Advanced Software Engineering and Industrial Practices

SFSU CSC Help

CSC 890 – Team 02

Prathiba Ramesh - Frontend Lead
Anudeep Katukojwala – Backend Lead

"Milestone 2"

October 17, 2022

Revision History Table

Revision ID	Revision Date	Revised By

Table of Contents

Short Description about 'SFSU-CSC-Help' Usage and it's working:

Link to Milestone 01 Document	Link to SFSU-CSC-Help Web Application
CSC890_Team02_Milestone1	http://34.125.120.197/

- This app allows SFSU CS department students to get instant advising on various subject matters that help them succeed in their field.
- Students will have all their questions answered instantly by AI powered chat bot in the language of their choice.
- This process will ease the work and reduce the effort of both department faculties and the students.

The following requirements detail the essential functionality required in our web application. Each requirement has a number associated with it in the closed bracket which indicates the priority level that functional requirement has.

1. Functional Requirements V2 Priority Level and code:

1- Must Have
2- Desired
3- Opportunistic

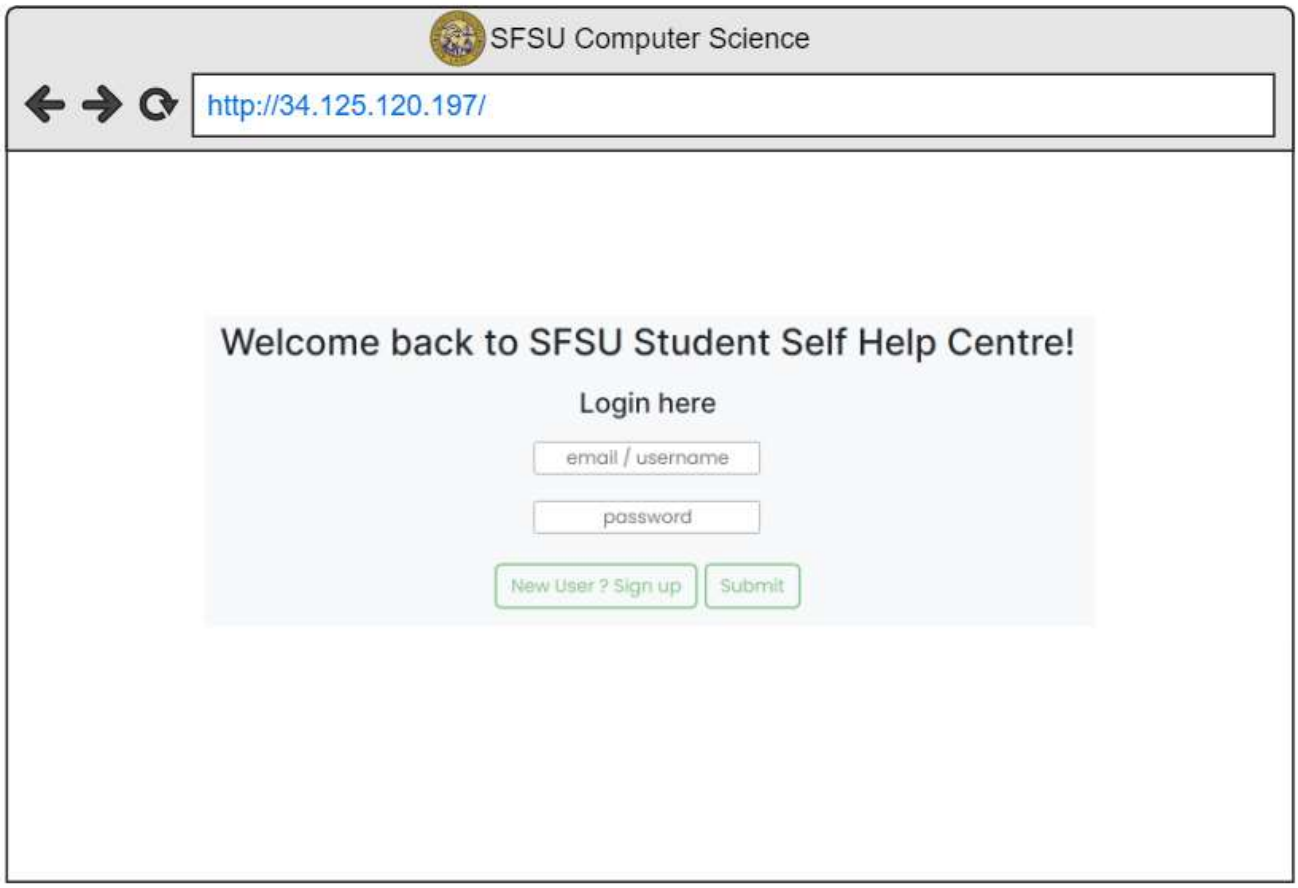
Function	Sub-Function	Priority
Account Management	*Registration & Login:* Users shall be able to register their accounts with the app, log into and log out of their accounts to gain access and deny access from the app, and manipulate their profiles to give their basic information, and general information on their academics which shall be used to provide suggestions	2
Email Confirmation	Application shall require users to verify their email address upon registration.	1
Administrator Capabilities	Website shall be able to trigger the matching alerts to the corresponding users.	2
Advisor List	Based on the user student id, list of advisors for various procedures will be given to the user upon request.	1
Prerequisite information	Users shall be able to get information for prerequisites of any CS course.	1
Course Plan	Based on the user profile and the semester they are in currently, a complete course plan or roadmap for their present and future semesters are provided, which is editable/customizable based on their own interest	1
Instructor details	Users shall be able to get the details of the instructor for the entered course number.	1
Deadlines information	Users shall be able to find deadlines for course enrollment, applying for ATC and other relevant procedures.	1

