SW Engineering CSC 648-848 Fall 2023
weLearn - A Comprehensive SFSU Tutoring Service

Team 4
Milestone 1

Akshat Sohal - Team Lead (email: asohal@mail.sfsu.edu)
Zuriel Respicio - Frontend Lead
Charter Lin - Backend Lead
Aakanksha Devarapally - GitHub Master
Andy Byeon
Jorge Perez

Date submitted: 10/01/2023

Date Revised: 10/05/2023

1. Executive Summary -

Our motivation for developing the tutoring application stems from a shared understanding of the challenges students face in their academic journeys, particularly in the demanding environments of schools and colleges. Graduation is the ultimate goal for every student, yet it often involves overcoming hurdles, including challenging courses. To address these challenges, we have conceived an innovative tutoring web application designed to provide struggling students with the necessary support to successfully navigate their courses.

What sets our application apart is its unique focus on the San Francisco State University (SFSU) community. Unlike generic tutoring platforms, our application exclusively connects SFSU students in need of academic assistance with fellow SFSU students who have excelled in the same courses. This tailored approach not only enhances academic outcomes but also fosters a sense of camaraderie within the SFSU student body. Our application not only enhances academic outcomes but also offers an opportunity for students to tutor courses that they excelled in.

Our startup team consists of 6 members. Everyone in the team works well together and the team leader makes sure that everyone is on task and focused on work. In our startup team, there is one team leader, one front-end lead, one back-end lead, and one github master. We also have one front-end helper and one back-end helper to help the front-end lead and back-end lead with work. Our startup team is very organized and we have two to three meetings a week.

2. Personae and main Use Cases -

Student:

Student Name: Jane (Photo from unsplash.com)

Characteristics	Goals	Pain Points
● SFSU student ● Basic knowledge of technology (i.e. navigating webpages) ● Hardworking and willing to learn ● ● Stressed because of difficult classes	● Get help with homework and understanding course material ● Find a tutor that can help with their specific class ● Choose a tutor that they feel comfortable meeting up with ● Book appointment with a tutor in-person or online	● Might struggle with complex UI and would need a user friendly interface ● Might not able to search for the right tutor for specific coursework ● Not many tutors available for that specific course

Tutor:

Name: Alex (Photo from unsplash.com)

Characteristics	Goals	Pain Points
● Student at SFSU ● Basic knowledge of tech (i.e. navigating web pages) ● Patient ● Flexible ● Passionate about subject/major	● Looking for a part time job as a tutor ● Register and apply as a tutor ● After application is approved, needs to schedule to meet up with other students ● Choose which students they are willing to tutor	● Might struggle with complex UI and would need a user friendly interface ● Won’t wait too long to get application approved ● Busy schedule (classes, hw, etc) and might struggle to fit in tutoring appointments

Admin:

Name: Vicky (Photo from unsplash.com)

Characteristics	Goals	Pain Points
• Professional • Very familiar with technical work • Experienced in managing databases and understanding SQL • Aware of company policies • Detail-oriented • Organized	• Approve tutor applications and their profiles before adding to website • Make sure no inappropriate content goes live and continuously monitor profiles • Protect user data	• If site gets too much traffic might take too long to approve tutors (more traffic = more applications) • As more tutors are registered, they have to monitor more profiles which will take up more time • Might need training if not familiar with SQL workbench

Use Cases:

1: Student is Searching for a Tutor for a class at SFSU

Jane is a student at SFSU and is struggling in one of her classes. She's usually a good student and puts a lot of effort into her classes. She's worried that she will fail her class if she doesn't get help and is looking for a tutor to help her understand the class material. She finds our website and searches for her class by the class name. She is relieved to find that there is a tutor for her class in the search results. She selects the tutor and chooses to message them asking to make an appointment. At this point, if she isn't already a registered user, she will be asked to register and make an account before she can send her message. After sending the message, she waits for the tutor to reach out to her. After the tutor contacts her confirming the appointment, she meets up with them and gets the help she needs.

2: Student Applies for a Tutoring Position

Alex has been a student at SFSU for a couple years and is looking for a part-time job that is convenient and won't interfere too much with his classes. He finds our website which offers tutoring services for SFSU students by other SFSU students. He always had a lot of interest in his subject and shared his passion for it with others. This seems like the perfect opportunity for him since he can use his knowledge from his major on the job and stay on campus while working. He finds a link on the page to the form to apply as a tutor and fills it out. If he isn't already registered, he'll be asked to make an account before he can submit the application. He submits the form and waits for admin to approve his application. Once approved, he will have a tutor profile page providing information about himself and the classes he tutors generated from his application. He will now receive messages from other students asking for tutoring.

