

SW Engineering CSC648/848 Fall 2021

TwoHelpYou

Project Application And Name: Provide tutoring to SFSU students – "TwoHelpYou."

Team Number: Team 01

Team Members and Roles:

1. Justin Lam (Team Lead) - jlam18@mail.sfsu.edu
2. Wesley Xu (Front-End Lead) - wxu3@mail.sfsu.edu
3. Dinesh Thapa (Back-End Lead) - dthapa@mail.sfsu.edu
4. Aviral Puri (Github Master) - apuri2@mail.sfsu.edu
5. Chung Hei Fong
6. Kurt Resayo

Milestone: Milestone 2

Date: 2021, October 28

History:

Version	Date
M2V1	2021, October 28

Table of Contents

1. Executive Summary	2
2. List of main data items and entities	3
3. Functional Requirements	5
4. UI Storyboards for each main use case	7
Use Case 1:	7
Use Case 2:	10
Use Case 3:	12
Use Case 4:	14
Use Case 5:	17
5. High level Architecture, Database Organization summary only	20
6. Identify actual key risks for your project at this times.....	22
7. Project management.....	23

1. Executive Summary

It is challenging to decide on any particular website among the various online tutoring services to help students excel in their coursework. Our products hope to provide the students of San Francisco State University (SFSU) with a better alternative. Our team aims to provide a simplified experience for those who choose to use our product. This experience would not be an overload of information or too broad in material to the users. By currently only focusing on the students here at SFSU, we hope to make it easy to tutor and find tutors. This specialization makes it easier to find the course they need help in and not some subject overview. Other services may only provide a wide range of assistants in the broad understanding of math or science. Still, our team hopes that our product "**TwoHelpYou**" can provide pinpoint assistance on their particular course at SFSU.

Our team's decision to name our product "**TwoHelpYou**" is also our goal. We can provide two parties with assistance, students past and present, to help other students. Our team hopes to only employ students and graduates from SFSU who have bravely pathfinding the courses offered at SFSU for tutoring and are willing to share their experience. By having students who have taken the course be tutors, we can provide other students seeking help with a unique experience. This experience would be more specialized and specific assistance that guides them down the path of success for that particular class that other services may not be able to provide. Such as the specific material covered in the syllabus and the style or method sought by the professor. Due to narrowing our audience of users, we can contract and simplify the search to make it easy for students to find help in the course they are taking. At the same time, it will easily allow tutors to sign up for which classes they are qualified to tutor by listing what courses they have completed. By employing students and graduates, we can also proudly support and give back to our community which is the dream of all alumni.

As students here at SFSU, we like to say we know the needs of our community because we are a part of it. Being part of this community with the ability to give back, we hope to connect students. By connecting the students, we can help them by providing a job or assistance in their classes. Hosting a platform for them to connect, we create a positive feedback loop of former and current students assisting one another. Our mission statement is to provide a platform and service for students to connect and help other students.

2. List of main data items and entities

1. Unregistered User:

- Unregistered users are the users who can view the contents of the website. However, these users don't have permission to schedule tutoring services.
- To schedule appointments or post questions, unregistered users are required to register

2. Registered User:

- Registered users can view the web page's contents, schedule appointments with tutors, and post questions in the forum.
- To schedule or post questions, registered users are required to login
- Required Data:
 - first_name: First name of the registered user created at the time of registration
 - last_name: last name of the registered user created at the time of registration
 - email: Valid SFSU email
 - password: Password created at the time of registration
 - username: Username created at the time of registration
 - usertype: Type of user could be tutor, student, or admin
 - photopath: Relative path of the photo

3. Admin:

- Admin is the registered user who has complete system control of the website
- These users have the privilege to add tutors

4. Tutor Posting:

- Posting is a service that a website provides where tutors can provide their posts for tutoring service courses.
- Required data:
 - tutorname: Tutor name to be displayed on the website
 - courses: Courses offered by the tutor
 - date: Date tutor is available for tutoring
 - time: Time tutor is available for tutoring

5. Registration Form:

- Registration form can create an account for the new unregistered user.
- Required Information:
 - first_name: Legal First Name
 - middle_name: Optional middle name
 - last_name: Legal Last Name
 - username: Unique username
 - email: Valid SFSU email
 - password: Character must have 8 or more characters, including at least one uppercase and one special character

6. Tutor Profile:

- Profiles are created for tutors with their status of availability
- Profiles can have reviews and rating
- Required Data:
 - First Name: First name of Tutor
 - Last Name: Last name of Tutor
 - Reviews: Reviews given to the tutor by the registered user
 - Rating: Rating given to the tutor by the registered user
 - Photo: Optional photo of the tutor

7. Tutoring Schedule:

- Tutor Schedule will be displayed on the tutor profile page.