Specific to SFSU CS department students	Website shall be specific to SFSU CS department students looking for any advice or information about their department It shall minimize their struggle to schedule the advising session with their advisors for every information they may need.	1
Downloadable Links	Links to download various forms within the CS department such as ATC, PTE, course waiver, credits transfer, etc shall be given on request through web application.	1
Graduate Seminars	Users shall be able to register and get alerts for upcoming graduate seminars. Application shall send the users emails or phone notifications for the scheduled seminars.	3
Multilingual Conversation	Users can opt in changing the language from English to their preferred choice when using our AI powered chat bot.	3

2. High Level UI Mockups and Story Boards

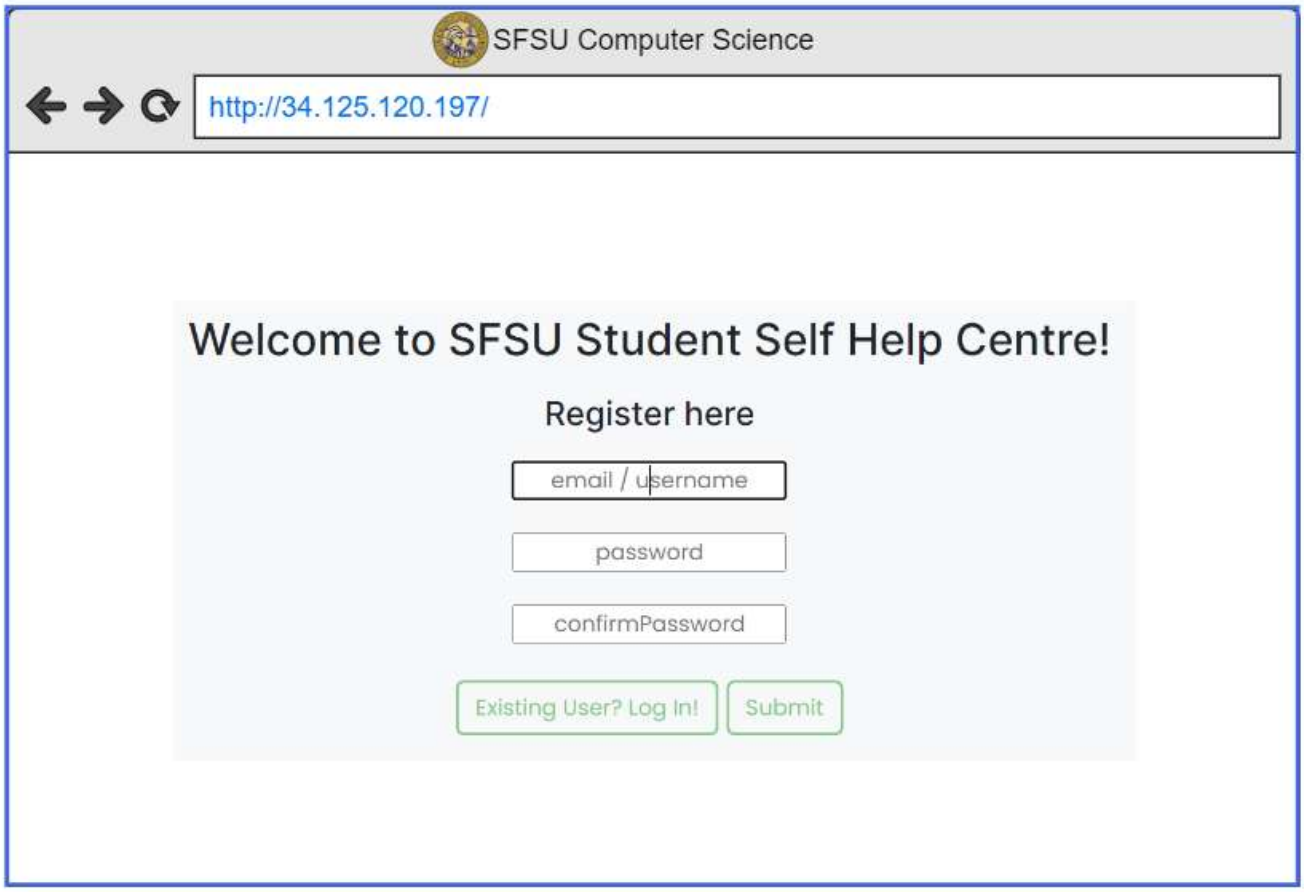
Login Page

- This is the page where users can log in to their account. If the user is in our database, then the page will redirect to the homepage. If not, will redirect to the registration page.
- req.body will have username and password