3: Admin Day-to-Day Responsibilities

Vicky was recently hired as administrator for our web application and has been trained on using SQL workbench. Every day he checks the new tutor applications and tutor profiles using workbench. He approves or rejects the new applications to the criteria set by our organization. If a tutor application is approved, their profile will go live on the site using the information provided in the application. Vicky then checks other existing tutor profiles to make sure they weren't modified to include any offensive content. He temporarily takes down any profiles containing such content. He will notify that tutor about the offensive content and tell them their profile won't be reinstated until they remove the content. He will also check new registered accounts for any offensive usernames or pictures. Vicky temporarily bans offensive accounts and will let them know their account will be reinstated after the user corrects the issue. Vicky plays a crucial role in ensuring that only qualified and reliable tutors join our platform and diligently monitors the platform to maintain it as a credible and safe educational environment. He greatly contributes to the quality and trustworthiness of our services.

4: Reviewing Messages and Posts on the Dashboard

Alex is a tutor on our website. Jane is an SFSU student who uses our website to find tutors. All registered users have a dashboard that is located on the home page. On the dashboard, Alex is able to see messages that were sent by students. The messages are from students who are interested in Alex being their tutor. Another function of the dashboard that Alex can use is it shows all of the posts that he created. Posts are a little introduction about the tutor and which subjects they excel at. Since Jane is a registered user that is not a tutor she sees all the messages that she has sent instead of messages that were sent to her.

3. List of main data items and entities - data glossary/description -

This part is asking for Attributes for each entity

Unregistered User:

- User ID: A unique identifier for tracking user interactions.
- Session ID: A temporary identifier for tracking user sessions.
- Access Timestamp: Timestamp indicating when the unregistered user accessed the system.

Registered User:

- Student ID: A unique identifier for each student.
- First Name: The first name of the student.
- Last Name: The last name of the student.
- Email Address: The email address associated with the student's account.
- Username: The username used for authentication.
- Password: Securely stored password for login.
- Enrollment Status: Indicates whether the student is currently enrolled.
- Tutor Record: Resume, video, photo, messages

Admin:

- Admin ID: A unique identifier for each administrator.
- First Name: The first name of the administrator.
- Last Name: The last name of the administrator.

Tutoring Record: what the tutor submits when applying to be a tutor examples: topic, Resume, video, etc.

- Topic: A subject that the tutor wants to help the student with/ a subject that the tutor excels at
- Description: A short bio about the tutor
- Class numbers: A list of class numbers for the courses that the user wants to tutor
- Resume: A documentation of accomplishments, skills, etc.
- Profile picture: An image of the tutor
- Video: A recording of an introduction or message created by the tutor

Messages:

- Message sender: email of the user sending the message
- Message recipient: email of the user receiving the message
- Message body: What the user wants to send to the recipient

4. Initial list of functional requirements -

Unregistered User:

- An unregistered user shall be able to access public information on the platform without the need for authentication.
- Unregistered users shall have the capability to view a list of available tutors on the platform.
- An unregistered user shall be able to register and become a registered user.

Registered user:

Registered users shall have access to functional requirements mentioned above along with the following:

- A registered user shall be able to create an account.
- A registered user shall have an email, username, and password
- A registered user shall be able to log in using email/username and password credentials.
- Registered user shall be able to search for SFSU classes they need help with
- Registered users shall have the ability to view tutor profiles.
- Registered users shall be able to send messages to tutors asking for appointments in person or online
- Registered users shall be able to apply as a tutor
- Registered users shall be able to post picture, resume, and optionally, a video introduction if they want to be a tutor.
- Registered users (if approved as tutor) shall be able to receive messages from student asking for tutoring

Admin:

- Admins shall be required to review and approve or reject tutor profiles.
- Admin shall be required to remove inappropriate or offensive content from the platform.
- Admin shall be able to access tutor profiles, resumes, profile pictures, and videos.

- Admin shall be able to access student username, email, and profile picture.

5. List of non-functional requirements -

1. Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0
2. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers
3. All or selected application functions shall render well on mobile devices
4. Data shall be stored in the database on the team's deployment server.
5. No more than 50 concurrent users shall be accessing the application at any time
6. Privacy of users shall be protected
7. The language used shall be English (no localization needed)
8. Application shall be very easy to use and intuitive
9. Application shall follow established architecture patterns
10. Application code and its repository shall be easy to inspect and maintain
11. Google analytics shall be used
12. No e-mail clients shall be allowed. Interested users can only message to sellers via in-site messaging. One round of messaging (from user to seller) is enough for this application
13. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.
14. Site security: basic best practices shall be applied (as covered in the class) for main data items
15. Media formats shall be standard as used in the market today

16. Modern SE processes and tools shall be used as specified in the class, including collaborative and continuous SW development
17. The application UI (WWW and mobile) shall prominently display the following exact text on all pages *“SFSU Software Engineering Project CSC 648-848, Fall 2023. For Demonstration Only”* at the top of the WWW page nav bar. (Important so as to not confuse this with a real application).
18. Public information shall include general details about the platform's services, FAQs, and contact information.

