SW Engineering CSC648-848 Fall 2023

SFSU TutorLink

Team 02

Jeremy Woodling - Team Lead - jwoodling@sfsu.edu

Abel Seyoum - Github Manager

Brandon Watanabe - Front End Lead

Ankita Mukherjee - Backend Lead

Guillermo Villar

Lars Severson

Milestone 4

Submitted	Revised
12/13/23	

Table of Contents

1. Product Summary:

Product Name: SFSU TutorLink

Product Description: Introducing SFSU TutorLink — a revolutionary web-based platform tailored to address the unique challenges faced by San Francisco State University (SFSU) students beyond the classroom. Our platform connects SFSU students with experienced tutors who have successfully navigated the intricacies of different timetables, side jobs, and school-life balance, ensuring they are best equipped to help peers swiftly learn and excel in their courses. With a user-friendly interface, students can schedule their tutoring sessions through our messaging system, and link up with tutors matching their timetables and goals. While this may not immediately seem tightly connected to SFSU, we demonstrate our commitment to the community through unique offerings. These include the ability to search through your courses and tutors, adherence to all SFSU guidelines, and our focus on fostering connections within the community, thereby making our university a more supportive and collaborative learning environment.

ALL major committed functions:
Priority 1 (Necessary):

1. Unregistered User:

1.1. Unregistered users shall be able to register for one and only one account by using their "@sfsu.edu" email
1.2. Unregistered users shall be able to search for a tutor by subject
1.3. Unregistered users shall be able to search for a tutor by class
1.4. Unregistered users shall be able to browse tutors
1.3. Unregistered users shall be able to view a tutor's photo
1.4. Unregistered users shall be able to access a tutor's resume

2. Registered User:

2.1. A registered user shall be able to login
2.2. A registered user shall be able to request a new password
2.3. A registered user shall inherit all functions available to an unregistered user
2.4. A registered user shall have one and only one dashboard
2.5. A registered user shall be able to send a message to a tutor
2.5. A registered user shall be able to submit an application for becoming a tutor
2.6. Tutors shall be able to upload one and only one picture
2.7. Tutors shall be able to upload one and only one resume

3. **Messages:**
3.1. Messages shall be able to be created by a user
3.2. Messages shall be able to be received by one and only one tutor

4. **Dashboard:**
4.1. The dashboard shall display zero to many of a registered user's tutor postings
4.2. User shall be able to view sent and received messages in the dashboard

5. **Admin:**
5.1. Admin shall be required to approve tutor applications prior to going live
5.2. Admin shall be required to deny inappropriate tutor applications
5.3. Admin shall be able to block users

Say what is unique in your product: Our app is unique in its SFSU-specific offerings. Students can search through their own courses and tutors, and they are linked with past students of those same courses, who can tailor their classes to the SFSU courses. This foundation enables us to focus our efforts not only on refining and improving these core elements but also on innovating and expanding upon them to deliver an exceptional and highly personalized learning journey that sets us apart from the competition.

Product URL: http://54.177.167.160/ **(Subject to change due to AWS issues)**

2. Usability test plan:

Selected function "Search"

Test Objectives:
The test aims to evaluate the usability of the "Search" function on the tutoring platform. Specifically, we want to assess the effectiveness and efficiency of users in searching for a tutor by tutor name, subject, and class name. The objective is to identify any potential usability issues and gather user satisfaction feedback.

Test Background and Setup:
System Setup: The testing environment will use a computer or mobile device with internet access and a standard web browser.

Starting Point: Testers will start from the platform's homepage and will be instructed to navigate to the "Search" function.

Intended Users: The intended users are individuals who may seek tutoring services, including students, tutors, and parents. Testers should have a basic understanding of web browsing.

URL: The system to be tested is accessible at **http://54.177.167.160/**

Test environment: Testing can be conducted in a controlled lab environment or at home. Monitoring may be implemented to observe user interactions. No prior training is required.

Plan for Evaluation of Effectiveness: Effectiveness will be measured by task success rate, i.e., the percentage of users who successfully find a tutor using the "Search" function within a given time frame. Tasks will include searching by tutor name, subject, and class name.

Plan for Evaluation of Efficiency: Efficiency will be measured by the time taken by users to complete each task. This will help assess the speed and ease of use of the "Search" function for finding a tutor.

Plan for Evaluation of User Satisfaction (Likert Scale Questionnaire):
A) Usability Task Description:

Task 1: Search by Tutor Name
Instructions: "Search for a tutor named 'John Smith' using the 'Search' function. Once found, click on the tutor's profile."

Task 2: Search by Subject
Instructions: "Search for a tutor who specializes in 'Math' using the 'Search' function. Click on the profile of the first tutor in the search results."

Task 3: Search by Class Name
Instructions: "Find a tutor for the 'Physics 101' class using the 'Search' function. Click on the profile of the tutor who teaches this class."