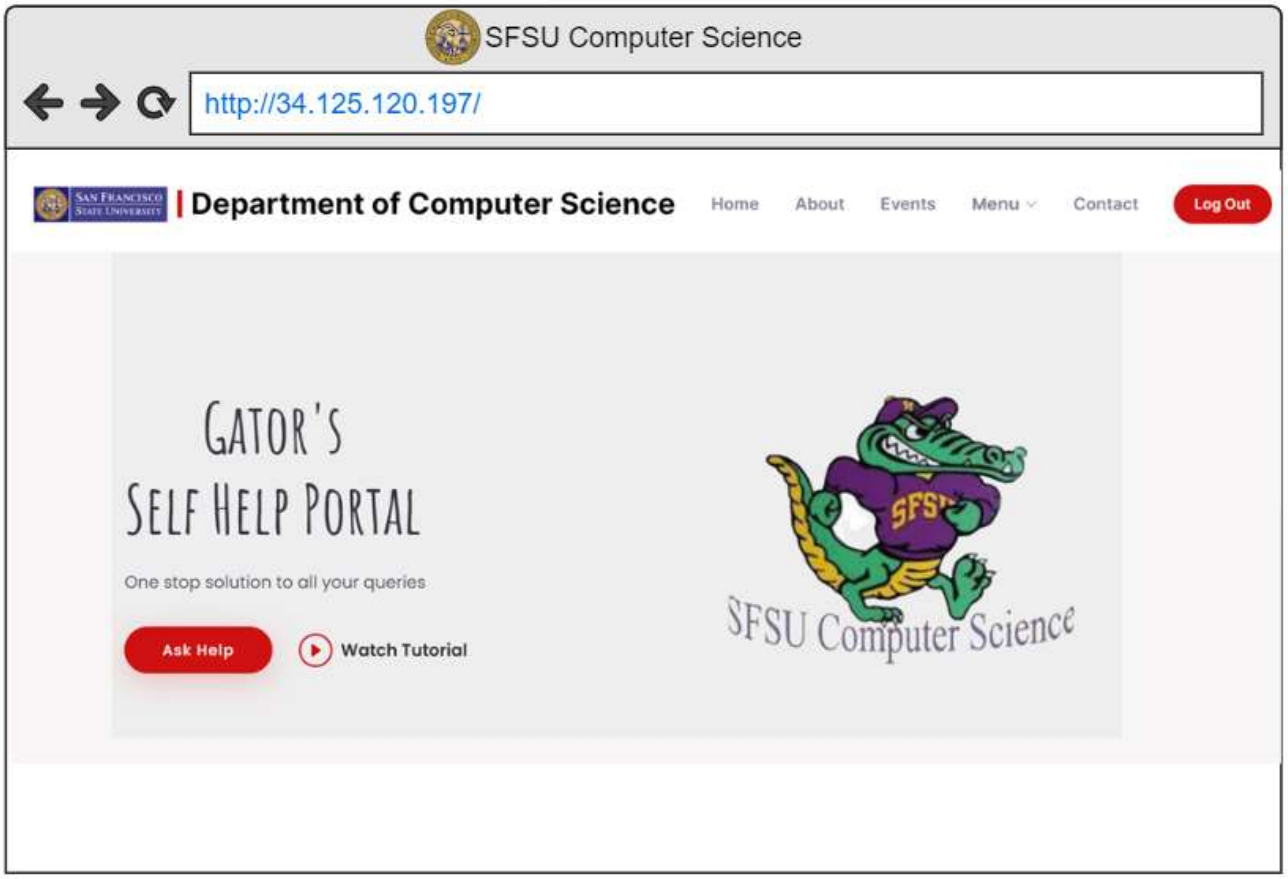
Registration Page

- This is the page where users can register for their account.
- req.body will have username, password, confirm password.