8. Forums:

- Forum is a place where students and tutors can post questions
- Registered users can log in to post questions or reply to the post
- Required data:
 - title: Title of the question posted by a registered user
 - comment: Comments on the post of the registered users
 - date_time: Record of date and time of comment and question in forum

9. Search:

- Users can search through the list of available tutors
- Users can search through the list of available courses tutors are offering
- Users can search through the list of forums

10. Message:

- Message can be sent from one registered user to another registered user
- Required Data:
 - message_id: message will have its own unique id
 - message: Message is the text sent and received by the registered user.
 - user_id: Message is associated with registered user

11. Comment:

- Registered users can comment on the forum.
- Required Data:
 - comment_id: comment has its own unique id
 - forum_id: comment is associated with the forum where the registered user has already posted questions or concerns.
 - comment: comment posted by the registered user when they answer in forum
 - user_id: comment is associated with the registered user

12. Category:

- Posting of tutors will have category and users can search through desired category
- Required Data:
 - category_id: Category has its own unique id
 - post_id: Category is associated with post_id where the post will be categorized by class name
 - category: List of category post belongs to

3. Functional Requirements

Priority 1:

Unregistered User

1. Unregistered user shall be able to create an account with an SFSU email
2. Unregistered user shall be able to search for a tutor using course number
3. Unregistered user shall be able to search for a tutor using major name
4. Unregistered user shall be able to search for a tutor using the tutor name
5. Unregistered user shall be able to look at the forum
6. Unregistered user shall be able to view tutor profile

Registered User

7. Registered user shall be able to use all functions from Unregistered User
8. Registered user shall be able to login with their SFSU email
9. Registered user shall be able to log out
10. Registered user shall be able to schedule a meeting with a tutor
11. Registered user shall be able to apply for tutor privileges
12. Registered user that is a tutor shall be able to set up their profile
13. Registered user that is a tutor shall be able to set up a schedule that users can see

Administrator

14. Administrator shall be able to use all the functions from Registered User
15. Administrator will have access to database and administrative tools

Priority 2:

Unregistered: User:

16. Unregistered user shall be able to search a post using the course number
17. Unregistered user shall be able to search a post using major name
18. Unregistered user shall be able to search a post using tutor name
19. Unregistered user shall be able to view a post that another student creates

Registered User:

20. Registered user shall be able to set up a group study session with other users and a tutor
21. Registered user shall be able to send a message to another registered user
22. Registered user shall be able to post to the forum
23. Registered user shall be able to report a user if they encounter an unruly registered user and the report will be sent to the admin.

Priority 3:**Registered User:**

- 24. Registered User shall use website tools during a virtual session with a tutor
- 25. Registered User that is a tutor shall be able to respond to student post to answer question
- 26. Registered User shall be able to answer a forum post
- 27. Registered User shall be able to search up certain forum posts