B) Likert Scale Evaluation Entries:
Task 1 Likert Entry: Search by Tutor Name

"Ease of completing the task"

○	○	○	○	○
Very Difficult	Difficult	Neutral	Easy	Very Easy

Task 2 Likert Entry: Search by Subject

"Satisfaction with the search results:"

○	○	○	○	○
Not Satisfied	Slightly Satisfied	Neutral	Satisfied	Very Satisfied

Task 3 Likert Entry: Search by Class Name

"Clarity of tutor information:"

○	○	○	○	○
Unclear	Slightly Clear	Neutral	Clear	Very Clear

Testers will provide Likert scale ratings based on their experience after completing each task. The scores will be analyzed to gauge overall user satisfaction with the "Search" function.

3. QA test plan and QA testing:

Test Objectives:

The QA test aims to ensure the functionality, reliability, and security of the "Search" feature on the TutorLink application. This includes validating the accuracy of search results based on tutor name, subject, and class name, and tutor description or search by subject categories.

HW and SW setup (including URL):

Hardware: Standard computer or mobile device.
Software: Web browser (Chrome, Firefox).
URL: **http://54.177.167.160/** (Subject to change due to AWS issues)

Feature to be Tested:

Feature: Search for a tutor by tutor name, subject, class name, and tutor description or search by subject categories.

Number	Description	Test input	Expected output	PASS/FAIL
1	Test % like in search for name field	Type “Shifty” in the search field	Get 1 results, all have “Shifty” in name field	PASS
2	Search by Subject Categories dropdown	Choose "CSC" in the subject category	Get 2 results, categorized under "CSC"	PASS
3	Search by Class Number	Type '1234' in the search field	Obtain 1 result; tutor profile includes 'PHYS 1234'	PASS

b) Browser Testing Results:

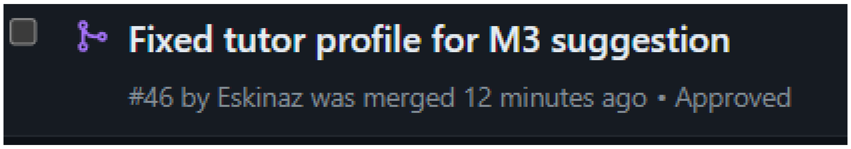
Number	Testcase	Browser	PASS/FAIL
1	Test % like in search for name field	Chrome	PASS
		Firefox	PASS
2	Search by Subject Categories dropdown	Chrome	PASS
		Firefox	PASS
3	Search by Class Number	Chrome	PASS
		Firefox	PASS

4. Peer Code Review

Email Requesting Code Review:
Sent from: aseyoum@sfsu.edu
Sent to: jwoodling@sfsu.edu

Can you please review my changes for the tutor profile fixes?
https://github.com/CSC-648-SFSU/csc648-03-fa23-team02/pull/46

Pull Request:



☐ Fixed tutor profile for M3 suggestion

#46 by Eskinaz was merged 12 minutes ago • Approved

Response:
Sent from: jwoodling@sfsu.edu
Sent to: aseyoum@sfsu.edu

Tested the code and its working, follows the standard, and the changes look good as well! Thanks Abel.

5. Self-check on best practices for security:

Asset to be protected	**Types of possible/expected attacks**	**Your strategy to mitigate/protect the asset**
User Authentication Data (e.g., passwords)	Broken Access Control, Cryptographic Failures	Encrypting passwords stored in the database. Ensuring robust access controls to prevent unauthorized access
Search Bar Input	Injection, Security Misconfiguration	Validating search bar input for up to 40 alphanumeric characters, implementing input sanitization, and using parameterized queries. Regularly testing and configuring search functionalities to prevent misconfigurations
SFSU Customer Registration Email	Injection, Security Logging and Monitoring Failures	Validating registration emails to include “sfsu.edu” at the end and implementing email verification mechanisms. Ensuring secure logging and monitoring practices to detect and respond to injection attempts. |
Application Source Code	Cryptographic Failures, Insecure Design	Conducting regular code reviews, implementing code obfuscation techniques, and restricting access to the source code. Ensuring proper use of cryptography in code design.

6. Self-check of the adherence to original Non-functional specs – performed by team leads:

#	Non-Functional Requirements	Status
1	Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0	DONE
2	Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers	DONE
3	All or selected application functions shall render well on mobile devices	DONE
4	Data shall be stored in the database on the team's deployment server.	DONE
5	No more than 50 concurrent users shall be accessing the application at any time	DONE
6	Privacy of users shall be protected	DONE
7	The language used shall be English (no localization needed)	DONE
8	Application shall be very easy to use and intuitive	DONE
9	Application shall follow established architecture patterns	DONE
10	Application code and its repository shall be easy to inspect and maintain	DONE
11	Google analytics shall be used	DONE
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	DONE
13	Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.	DONE
14	Site security: basic best practices shall be applied (as covered in the class) for main data items	DONE
15	Media formats shall be standard as used in the market today	DONE
16	Modern SE processes and tools shall be used as specified in the class, including collaborative and continuous SW development	DONE

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).	DONE

DONE - it is done
ON TRACK - in the process of being done and you sure it will be completed on time
ISSUE - means there is some problem, includes explanation