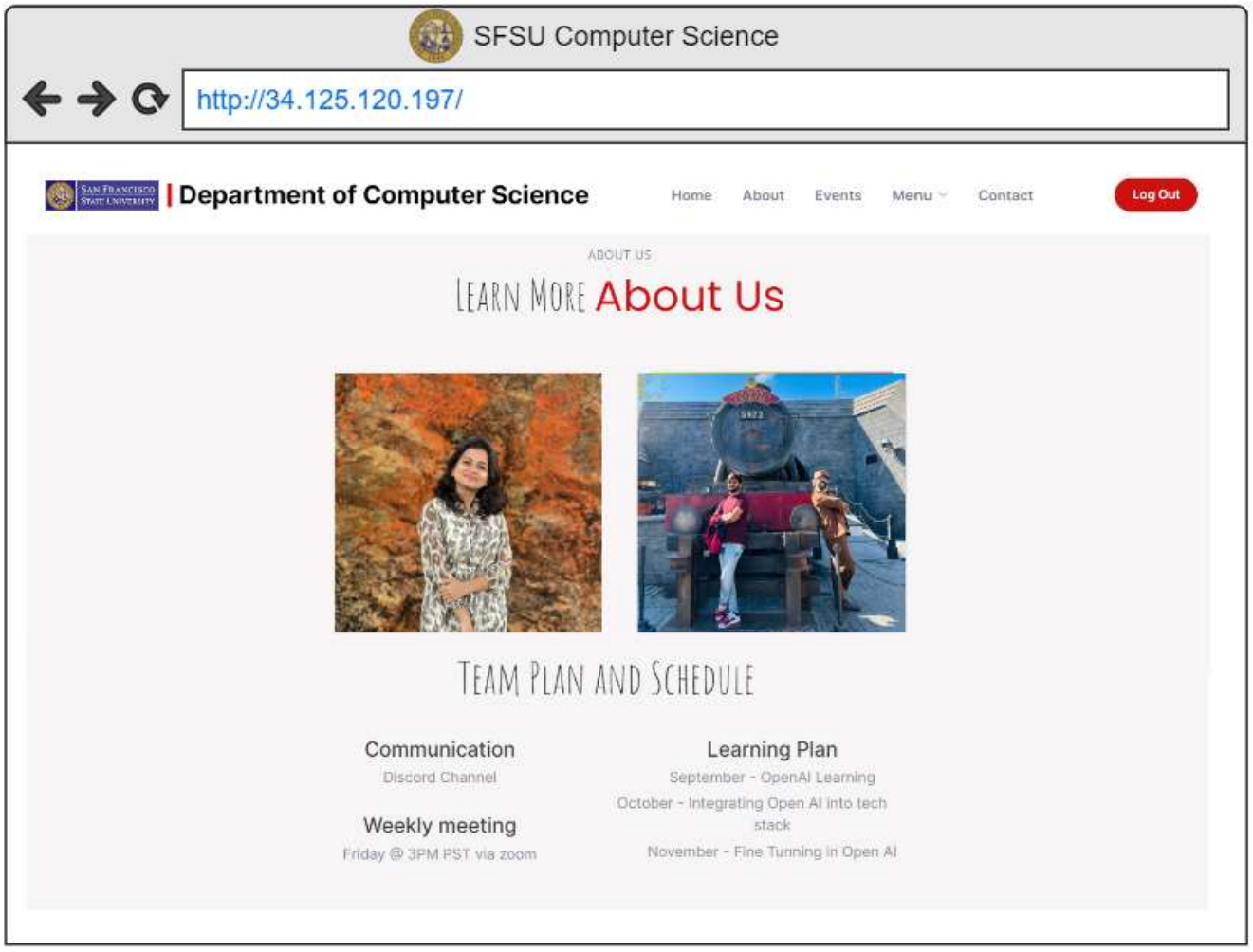
Home Page

This is the page where we share our mission and goals to the user. We will also talk about the application and link to quick help is available here.



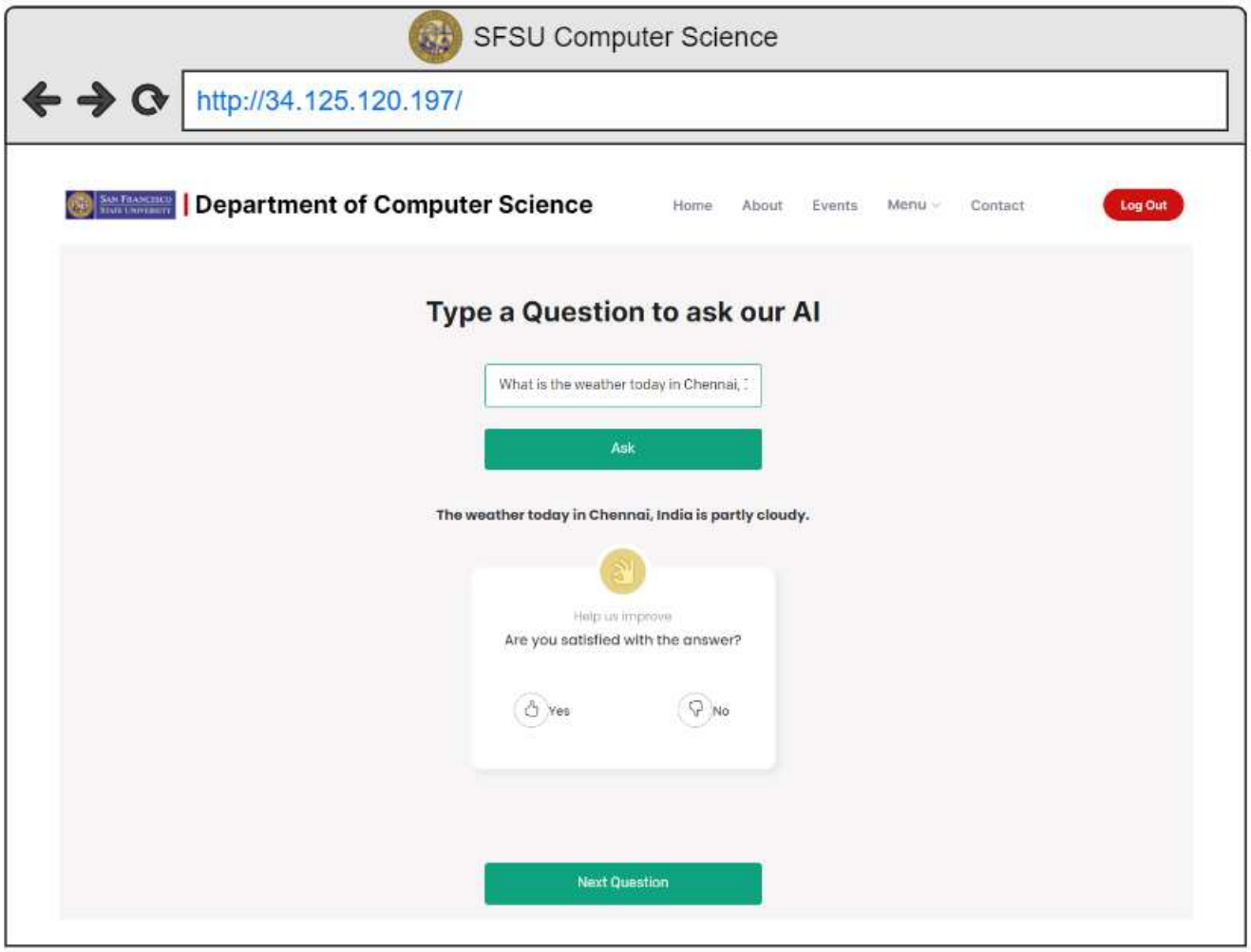
About Page

This is the page where we share app owner details and team plan, team schedule or study plan details.



