Sprint Four Iteration Plan

July 22nd 2024

1. Process

Start date: Mon July 22, 2024
End date: Fri Aug 2, 2024

1.1 Roles & responsibilities

To enhance our team's collaboration, we have clearly defined and assigned specific tasks and responsibilities to each member. Peer programming is required this sprint, particularly for tasks that are closely related and demand coordination. For instance, we used peer programming for the tasks of accepting requests for tutoring appointments and displaying the list of requests. These tasks were very similar and required careful synchronization. Additionally, it's essential that each team member promptly notifies others of any significant updates to their branch. This ensures that everyone can seamlessly continue working on their tasks and maintain project momentum.

1. Michael Walker: Scrum Master
   - Create final set of user stories
   - Break down user stories to be more granular for each sprint
   - Facilitate team meetings
   - Maintain JIRA board and assign tasks on JIRA
   - Feature(s): Sending and creating appointment requests to tutors (includes date, time, hour, topic, and description (frontend + connection to database)
2. Aarushi Doshi: Developer + tester
   - Assist in breakdown of user stories to be more granular for each sprint
   - Check/Approve all Pull Requests before merging to sprint3 branch
   - Test application after each merge to ensure code is working
   - Inform teammates of new merges
   - Deploy to main branch
   - Feature(s): Display static calendar for students (not editable) (frontend)
3. Arina Azmi: Developer + Designer
   - Take meeting minutes and keep team updated on new developments
   - Check/Approve all Pull Requests before merging to sprint3 branch
   - Confirm final UI before merging branches with sprint3
   - Assist in breakdown of user stories to be more granular for each sprint
   - Feature(s): Create editable calendar for tutors (should allow creating, deleting, updating and moving events) (frontend + connection to database)
4. Miri Huang: Developer
   - Assist in breakdown of user stories to be more granular for each sprint
   - Feature(s): Create an 'Appointments' tab where tutors can view a list of their pending, current, and archived appointments. Students should have a similar page with pending, upcoming and archived appointments. (frontend)
5. Anusha Karkhanis: Developer
   - Assist in breakdown of user stories to be more granular for each sprint
   - Feature(s): Tutors should be able to accept or deny requests from their 'Pending' section in the appointments tab. If accepted, this should modify, the tutors, available times on their calendar and it should move the request from 'Pending' to 'Upcoming' for both, the tutor and the student (frontend + connection to database)

1.2 Events

1. Daily Standups: These meetings will take place online (Discord or Slack) and occasionally in-person, at 9pm or 10pm EST every weekday and after CSCC01 lecture (Tuesday 7 pm). During each standup meeting, each team member will have the opportunity to discuss what they have completed since the last meeting, what they are planning to complete by the next meeting, and if they are having any issues and require assistance. This will also be the time to make any larger team decisions, such as changing UI, using a different API, not being able to complete a task, creating new schemas, etc.
2. Weekly Scrum Meeting: Next Saturday at 5 PM, the entire team will meet to review the progress of the developed features and overall weekly achievements. We will examine the merged code to ensure the application is functioning end-to-end and address any significant blockers teammates are facing.
3. Sprint Retrospective Meeting: On Thursday, August 1st, the team will meet at 8 PM to reflect on the project's overall progress. We will discuss the key takeaways from this sprint and the entire project, and identify lessons learned. This will help us in future projects and improve our team dynamics.
4. Final Sprint 4 Meeting: On Friday, August 2nd, the team will meet online at 4 PM to finalize the sprint. We'll ensure all code is functioning, documentation is complete, and no tasks are pending. We will also plan and discuss the demo presentation.

2. Product: Goals and Tasks

2.1 Editing Availability

The ability for tutors to edit their weekly availability is essential for managing their schedules efficiently. This feature allows tutors to define and adjust their availability for tutoring sessions through a user-friendly calendar interface.

- Calendar Interface Details
    - Tutors will use the Calendar for managing their availability.
    - The calendar will allow tutors to drag and select time slots for each day of the week.
    - Tutors can set their availability for up to two months in advance and make adjustments on a weekly basis.
- Event Management
    - Tutors can create events by selecting available time slots.
    - Events can be moved around, resized to adjust the time range, or deleted as needed.
    - Changes made to events will be immediately reflected in the calendar.
- API Utilization
    - The DayPilot Calendar will handle the calendar interface and event interactions.
    - The backend will update in real-time with each event saved as an object containing start and end times along with the date of availability.

2.2 Viewing Tutor Availability

The feature allowing students to view a tutor’s availability for the upcoming two months is crucial for efficient scheduling. It provides students with a clear view of when a tutor is available, helping them plan and book sessions more effectively.

- Calendar Interface Details
    - Students can select a specific tutor on the Home Page and click the 'View Availability' button to access the tutor's calendar.
    - The calendar, powered by DayPilot, will display a weekly view of the tutor's availability for the next two months.
    - Students can navigate through weeks using the monthly navigator on the left side of the calendar.
- Event Management
    - The availability events displayed on the calendar are static and cannot be edited by anyone other than the tutor.
    - This ensures that students see accurate and unaltered availability information.
- API Utilization
    - The DayPilot Calendar will be used for displaying the tutor's availability and handling calendar interactions.
    - The calendar will present the availability as static events, reflecting the tutor's defined schedule for the upcoming two months.

