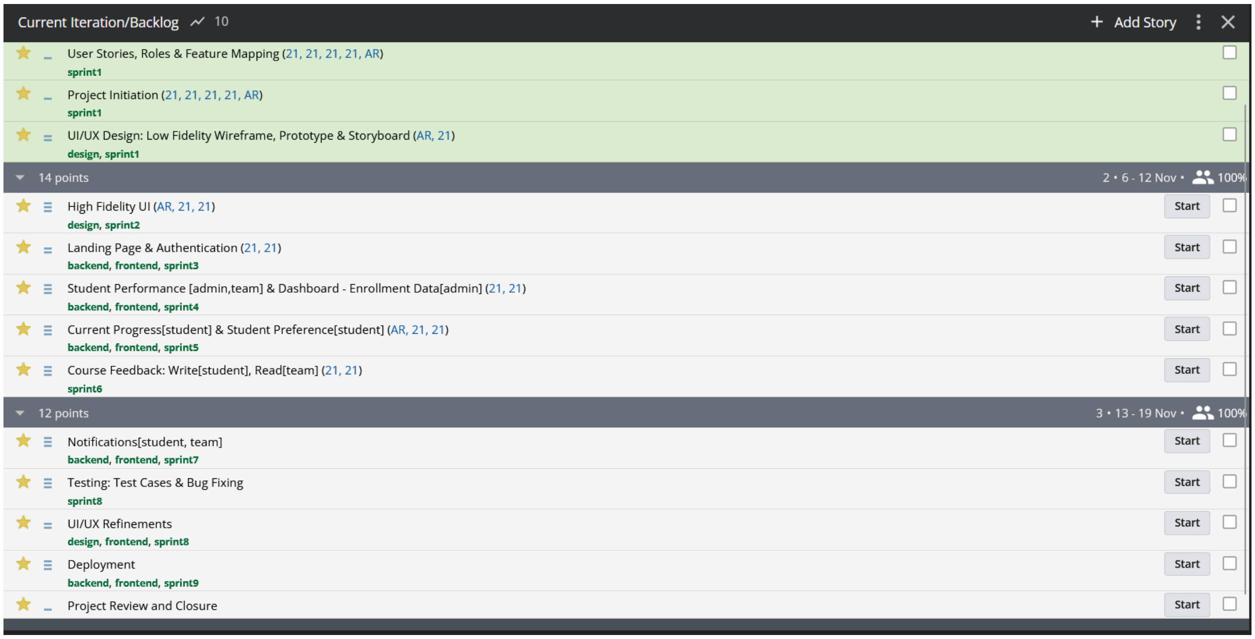
This section outlines our usage of labels in the project:

- **Sprint-number**: Helps in keeping track of the feature requests in a sprint wise manner.
- **Design, Frontend, Backend**: Three categories that define the type of feature request.

Pivotal Tracker

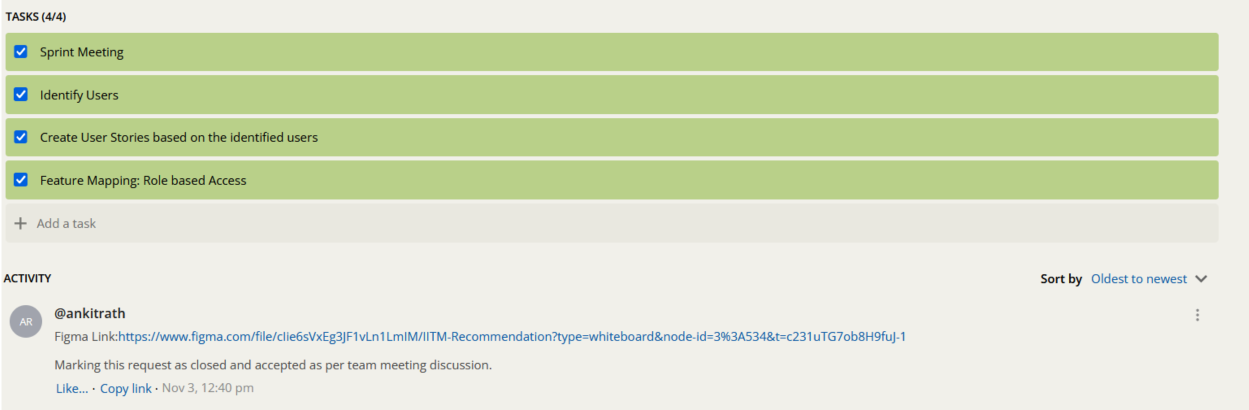
- The Project Manager has created feature requests from the user stories & requirements.
- Feature Requests have been defined and scheduled in a sprint wise manner.
- They are further categorized into design/frontend/backend tasks.
- Each feature request has been assigned a member by the Project Manager.
- Sprints kick off with team meetings and end with team review.
- Work progress gets mapped in the Activity section where members post their updates.
- Upon completion of sprint and team review, if all members approve the work, the feature request gets moved to accepted.

Below are some screenshots of our Project Management tool.