Administrator

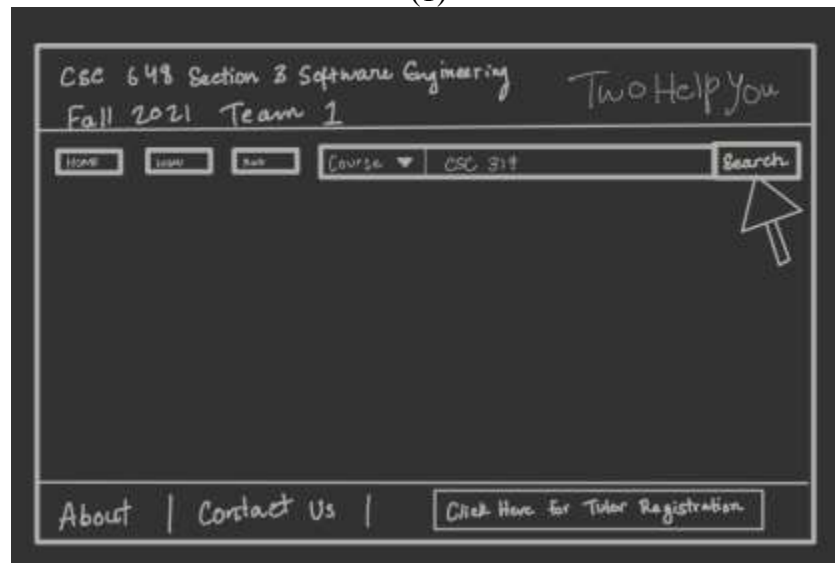
- 28. Administrator shall be required to approve post
- 29. Administrator shall be able to delete post
- 30. Administrator shall be required to approve tutor
- 31. Administrator shall be able to suspend accounts
- 32. Administrator shall be able to delete accounts

4. UI Storyboards for each main use case

Use Case 1:

Mark needs help with his web development homework after hours. (1) He uses the search function on TwoHelpYou to find his course. (2) Using the search results, he selects a tutor whom he likes. (3) Mark clicks on the schedule session button to set up a session with Arthur (tutor). (4) He fills out the form and clicks submit. (5) He is prompted to log in. (6) He fills out the registration form and clicks submit.

(1)



(2)



(3)

CSC 648 Section 3 Software Engineering
Fall 2021 Team 1

Two Help You

Home Login Join Courses ▾ CSC 317 Search

😊 Arthur C. ★★★★★ Schedule Session

Courses

Qualifications

Profile

Forums

About | Contact Us |

(4)

CSC 648 Section 3 Software Engineering
Fall 2021 Team 1

Two Help You

Home Login Join Courses ▾ CSC 317 Search

Tutor Scheduling Form

Subject Course Number ▾

Email / contact Info

Session length: Duration ▾

Preferred Time: Time ▾

Extra Info / Questions

Submit

About | Contact Us |

(5)

CSC 648 Section 3 Software Engineering
Fall 2021 Team 1

Two Help You

HOME LINK FOR COURSE CSC 317 Search

Subj
Em
Se
Pre

Email
Password
Don't have an account?
Register

LOGIN

Extra INFO / LIVES LIONS

Submit

About | Contact Us |

(6)

CSC 648 Section 3 Software Engineering
Fall 2021 Team 1

Two Help You

HOME LINK FOR COURSE CSC 317 Search

Your scheduling request to Arthur has been
successfully submitted.
The tutor will contact you to
confirm the session

About | Contact Us |

Use Case 2:

Anish wants to sign up as a tutor on TwoHelpYou. (1) He clicks the registration link on the home page. (2) He fills out the registration form and clicks submit. (3) He is prompted to log in, and he does so. (4) Anish receives a confirmation that his application has been submitted.

(1)

CSC 648 Section 2 Software Engineering
Fall 2021 Team 1

Two Help You

HOME LINK TAB All Search

About | Contact Us | Click Here for Tutor Registration.

(2)

CSC 648 Section 2 Software Engineering
Fall 2021 Team 1

Two Help You

HOME LINK TAB All Search

Tutor Registration

First Name Last Name

Email

Qualification
1. Degree
2. Diploma
3. Certification

Skills / Expertise
C++
Java
JavaScript
PHP

Upload Resume PDF / PNG Submit

About | Contact Us |

(3)

Hand-drawn wireframe of a web application interface. The header contains the text "CSC 648 Section 3 Software Engineering Fall 2021 Team 1" on the left and "Two Help You" on the right. Below the header is a navigation bar with buttons for "HOME", "Login", "Register", and a search bar labeled "Search". The main content area is titled "Tutor Registration" in the center. On the left side of this area, there is a vertical sidebar with buttons for "Email", "Password", and "Forgot Password". The central area contains a "LOGIN" button, a "Register" button, and the text "Don't have an account?". Below the central area, there is a "Submit" button. At the bottom of the page, there is a footer with links for "About" and "Contact Us".