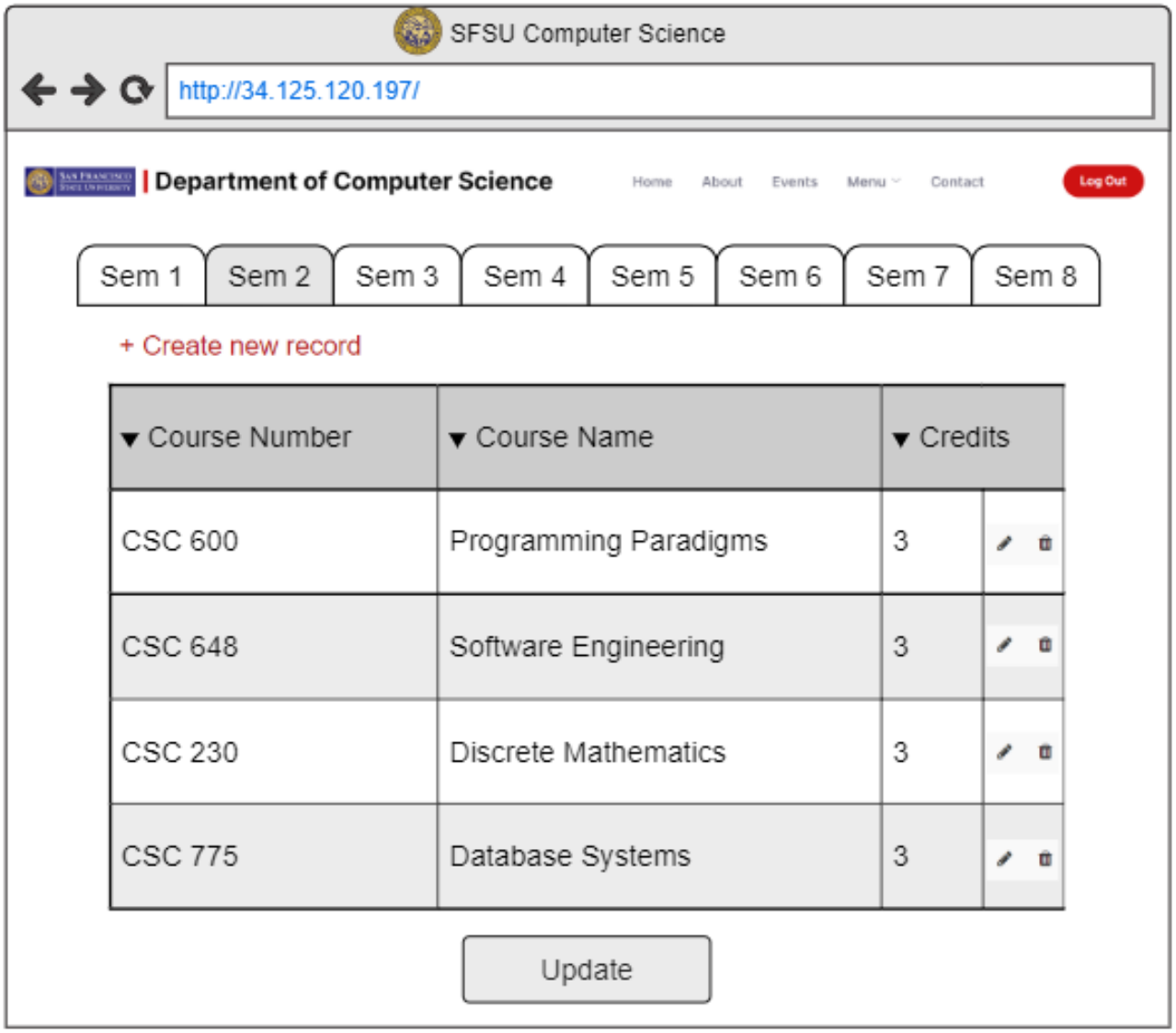
Help/ Q & A Page

- This is the page where SFSU CSC students will be able to ask their academic queries and get appropriate answers.
- They can also provide feedback to the answers they received.
- req.body will have question, feedback