Current Iteration/Backlog 10 + Add Story

User Stories, Roles & Feature Mapping (21, 21, 21, 21, AR) sprint1	□
Project Initiation (21, 21, 21, 21, AR) sprint1	□
UI/UX Design: Low Fidelity Wireframe, Prototype & Storyboard (AR, 21) design, sprint1	□
14 points	2 • 6 - 12 Nov • 100%
High Fidelity UI (AR, 21, 21) design, sprint2	Start □
Landing Page & Authentication (21, 21) backend, frontend, sprint3	Start □
Student Performance [admin,team] & Dashboard - Enrollment Data[admin] (21, 21) backend, frontend, sprint4	Start □
Current Progress[student] & Student Preference[student] (AR, 21, 21) backend, frontend, sprint5	Start □
Course Feedback: Write[student], Read[team] (21, 21) sprint6	Start □
12 points	3 • 13 - 19 Nov • 100%
Notifications[student, team] backend, frontend, sprint7	Start □
Testing: Test Cases & Bug Fixing sprint8	Start □
UI/UX Refinements design, frontend, sprint8	Start □
Deployment backend, frontend, sprint9	Start □
Project Review and Closure	Start □



TASKS (4/4)

- ☑ Sprint Meeting
- ☑ Identify Users
- ☑ Create User Stories based on the identified users
- ☑ Feature Mapping: Role based Access

+ Add a task

ACTIVITY

Sort by Oldest to newest

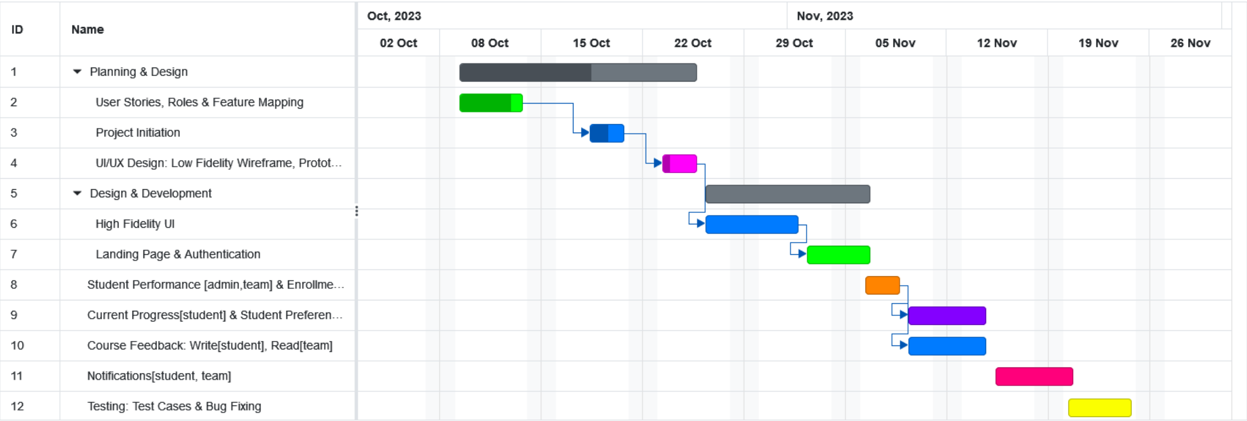
AR @ankitrath

Figma Link:https://www.figma.com/file/clie6sVxEg3JF1vLn1LmIM/IITM-Recommendation?type=whiteboard&node-id=3%3A534&t=c231uTG7ob8H9fuJ-1

Marking this request as closed and accepted as per team meeting discussion.

Like... · Copy link · Nov 3, 12:40 pm

Gantt Chart:



ID	Name
1	Planning & Design
2	User Stories, Roles & Feature Mapping
3	Project Initiation
4	UI/UX Design: Low Fidelity Wireframe, Protot...
5	Design & Development
6	High Fidelity UI
7	Landing Page & Authentication
8	Student Performance [admin,team] & Enrollme...
9	Current Progress[student] & Student Preferen...
10	Course Feedback: Write[student], Read[team]
11	Notifications[student, team]
12	Testing: Test Cases & Bug Fixing

2. Meeting Minutes

User Stories, Roles & Feature Mapping

- **Date:** 11-10-2023
- **Meeting Duration:** 30 minutes

Meeting Details/Minutes:

Agenda:

- Review the user stories provided.
- Discuss the roles and responsibilities within the team.
- Map the features to the user roles.