6. Competitive analysis -

+ = feature exists

++ = superior

- = does not exist

Feature	**Competitor 1: Varsity Tutor**	**Competitor 2: Chegg Tutor**	**Competitor 3: Wyzant**	**Our Future Product**
Search for tutors by SFSU Class	-	-	-	++
SFSU Specific Class search	-	-	-	++
Tutor that is a student	-	-	-	+
Search by SFSU majors	-	-	-	++
In Person/Online Appointment	+	+	+	++

Viewing Tutor profile	**-**	**-**	+	+

Our application stands out from the competition as users can search for coursework-specific tutors. Moreover, niche search for the tutor makes the user experience better for the students. Normally if a student and tutor want to have a one-to-one session they would schedule an online meeting time. In our application, we give the option for students and tutors to meet in person or online. We believe if the tutor and student are both SFSU students, they would have a stronger connection.

7. **High-level system architecture and technologies used -**

- List all main SW components and versions (DB, WWW server)

  DB - MySQL

  WWW - NGINX

- List deployment cloud servicer you plan to use

  AWS Free Tier, EC2

- List front-end frameworks

  1. Bootstrap
  2. React
  3. JavaScript
  4. CSS

- List browsers you plan to support (chose 2 market leading browsers, last two versions from each)

  Market leading browsers - Google Chrome and Safari

  Google Chrome versions - Chrome 115 and Chrome 116

Safari versions - 16.6 and 15.6.1

- List any major additional external open source APIs you plan to use (e.g. Google analytics, Google map APIs, APIs/service for creating thumbnails - check Architecture class slides)

  Google analytics

- Here just say if you used ChatGPT and describe it as below, section 8

  Yes, we use ChatGPT. We used it to help us come up with use cases and personas.

- In no more than one brief paragraph describe how will you implement search functionality (OK to do it as suggested in class architecture slides, but say it here)

To implement search functionality, we will utilize MySQL's % SEARCH feature. This feature allows for efficient and flexible searching within our database. When a user initiates a search query on our website, we'll construct SQL queries that incorporate the % SEARCH functionality to find relevant records based on their input. By leveraging this feature, we can provide accurate and speedy search results, enhancing the user experience by making it easier for them to discover content on our website.

8. **ChatGPT -** Front end team works or together

   We used GPT-3.5. We have reviewed and followed ChatGPT class slides and policies. Yes, ChatGPT was helpful, we would recommend using this in the future.

   Tasks:

   Use cases - Medium

   The benefits it offered were examples, saving time, and helping with brainstorming.

   ChatGPT provided examples of how use cases looked. After seeing a few examples we started brainstorming how we could create our own use cases. Definitely seeing examples made it easier to create the use cases. ChatGPT made the time to create the use cases shorter. We drafted our assignment content before using ChatGPT.

Personas - Medium

The benefits it offered were it gave a variety of examples, explained in detail what personas purpose was, etc. At the start, we got confused about what personas were supposed to be. After seeing an example in class and using ChatGPT for what we need to include in a personas we were able to finish the personas section very quickly. We are still learning how to better utilize ChatGPT but we definitely gained more knowledge of how to use ChatGPT more properly and efficiently. Sometimes ChatGPT can be useful and sometimes it doesn't give good answers. We used ChatGPT first for the structure and how it should look.

9. **Team and roles**

Akshat Sohal - Team lead

Zuriel Respicio - Front team lead

Charter Lin - Back team lead

Andy Byeon - Team Member working with Front-End Team Lead

Aakanksha Devarapally - GitHub master & document editor

Jorge Pérez - Team Member with Back-End Team Lead

10. **Checklist: checklist gets filled out by team lead choices are DONE/OK or ON TRACK or ISSUE -** Only team lead is allowed to work on this

- So far all team members are engaged and attending team sessions when required **OK**

- Team found a time slot to meet outside of the class **Done**
- Back end, Front end leads and Github master chosen **Done**
- Team ready and able to use the chosen back and front end frameworks and those who need to learn are working on learning and practicing **Done**
- Team reviewed class slides on requirements and use cases before drafting Milestone 1 **Done**
- Team lead ensured that all team members read the final M1 and agree/understand it before submission **OK**
- Github organized as discussed in class **Done**
- Every team-member did a final review of Milestone 1 **Done**