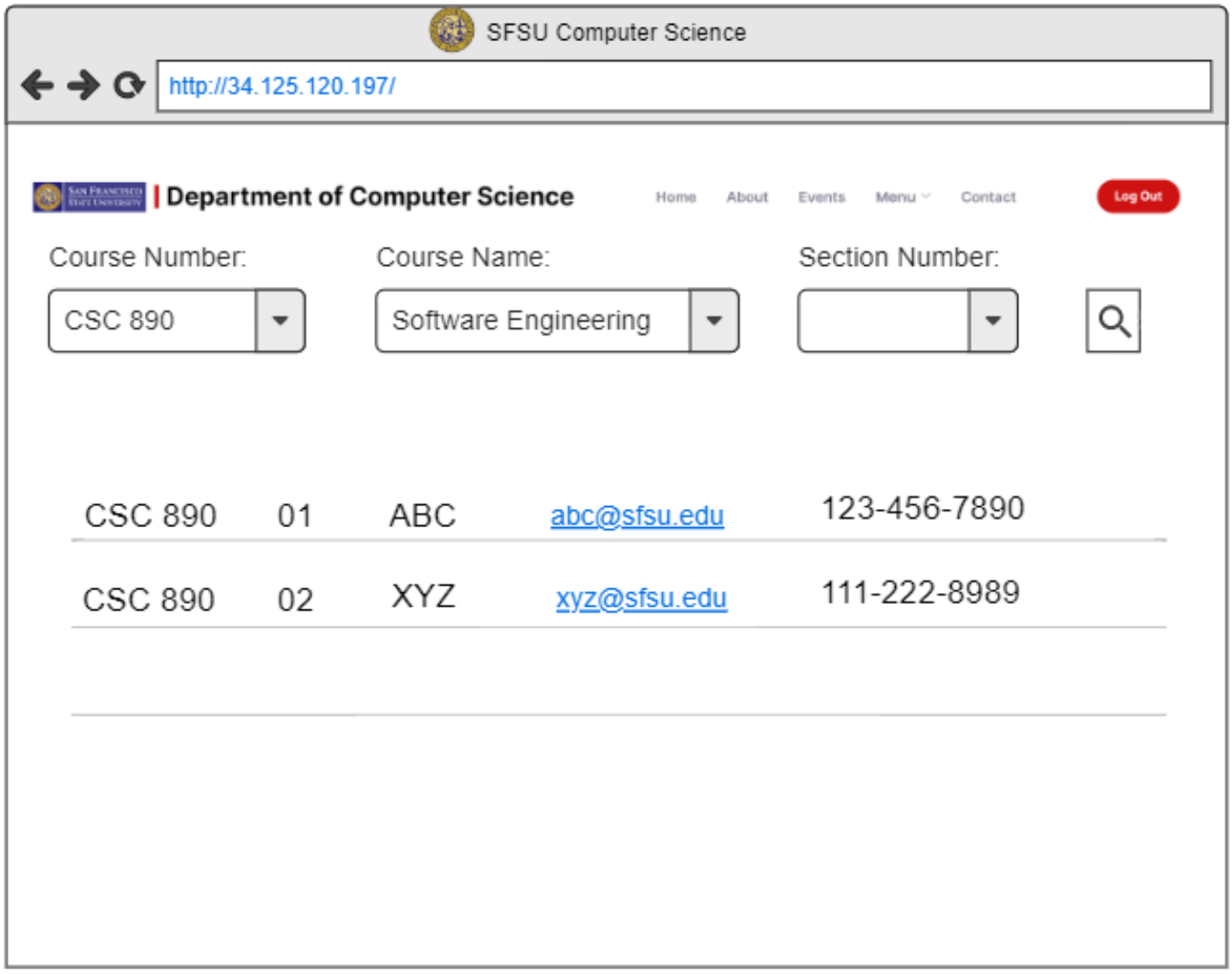
Course Plan

- This is the page where SFSU CSC students will be able to look at their roadmap or course plan for each semester (Available for both graduate and undergraduate students)
- They can add new row, edit the existing plan according to their need and delete a row as well. Changes will be updated in database.
- req.body will have course number, course name and credits



Instructor Details

- This is the page where SFSU CSC students will be able to search for instructor details based on the search parameters given
- They can search with course number, section number and filter/sort
- req.body will have course number, course name and section number



3. High Level Architecture, Database Organization

Third Party API (OPENAI - GPT-3)

- We are using the GPT-3 model (developed by OpenAI) as our API to generate answers (completions) based on the question provided (prompt).
- GPT-3 is the third-generation language prediction model developed by OpenAI.
- The necessary data is fetched through Post method to API call https://api.openai.com/v1/completions
- API key is fetched from OpenAI and used to connect to our API
- We are using the Q&A prompt from OpenAI in our application.
- There are different parameters to be set for API to work according to our requirement:
    - **model*: 'text-davinci-002'***
        - It is the most capable GPT-3 model. It responds to prompt more efficiently, quickly and supports inserting completions within text.
        - **Fine-tuned model: *'davinci:ft-personal-2022-10-16-15-36-02'***
            - We have fine-tuned the model based upon our use case and added them into API call.
              Note: Doesn't include all dataset. Only 1 functionality (deadline details for M2)
    - **prompt: *'Provide appropriate answers for the given question'***
        - The prompt(s) to generate completions for, encoded as a string, array of strings, array of tokens, or array of token arrays.
        - if a prompt is not specified the model will generate as if from the beginning of a new document.
    - **temperature: *'0.8'***
        - The temperature controls how much randomness is in the output.
        - In general, the lower the temperature, the more likely GPT-3 will choose words with a higher probability of occurrence.
    - **top_p: *'1'***
        - An alternative to sampling with temperature, called nucleus sampling, where the model considers the results of the tokens with top_p probability mass.
        - So, 1 means tokens comprising the top 100% probability mass are considered.
    - **max_tokens: *'100'***
        - This parameter sets an upper bound on how many tokens the API will return.

- **frequency_penalty & presence_penatly:** ***'0'***
  - The frequency and presence penalties found in the Completions API can be used to reduce the likelihood of sampling repetitive sequences of tokens.
- **Stop:** ***'\n'***
  - Up to 4 sequences where the API will stop generating further tokens. The returned text will not contain the stop sequence.