Meeting Summary:

1. The meeting began with a discussion of the user stories provided. Each user story was reviewed, and the team had a clear understanding of the project's primary and secondary users and their requirements.
2. Roles and responsibilities were discussed. The team identified the roles of developers, UI/UX designers and project manager. Responsibilities for each role were clarified.
    - Ankit Kumar Rath: Design, Project Manager
    - Yatin Chug: Developer, Architect
    - Chaitanya Kumaria: Quality Assurance Tester, Developer
    - Om Sharma: Developer, Technical Writer
    - Aayush Mahapatra: Developer, Architect
3. Feature mapping was done based on user roles. The following features were mapped:
    - Authentication: Required for all user roles.
    - Landing Page: Accessible by all user roles.
    - Dashboard (Current Progress): Available to all users.
    - Dashboard (Course Registration): Accessible by all users.
    - Dashboard (Course Feedback): Write access for primary and secondary users, read access for tertiary users.
    - Dashboard (Notifications): Available to all users.
    - Dashboard (Enrollment Data): Accessible by all users.
    - Dashboard (Student Performance): Read access for primary and secondary users.
    - Dashboard (Student Preference): Read access for primary and secondary users.
4. The project manager highlighted the importance of aligning the upcoming sprints with these user roles and features.

Action Items:

- Developers prepare for Sprint 2 based on feature mapping.
- UI/UX designer to start working on low-fidelity wireframes.

Project Initiation - Setup, Assigning Responsibilities

- **Date:** 18-10-2023
- **Meeting Duration:** 30 minutes

Meeting Details/Minutes:

Agenda:

- Discuss the progress of project initiation.
- Assign responsibilities for the setup phase.

Meeting Summary:

1. The meeting began with a review of the progress made during the project initiation phase. The team discussed the completion of the project charter, roles, and responsibilities matrix.
2. The project setup was discussed, including the establishment of project management tools, such as Trello, and the setup of the development environment.
3. Roles and responsibilities were reconfirmed, ensuring that all team members understood their roles and were ready to take on their assigned tasks.

Action Items:

- Developers to complete the setup phase and be ready for Sprint 4.

Low Fidelity Wireframe, Storyboard, Wireframe Prototype

- **Date:** 23-10-2023
- **Meeting Duration:** 30 minutes

Meeting Details/Minutes:

Agenda:

- Review the progress of low-fidelity wireframes.

- Discuss the initial storyboard and wireframe prototype.

Meeting Summary:

1. The meeting began with a review of the progress made on the low-fidelity wireframes. The UI/UX designer presented their initial designs, and the team provided feedback.
2. The team discussed the initial storyboard, ensuring that it accurately represented the user flow and interaction.
3. The wireframe prototype was demonstrated to the team, allowing everyone to interact with the design and provide feedback.
4. The project manager emphasized the importance of user feedback and iterative design in the upcoming sprints.

Action Items:

- UI/UX designer to refine wireframes based on feedback.
- Developers prepare for Sprint 3, focusing on project initiation and setup.

3. Design of Components

The system has been divided into multiple components. Each component has been defined below:

Landing Page: Users can view the details of the website using the landing page, this is the first page that will be shown to the user.

Authentication: Users can login into the system using the authentication component. Based on the user role, we can decide which components can be visible to the user and what cannot be accessed by the user.

Current Progress: This component allows the user, specifically the student to view the current progress they have made. This includes showing completed courses, grades obtained and due courses.

Course Recommendation: The course recommendation component allows the user to specify their preference based on their commitment and gives them an ideal suggestion for the course they should select.

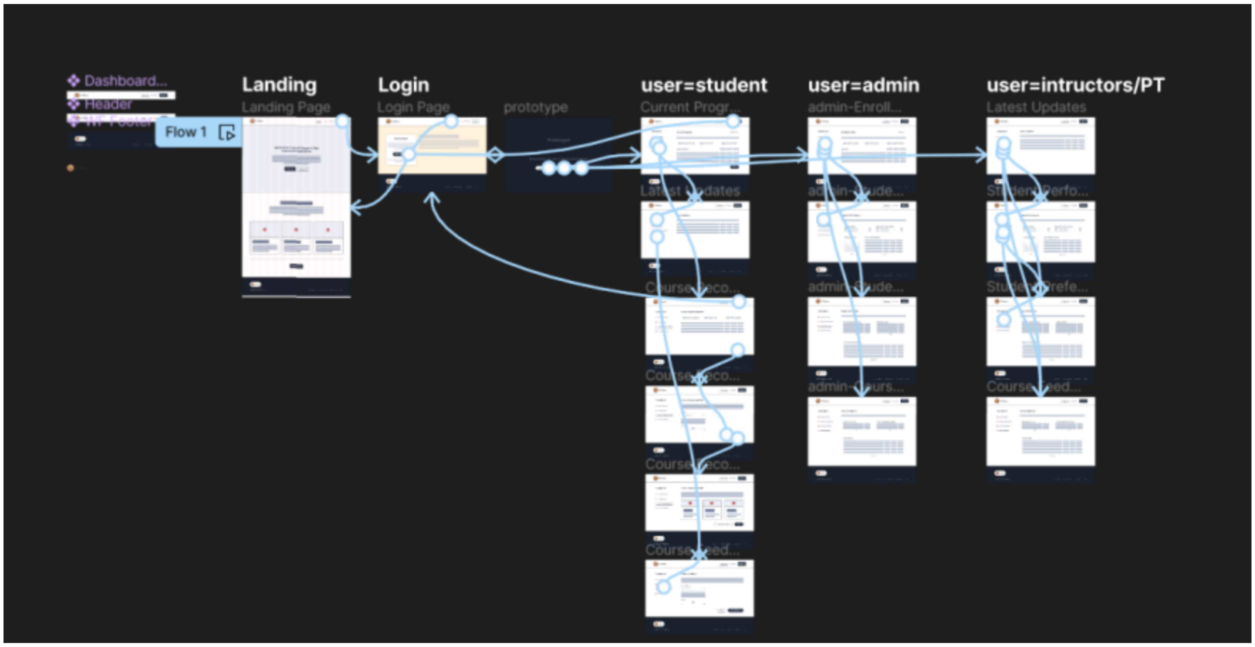
Course Feedback: The component allows students to give feedback for the courses they have completed. The planning team can view this feedback.