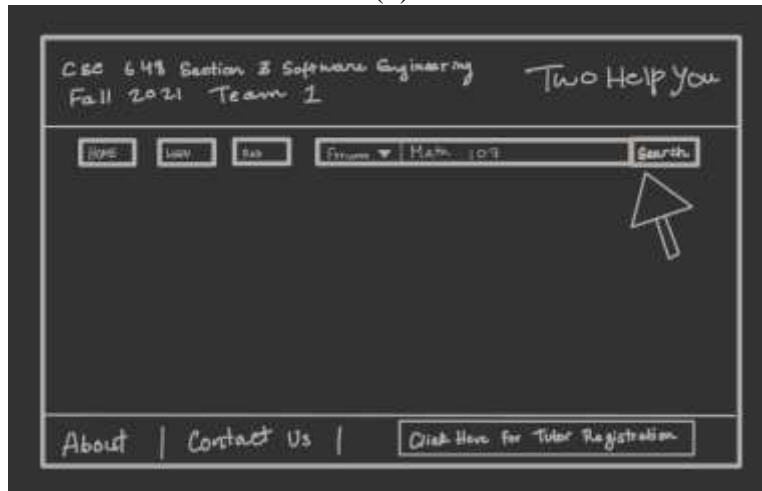
(4)

Hand-drawn wireframe of a web application interface showing a confirmation message. The header contains the text "CSC 648 Section 3 Software Engineering Fall 2021 Team 1" on the left and "Two Help You" on the right. Below the header is a navigation bar with buttons for "HOME", "Login", "Register", and a search bar labeled "Search". The main content area displays the message "Your application has been successfully submitted." in a large, centered font. At the bottom of the page, there is a footer with links for "About" and "Contact Us".

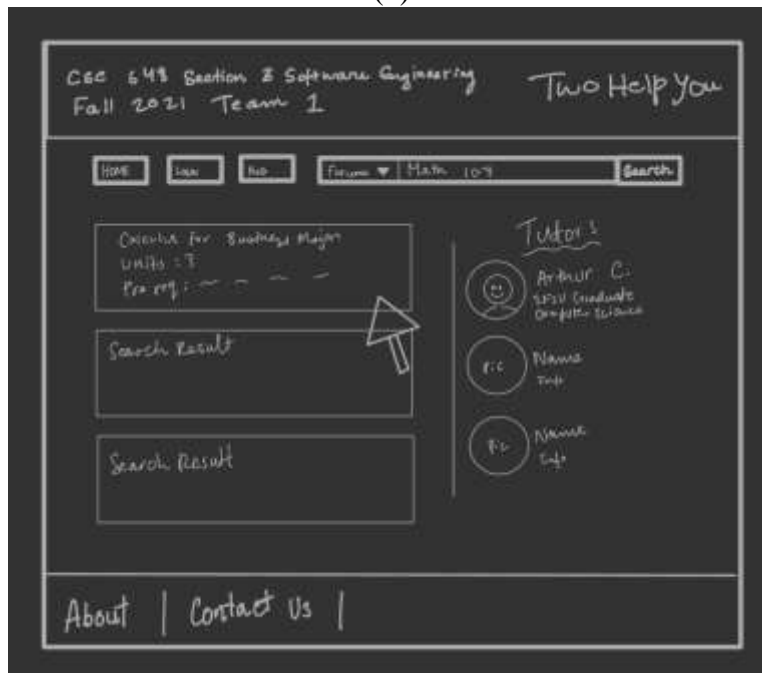
Use Case 3:

Calvin needs help on one of his Math 107 homework questions. (1) He uses the TwoHelpYou search function to search for forums related to his class. (2) He is presented with search results that contain forums related to math 107, and he selects one that fits his needs. (3) He is taken to the forum page to see an answer posted by a verified tutor.