2.3 Requesting Tutoring Appointments

The feature allowing students to request appointments with tutors based on their availability is essential for streamlining the scheduling process. It enables students to book sessions at times that are convenient for both parties, ensuring a more organized and efficient scheduling experience, and that students get the help they need, when needed.

- Appointment Request Interface
    - On a tutor's profile page, students can click the 'Book Appointment' button to open a pop-up form.
    - The pop-up will prompt students to provide necessary details for the appointment request listed below:
        - Course and Description: Students must specify the course they need help with and provide a description of their specific requirements.
        - Date Selection: A dropdown menu will display all available dates for the tutor. Students will select a specific date when the tutor is available.
        - Time Range Selection: Another dropdown menu will show available time slots for the selected date. This menu will only display hours when the tutor is available on that particular day.
- Once all required information is filled out, students can click the 'Submit' button at the bottom of the form to send the appointment request to the tutor.

2.4. Accepting and Denying Appointments

The feature for viewing appointments provides tutors and students with an organized way to manage and track their sessions. This includes the ability to see pending requests, upcoming sessions, and archived sessions, ensuring that both parties can keep their schedules up-to-date and well-managed.

- For tutors:
    - Appointments Tab
        - Tutors can access the 'Appointments' tab from the navigation bar.

        - They can toggle between three sections on this page: 'Upcoming Tutoring Sessions,' 'Pending Requests,' and 'Archived Tutoring Sessions.'
    - Pending Requests
        - Tutors will see new appointment requests in the 'Pending Requests' section. These are requests made by students that are waiting to be either accepted or denied by the tutor
        - Requests are displayed as cards on the left side of the screen.
        - Each card includes the student's profile picture and the course they need help with.
        - Clicking a specific request will reveal the full description on the right side of the screen.
    - Upcoming Sessions
        - Accepted requests will be listed in the 'Upcoming Tutoring Sessions' section.
        - This will have a similar layout and UI as the 'Pending Requests' section
    - Archived Sessions
        - The 'Archived Tutoring Sessions' section displays a list of past meetings.
        - This will have a similar layout and UI as the 'Pending Requests' section
- For Students
    - Appointments Tab
        - Students will also use the 'Appointments' tab to view their requests, upcoming tutoring sessions, and archived/past tutoring sessions
        - They will see a list of their pending requests, displayed similarly to the tutor's interface, but showing the tutor's name and profile picture.
    - Pending Requests
        - Students can view requests they have sent out and are waiting to be approved by tutors.
    - Archived Sessions
        - The 'Archived' section for students shows a list of past meetings, similar to the tutor's view.

2.5. Accepting and Denying Appointments

The ability for tutors to accept or deny requests from students is crucial for effective appointment management. This feature ensures that tutors can control their schedule by reviewing and responding to each request, which helps maintain an organized and manageable workload. For students, being able to view whether their request has been accepted or denied provides transparency and helps them track the status of their appointment. This clarity allows students to plan accordingly and seek alternative options if needed, ensuring a smoother scheduling process for both parties.

- For tutors:
    - Pending Requests
        - When a specific student is selected, tutors can review the details of the tutoring session request, message the student, and then choose to accept or deny the request.
        - If accepted, the request will move from 'Pending' to 'Upcoming' and the tutor's availability calendar will update in real time to reflect the new booking.

- For Students
    - Pending Requests
        - Students can view requests they have sent out and are waiting to be approved by tutors.
        - If a request is approved, it will move to the 'Upcoming' section.
        - If a request is denied, it will be removed from the list.

3. Product: Artifacts

3.1 Division of Responsibilities

Responsibilities for this sprint are assigned based on task complexity, team members' availability, and personal preferences. Peer programming is used for related tasks, such as approving requests and updating the list of requests, to ensure consistency and efficiency.

The calendar editing feature is given to the team member with the most availability at the start of the sprint, as it is crucial for the functioning of other features. Completing this feature is essential for enabling other functionalities that depend on accurate calendar management.

In this sprint, we focus on the final set of features, which are central to our project's core functionality. These features are vital as they enable students to request sessions with their chosen tutors and help tutors manage their availability effectively. By completing these features, we finalize our project and ensure it is fully integrated and operational.

3.2 JIRA Management

We will keep track of user stories in our JIRA backlog for the sprint and assign points to each user story to estimate how difficult a task is. We will reassess the points after each week to ensure that we have taken on an appropriate amount of work for the sprint. Child stories will be created for each user story, to make the tasks more granular. Once a user story is completed, it will be marked as "Done" on our JIRA board. Currently, we have decided to remove the 'AI Transcription' feature and the 'Calendar of upcoming tutoring sessions' feature due to lack of time and not enough resources. If time permits, a team member can take on one of these features.

JIRA Board

3.3 Team Organization

At the beginning of each sprint, the team will fill out a when-to-meet to indicate the times that they will be available over the following 2 weeks. This way, we can set specific times for our standups and even know when other teammates may be available to help with a blocker.

To ensure that the team is always on the same page, after each standup, Arina will make a short to-do list, listing the tasks that are yet to be completed for the sprint, alongside who needs to complete each task and which ones require collaborative effort. This will help all team members be aware of which tasks are left, and we will be able to plan out our schedule more effectively.

3.4 Sprint Retrospective

Once this Sprint is complete, our team will meet to go over all aspects of the sprint and discuss what went well, what could be improved and what didn't go well for the sprint. This will help us improve for the upcoming sprints.