Notifications: Allows users to view the latest updates. Team can view alerts based on student performance.

Enrollment Data: Admin can upload, delete and view the enrollment data. This is a key requirement as multiple components depend on this component.

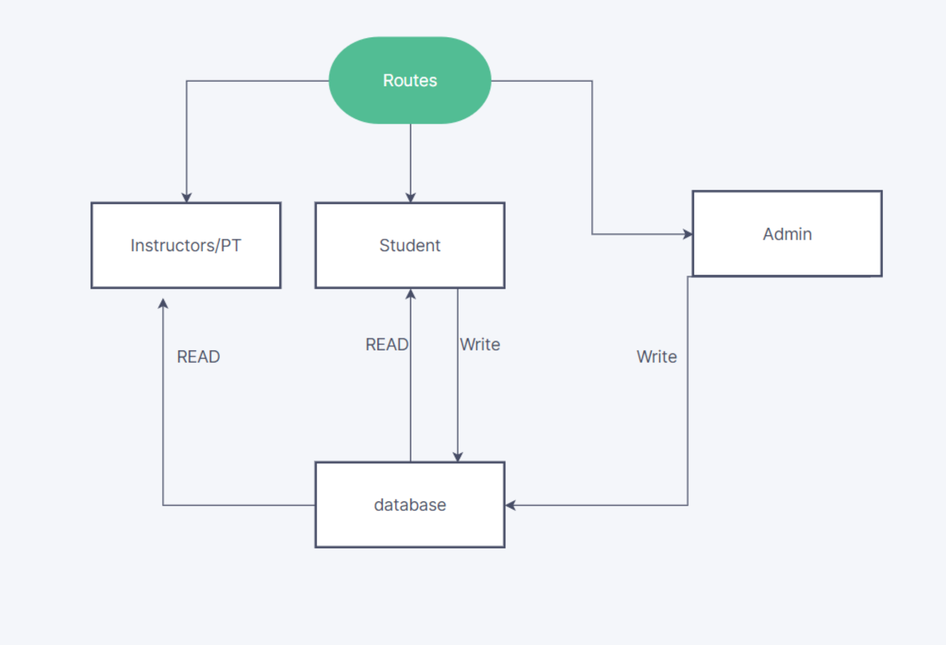
Student Performance: The team can view student performance. It includes, course wise performance, best performing students and pass fail ratio.

4. Software Design



Class Diagram:

- There will be 4 classes
  - **Routes**: Which consists of all the routes of the single file application
  - **Instructors**: The instructor will have read access to the database
  - **Admin:** admin will update the student database so, they will have write access to the database. They will be pushing the data
  - **student :** Both read and write access to the database. They will enter their info, and see their progress and recommendations.