(1)



(2)



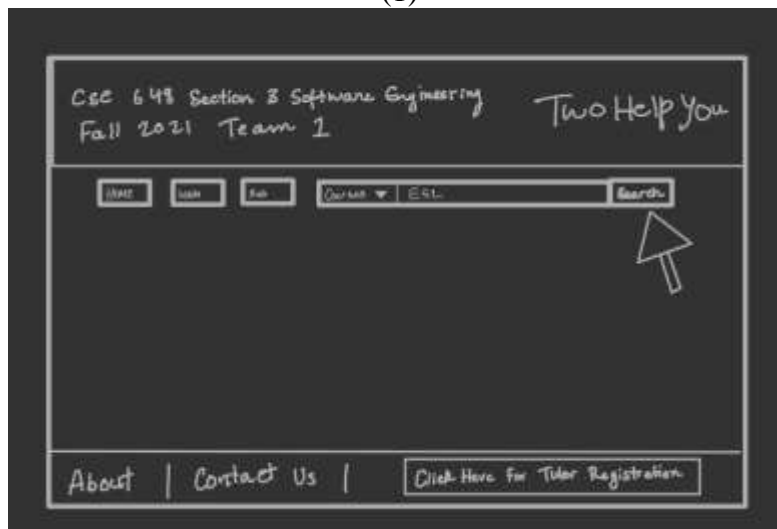
(3)

CSC 648 Section 2 Software Engineering Fall 2021 Team 2		Two Help You	
Home	Index	Post	Forum 107 Math 107 Search
Forum Title		Post P.n	
Author	Post Content / Question		
Responses			
name	Content / Answer		Verified Tutor ✓
name	Content Answer		
About Contact Us			

Use Case 4:

Salome needs help with her English course. (1) She uses the search function on TwoHelpYou to find her course. (2) Using the search results, she selects a tutor whom she likes. (3) Salome clicks on the schedule session button to set up a session with Jake (tutor). (4) She fills out the form and clicks submit. (5) She is then prompted to log in, and she does so. (6) She is given a confirmation that her request has been sent.

(1)



(2)



(3)

CSC 648 Section 2 Software Engineering
Fall 2021 Team 1

Two Help You

HOME LINKS TAB COURSE ▾ CSCI Search

Profile: Jake Bourdain ★★★★★ Schedule Session

COURSES

Qualifications

Products

FORUMS

About | Contact Us |

(4)

CSC 648 Section 2 Software Engineering
Fall 2021 Team 1

Two Help You

HOME LINKS TAB COURSE ▾ CSCI Search

Tutor Scheduling Form

Subject Course Number ▾

Email / Contact Info

Session length: Duration ▾

Preferred Time: Time ▾

Extra Info / Questions

Submit

About | Contact Us |

(5)

CSC 648 Section 3 Software Engineering
Fall 2021 Team 1

Two Help You

HOME LOGIN REG Courses ▼ ESL Search

Email

Password

LOGIN

Don't have an account?

Register

Submit

Extra LHD / LIVES LIONS

About | Contact Us |

(6)

CSC 648 Section 3 Software Engineering
Fall 2021 Team 1

Two Help You

HOME LOGIN REG Courses ▼ ESL Search

Your scheduling request to Arthur has been
successfully submitted.
The tutor will contact you to
confirm the session

About | Contact Us |

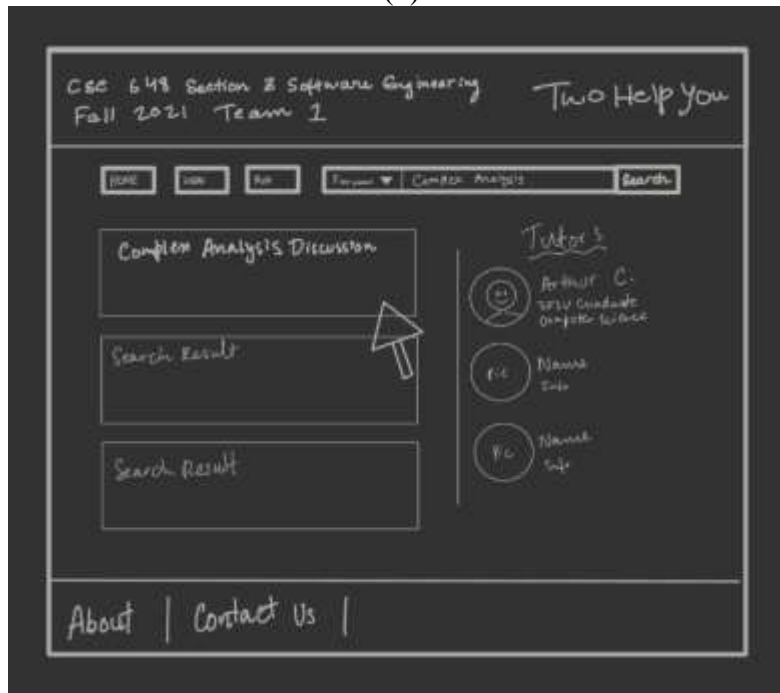
Use Case 5:

Theodora wants to participate in extracurricular discussions related to Complex Analysis in her free time. (1) She uses the search function of TwoHelpYou to look for forums related to Complex Analysis. (2) She is served several search results, which are forums that relate to Complex Analysis. She selects one she thinks is interesting. (3) She is taken to the forum page, where she clicks the post reply button to contribute to the discussion. (4) She is then prompted to log in, and she does so. (5) Theodora can now use the text box and post reply button to post her response to the forum.

(1)



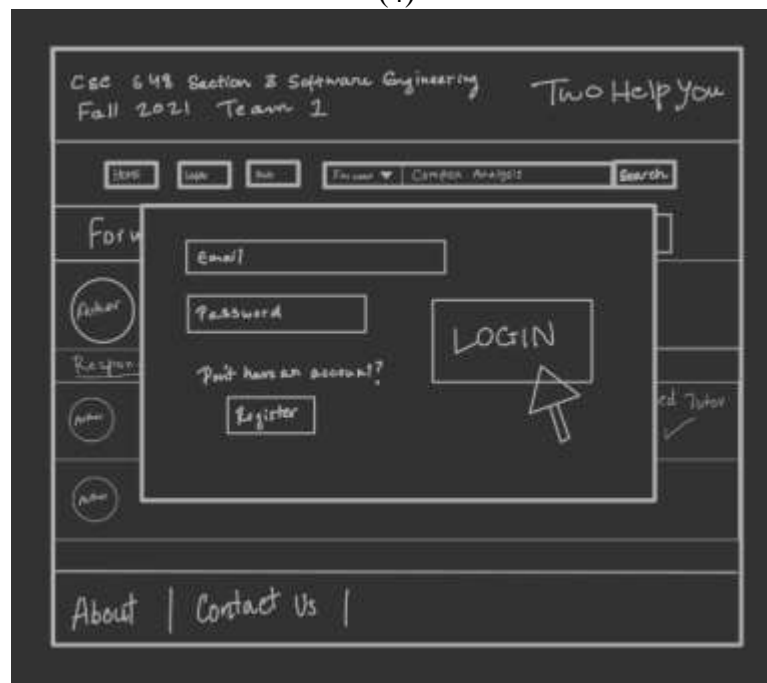
(2)



(3)



(4)



(5)

The wireframe depicts a forum interface with the following components:

- Header:** Contains the text "CSC 648 Section 2 Software Engineering Fall 2021 Team 1" on the left and "Two Help you" on the right.
- Navigation/Filtering:** A row of buttons labeled "HOME", "LINK", and "LOG", followed by a search bar containing the text "Forum" and "Complex Analysis", and a "Search" button.
- Section Header:** A large box labeled "Complex Analysis" with a "Post Reply" button to its right.
- Post Content:** A post by a user named "Akhil" with the title "The mathematics of imaginary numbers".
- Responses Section:** A section titled "Responses" containing a table with two columns: "Content / Answer" and "Verified Tutor".
- Form Fields:** Below the responses table, there is a "Text" input field with placeholder text "...", a "Post Reply" button, and a mouse cursor icon pointing at the button.
- Footer:** A row of links labeled "About" and "Contact Us".