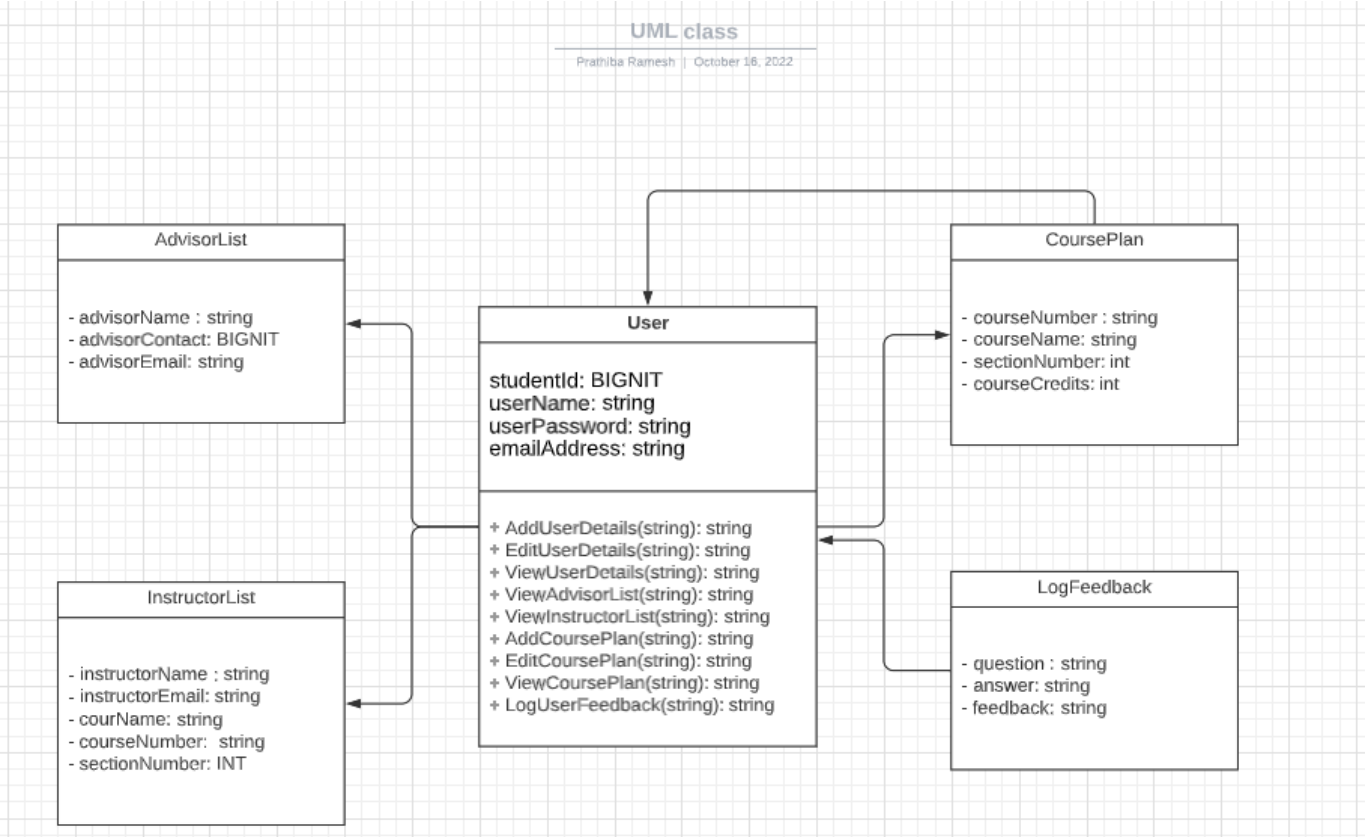
- Based on the parameters provided along with the API key to the OpenAI API call, response is provided, which we display in the UI.