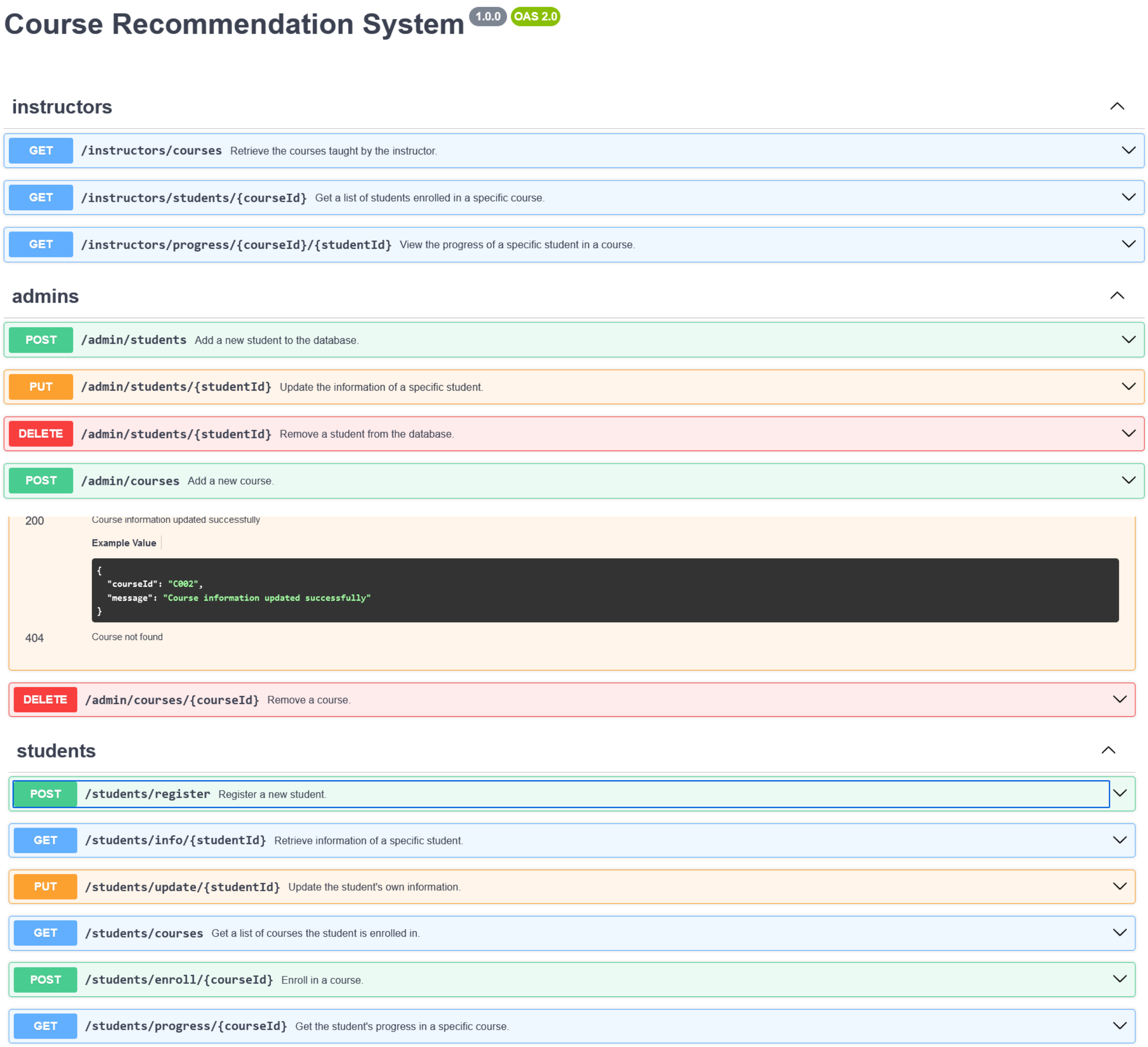


Milestone 4 - API Endpoints:

During milestone 4, we developed a RESTful API for our project and documented it in a YAML file. Drawing from the class diagram, we identified distinct resources and established corresponding endpoints. These endpoints encompassed the URL, anticipated request body, expected response, status codes, and error messages. Adhering to the OpenAPI specification, our API is testable through tools such as SwaggerUI, Insomnia, or Postman.

Code Review, Issue Reporting, and Tracking:

A screenshot of the Swagger documentation for the API can be found in the attached image.



Course Recommendation System 1.0.0 OAS 2.0

instructors

GET /instructors/courses Retrieve the courses taught by the instructor.

GET /instructors/students/{courseId} Get a list of students enrolled in a specific course.

GET /instructors/progress/{courseId}/{studentId} View the progress of a specific student in a course.

admins

POST /admin/students Add a new student to the database.

PUT /admin/students/{studentId} Update the information of a specific student.

DELETE /admin/students/{studentId} Remove a student from the database.

POST /admin/courses Add a new course.

200 Course information updated successfully

Example Value

```
{
  "courseId": "C002",
  "message": "Course information updated successfully"
}
```

404 Course not found

DELETE /admin/courses/{courseId} Remove a course.

students

POST /students/register Register a new student.

GET /students/info/{studentId} Retrieve information of a specific student.

PUT /students/update/{studentId} Update the student's own information.