5. High level Architecture, Database Organization summary only

Main Database Schema:

1. User table
 - id (primary key)
 - first_name
 - last_name
 - username(unique)
 - email(unique)
 - password
 - active
 - created
 - usertype
 - photopath
2. Post table
 - post_id (primary key)
 - date
 - time
 - course
 - major
 - user_id (foreign key)
3. Forum table
 - forum_id (primary key)
 - title
 - comments
 - date
 - time
 - user_id(foreign key)
4. Message table
 - message_id (primary key)
 - message
 - user_id (foreign key)
5. Reviews table
 - review_id(primary key)
 - review
 - rating
 - user_id (foreign key)
6. Category table
 - category_id(primary key)
 - category
 - post_id (foreign key)

7. Comment table

- comment_id(primary key)
- comment
- forum_id (foreign key)
- user_id (foreign key)

Media Storage: The uploaded image by the users will be stored in a static folder. Database will be storing the relative path to the image.

Search: Using LIKE query, we can match part of the full data present in a column. Along with wildcards like % in different combinations, we can match our keyword with the pattern of the data present in columns.

Rating: Algorithm to find the average of the rating from the users will be used.

Reviews: Reviews will be sorted by the date created from newest to oldest.

6. Identify actual key risks for your project at this times

Skill Risks:

- The team needs to learn the API: Utilization of APIs required like Google Analytics API will require time to learn and implement despite being ensured that the tools are easy to learn and use. The best way to learn how to use the API is to practice implementing it; currently, we are not at the stage that requires its usage.
- Team members need to familiarize themselves with frontend tools: Needing to refresh and learn new frontend tools like Handlebars and React requires actual usage and implantation that has not taken place yet due to not needing to utilize the tools to improve the website. As the team begins to build the website further, the team will remember or reach out to appropriate sources of knowledge to work on the frontend development.
- The team needs to work with React: Many team members are unfamiliar with React outside of those utilizing it in another class. Finding good sources and reaching out to known colleagues familiar with the tool would be the best approach to improving and using it for frontend development.

Technical Risks:

- CPU/Server utilization using Amazon's free tier of AWS: Using the free tier of AWS has risk due to limitations imposed. Exceeding the limits will charge fees on the account. If the fees remain unpaid can result in account suspension. The careful monitoring of the account is the best approach to ensure that the account remains active.
- Github usage when working on multiple parts in parallel: The risk of merge collisions is ever prevalent when working with various branches and individuals on any project. By carefully testing the branches before and after each merge by leads and Github master heavily reduces issues. This ensures that the Main branch always remains operational and stable.

Team Risks:

- Frontend and backend communication on the connection between UI with database: Naming variables to populate or generate database items requires strong communication between the two leads and unification within the respective ends to ensure uniformity. Constantly reviewing team member codes and adhering to a uniformed naming convention and frequent communications between leads can significantly reduce issues.
- Pacing of work and members so that workflow is continuous: The delegation of work and tasks in a way that works is not in a state of waiting due to requiring a task to have completed. By establishing and enforcing internal deadlines enormously decrease tasks gridlock.

7. Project management

To ensure the successful delivery of Milestone 2 and the remaining project as we advance, the team has incorporated the usage of Asana to keep track of internal deadlines and assigned tasks. The team has successfully utilized the tool in Milestone 1 to ensure the timely completion and submission of Milestone 1.

Aside from using Asana to track tasks and internal deadlines, the team holds meetings utilizing Discord and Zoom as platforms. The Team Lead, Frontend Lead, and Backend Lead meets every Friday to coordinate tasks and communicate any questions or concerns. The Frontend Lead and Backend Lead meet up with their respective development members to work and coordinate when necessary. All meetings that take place have minutes taken and posted for all members to inform everyone what is going on. Aside from the scheduled meetings, other meetings are held, when necessary, primarily using Discord.

Outside of team meetings, the team has chosen to utilize Discord as an instant messaging platform to communicate freely. The use of the platform allows the team to post announcements, questions, updates, and most importantly, keep everyone up to date with the project.

The team has agreed to the Github Master's repository management with communications with the Leads' needs. This management style revolves around having two primary branches, which are the Main and Development branches. Aside from these two branches, new branches are constantly being created and deleted as needed for development. No members are to code onto the two primary branches directly. Every sub-branch designed for development will be tested and reviewed, then merged into the development branch, where it will be rechecked prior by another team member. Only when the development branch is stable, and it is deemed necessary to be merged into the main branch where it will be tested before pushing to the AWS EC2 instance.