DB Organization

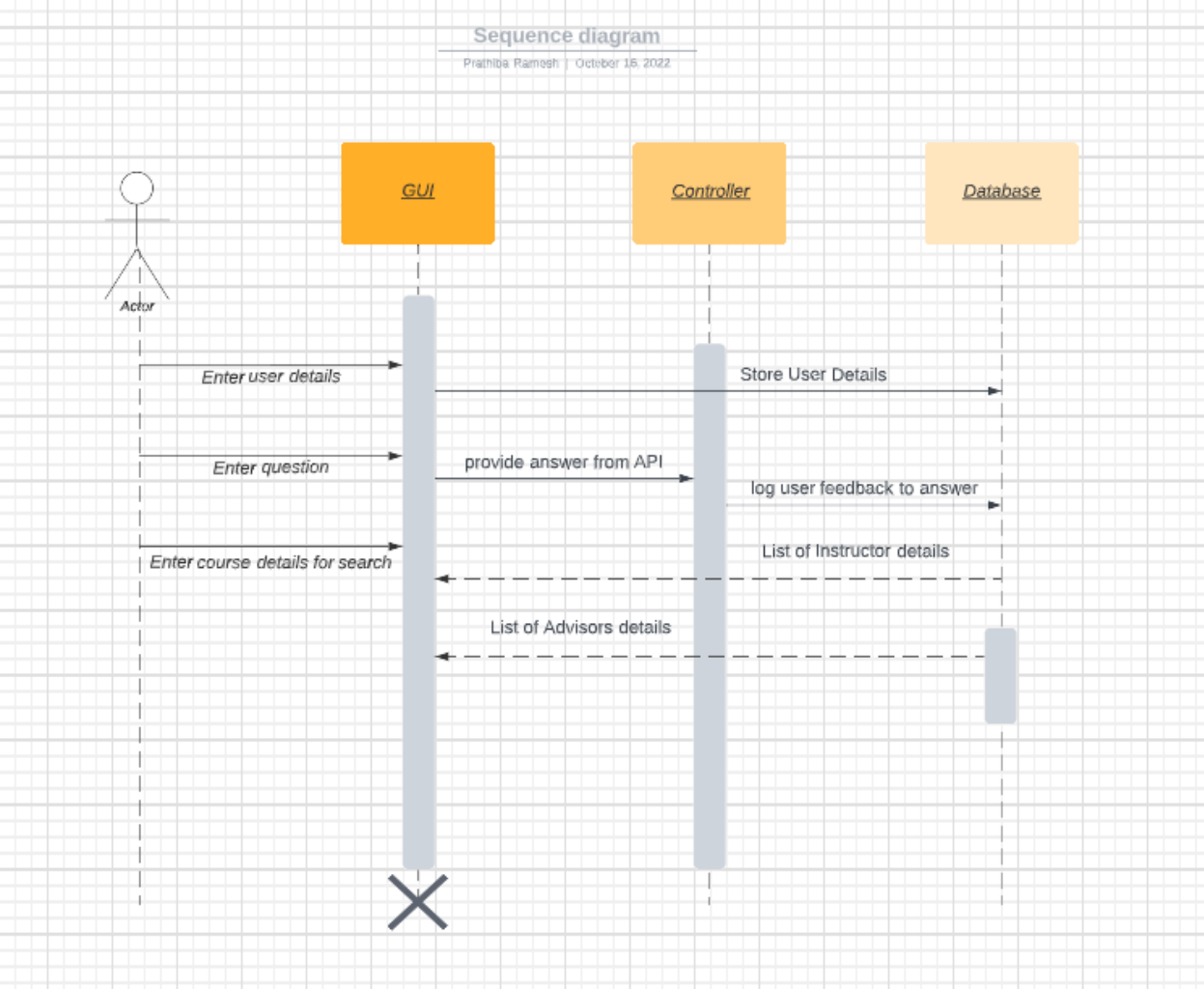
Table Name	Attribute Name	Data Type
users	studentId	BIGINT, Primary Key
	userName	NVARCHAR (50)
	userPassword	NVARCHAR (50)
	emailAddress	NVARCHAR (250)
userquestionlog	studentId	BIGINT, Foreign Key
	question	NVARCHAR (250)
	answer	NVARCHAR (500)
	feedback	NVARCHAR (50)
userCoursePlan	studentId	BIGINT, Foreign Key
	semesterNumber	INT
	courseNumber	NVARCHAR (50)
	courseName	NVARCHAR (250)
	courseCredits	INT
advisorList	studentId	BIGINT, Foreign Key
	advisorName	NVARCHAR (50)
	advisorContact	BIGINT
	advisorEmail	NVARCHAR (250)
instructorDetails	studentId	BIGINT, Foreign Key
	courseNumber	NVARCHAR (50)
	courseName	NVARCHAR (250)
	sectionNumber	INT
	instructorName	NVARCHAR (50)
	instructorEmail	NVARCHAR (250)

4. High Level UML Diagrams

UML Class Diagram



UML Sequence Diagram



Summary:

First time users will fill in their personal data which includes the username, id, password and email address. The details entered are validated and is stored and is sent to the user's specific database. After logging into our website users are able to seek help by asking their questions to our chatbot and after appropriate answers are provided, the feedback is logged by users, which will be sent back to database and stored. Users will also be able to fetch instructor details by searching with course number and they can even fetch their advisors list, which is auto populated based on their student id. This data is pulled from backend on request.

5. Identify actual key risks for your project at this time

Teamwork risk:

- Distributed the work among the team to complete vertical prototype during which we faced minor merge conflicts on integrating the frontend and backend part.
- It may take time for a team to study a new programming language, piece of software and the OpenAI API implementations.
- We are pushing ourselves hard to engage and learn new concepts in a scheduled manner and the knowledge is shared with the team during weekly meetings.

Schedule risk:

- While we both are working on the group project, it may be difficult to find the time for it. Both of us have more than one class and both of us work as well. This can cause some time management or scheduling issues.
- But we face this by maintaining deadlines and being responsible about these deadlines. These deadlines have been easier to face because we have been meeting early and consistently.
- We have chosen DISCORD as means of keeping track of planned items and our main means of project management tool.

Security risk:

- Security will always be a risk to the users, and we are considering looking more into the matter.
- Simple things such as making functions hide passwords or usernames from the inspect window or looking into using a VPC/VPN to protect our database will be good first steps into tackling this issue.
- Also, we are making sure that the API key is not exposed to the public.

Skill risk

- While we both are capable and ready to work on our project, some parts of this project are completely new to us as students.
- For example, using the Open AI API's to help us build the chatbot. We have different backgrounds and acknowledge this, so in order to help ease this issue we are doing a transparent style of communication where we discuss any roadblocks or skill gaps and listen to the other teammate suggestion, make plan on how to resolve it and move forward.
- Every week we take a day to read on Open AI to educate ourselves more how to use it in our web application.

Legal/Content Risks:

- This is something our team takes it seriously and try to read in detail about every API or software that we plan to use it in our software.
- Any content that we consume or any software that we use always goes through rigorous checks to avoid any legal issues we may encounter.
- We do this by keep strict guidelines of reading the privacy and legal notice page of every software that is under consideration to use in our application.