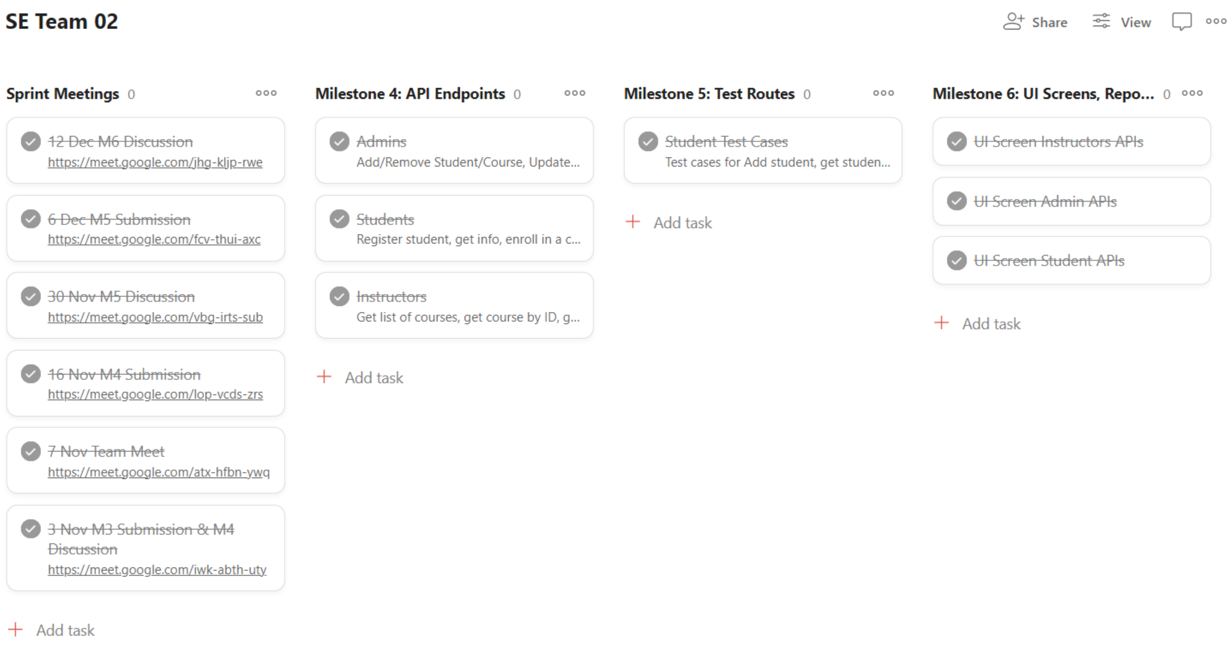
GET /students/courses Get a list of courses the student is enrolled in.

POST /students/enroll/{courseId} Enroll in a course.

GET /students/progress/{courseId} Get the student's progress in a specific course.

All code was reviewed using the following process:
Developers submitted their code for review using pull requests.
Other developers reviewed the code and provided feedback.
The original developer made changes based on the feedback.
The code was then merged into the master branch.
Issues were tracked using a project management tool, i.e Pivotal Tracker.

Issue Tracking: Kanban Board



SE Team 02

Sprint Meetings 0

- ~~12 Dec M6 Discussion~~
  https://meet.google.com/jhg-kljp-rwe
- ~~6 Dec M5 Submission~~
  https://meet.google.com/fcv-thui-axc
- ~~30 Nov M5 Discussion~~
  https://meet.google.com/vbg-irts-sub
- ~~16 Nov M4 Submission~~
  https://meet.google.com/lop-vcds-zrs
- ~~7 Nov Team Meet~~
  https://meet.google.com/atx-hfbn-ywq
- ~~3 Nov M3 Submission & M4 Discussion~~
  https://meet.google.com/iwk-abth-uty

Milestone 4: API Endpoints 0

- ~~Admins~~
  Add/Remove Student/Course, Update...
- ~~Students~~
  Register student, get info, enroll in a c...
- ~~Instructors~~
  Get list of courses, get course by ID, g...

Milestone 5: Test Routes 0

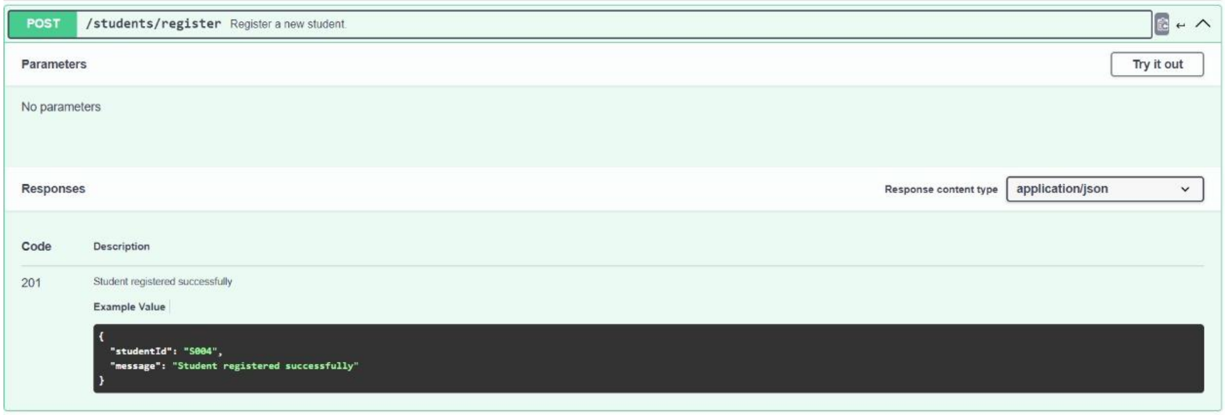
- ~~Student Test Cases~~
  Test cases for Add student, get studen...

Milestone 6: UI Screens, Repo... 0

- ~~UI Screen Instructors APIs~~
- ~~UI Screen Admin APIs~~
- ~~UI Screen Student APIs~~

<u>Milestone 5 - Test cases, test suite of the project:</u>

Summary: This milestone focused on defining and documenting comprehensive test cases and building a test suite for the project's API endpoints. Each endpoint was covered with exhaustive scenarios, ensuring thorough functionality and error handling.



POST `/students/register` Register a new student.

Parameters

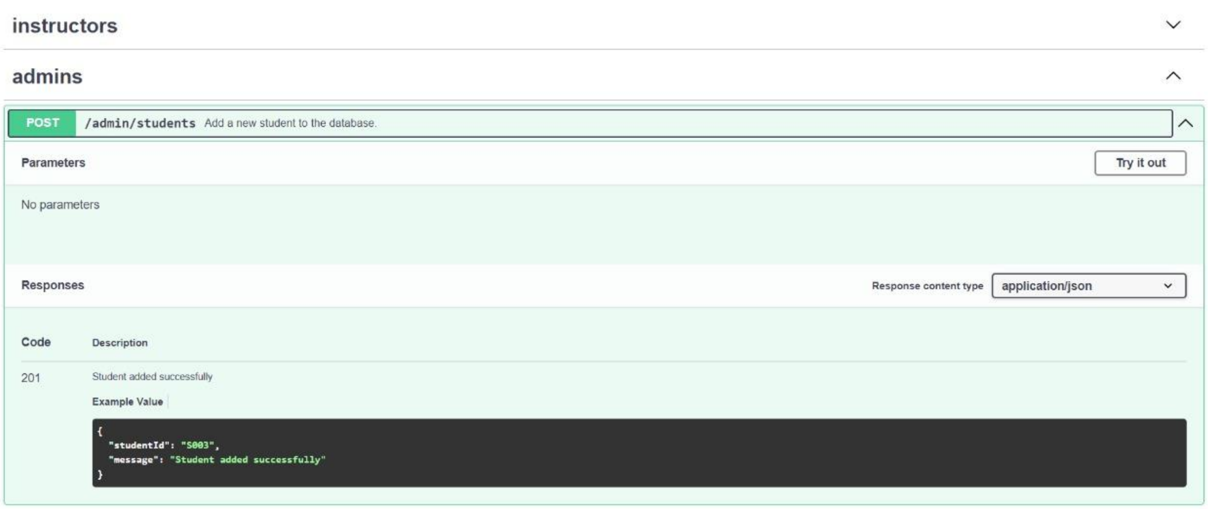
No parameters

Responses

Response content type: application/json

Code	Description
201	Student registered successfully Example Value

```
{
  "studentId": "S004",
  "message": "Student registered successfully"
}
```



instructors

admins

POST /admin/students Add a new student to the database.

Parameters

No parameters

Responses

Response content type: application/json

Code	Description
201	Student added successfully

Example Value

```
{
  "studentId": "S003",
  "message": "Student added successfully"
}
```

Pic: Sample Api and the result

Implementation Detail:

Technologies Used in Backend:

- Flask Restful
- flask_cors
- Falak security too Token-based authentication SQL Alchemy
- SQLite3
- Pytest
- Unit test

Technologies Used in Front end:

- ReactJS
- CSS and HTML
- JavaScript

Tools Used:

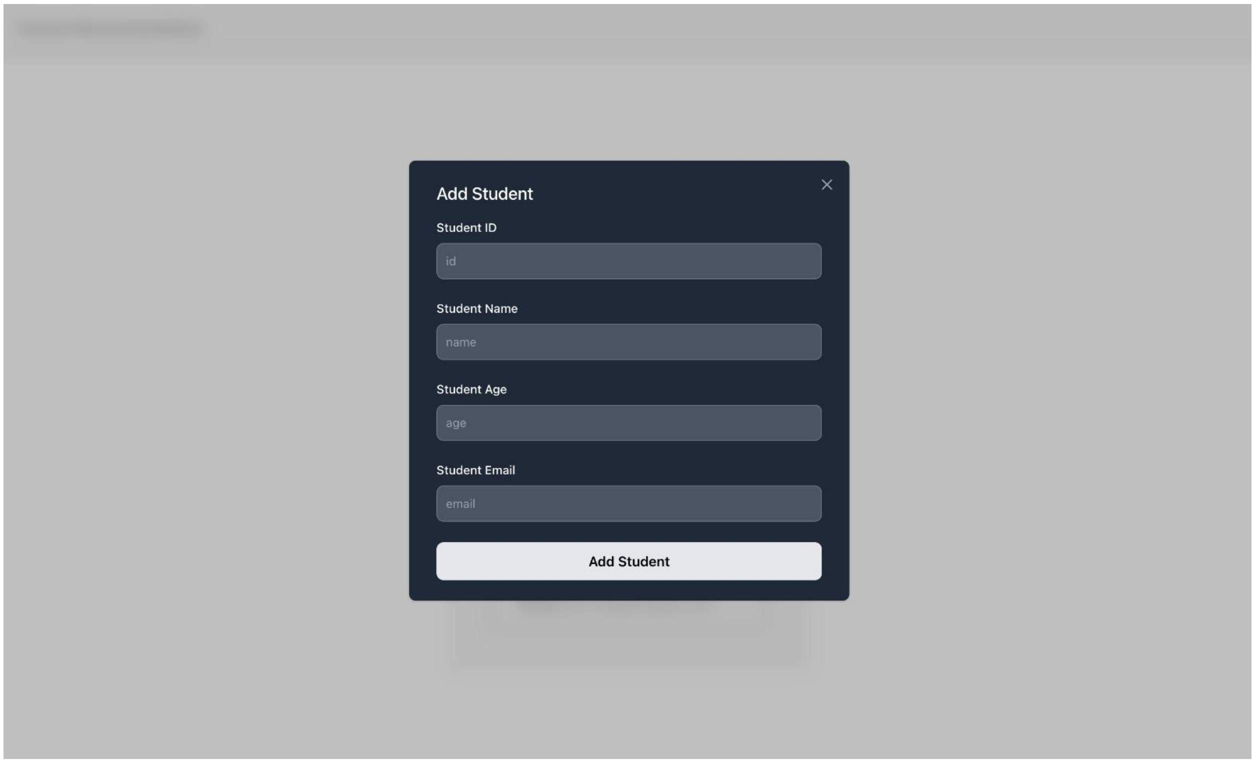
The following tools were used in the development of the project:

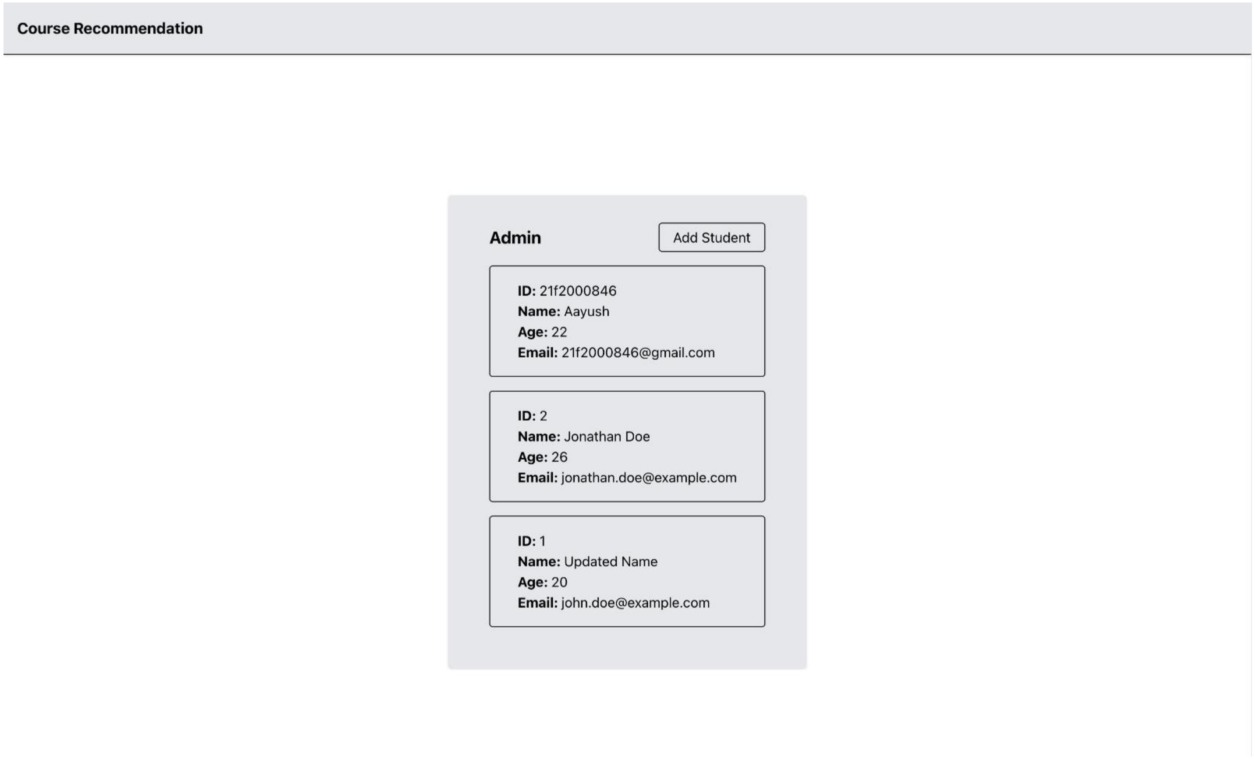
- VScode IDE: Used as the development environment.
- Pivotal Tracker: Used for schedule tracking.
- Google Drive: Used for document collaboration.
- Google Meet: Used for collaborative development and pair programming.
- Figma: Used for creating storyboards
- GitHub: Used for code version management

Milestone 6 - UI screens for api endpoints:

Instructions to Run the Application:

- Git clone the repository.
- Change the directory to the "application" directory inside the "Milestone-6-Final-Submission" directory





ID: 21f2000846
Name: Aayush
Age: 22
Email: 21f2000846@gmail.com

ID: 2
Name: Jonathan Doe
Age: 26
Email: jonathan.doe@example.com

ID: 1
Name: Updated Name
Age: 20
Email: john.doe@example.com

🔗 PPT Link

🔗 Video