San Francisco State University
SW Engineering CSC 648/848 Fall 2022
Req Check
TEAM #6

Names	**Role**	**Emails**
Alex Sanchez	Team Lead	asanchez26@mail.sfsu.edu
Victoria Wilson-Anumudu	GitHub Master	vwilsonanumudu@mail.sfsu.edu
Syed Faiz	Back End Lead	sfaiz@mail.sfsu.edu
Eric Falk	Front End Lead	efalk1@mail.sfsu.edu
Vivek Santoki	Front End	vsantoki@sfsu.edu
Erik Rodriguez	Front End	erodriguez7@mail.sfsu.edu

Milestone 4

Milestone Version	Date
M4V1	December 01, 2022

Table of Contents

Product Summary

Function	Priority	Actors	Function
4	1	Application	The application should protect the encapsulated user data and disclose information based on roles.
43	1	Course	Each course shall be given a category of major, minor, GE, or elective.
48	1	Course	Each course shall have an enrollment status based whether a grade has been given or not.
1	1	Degree Planner	To use the degree planner feature, students must fill the appropriate subjects that they want to take for their degree.
37	1	Rec Checker	The Req checker shall be available on the WWW platform.
42	1	Requirement	Each requirement change shall update the overall course completion right away.
57	1	System	System will be available during all hours of the day including weekends and holidays.
25	1	System	The system shall display the course requirements for graduation.
26	1	System	The system shall differentiate courses by requirements and electives.
31	1	System	The system shall inform the student any prerequisites required prior to enrolling in a course.
46	1	System	The system shall give numerical status on the number of semesters left before graduating.
3	1	Timeline	The timeline shall depend on how much units a particular student must complete to obtain their degree.
18	1	Student	Students shall have a selection of departments they must choose from.
17	2	Admin	Admin shall add the prerequisite on a defined hierarchy of courses.
20	2	Admin	Admin shall have a list of all the courses.
21	2	Admin	Admin shall have a list of courses taken by a student.
6	2	Application	The application should automatically verify a student's academic records via the administrator.
7	2	Application	The application should be able to authenticate user identity by the university.

51	2	Course	Each course shall display the semester’s it is being offered for.
52	2	Semester	Each semester shall display the courses that can be taken prior to enrollment.
58	2	Student	Students shall be authenticated by the university to use the system.
28	2	System	The system shall update all courses based on students input data.
29	2	System	The system shall inform students of course exemptions.
32	2	System	The system shall update any prerequisite requirements.
34	2	System	The system shall enforce any prerequisites prior to enrolling in courses.
35	2	System	The system shall require satisfaction of prerequisites to enroll in courses.
36	2	System	The system shall give alternatives to satisfying prerequisites if applicable.
49	2	System	The system shall not allow students to enroll in courses they do not qualify for.
50	2	System	The system shall require students to complete prerequisites to enroll.
30	2	System	The system shall inform the student the number of semesters left to graduate.
14	2	Admin	Admin shall set new records and shall make a hierarchy of prerequisite courses.
15	2	Admin	Admin shall arrange all the courses department wise.
19	2	Course	Course shall have less dependency to any other entity.
45	2	Exemption	Each exemption shall be based on course requirements and student grade level.
53	2	Main Page	The main page shall display the user's schedule and classes with professor’s name, date, and class number.
12	2	Professor	A professor shall fill up a seat once a student had dropped the course.
10	2	Professor	Professors shall only approve a defined number of students to their courses.
11	2	Professor	A professor shall fill in dropped students’ seats using a waitlist of students.

22	2	Student	Students shall see the list of professors for a course.
13	2	Student	Each student shall have a chance to request enrollment and each enrollment seat shall be filled up by a first come first serve basis.
33	2	System	The system shall be notified if a course has not been passed.
23	2	System	The system shall display the student's grade level (current, transfer, new).
24	2	System	The system shall inform students of any changes in department requirements.
41	2	System	The system shall give a grade level status based on overall course completion.
27	2	System	The system shall notify students of all newly required courses.
47	3	Course	Each course shall have numerical options of perquisites.
44	3	Course	Each course shall be appointed based on a student's grade level (current, transfer, new).
8	3	System	System shall provide all information on passed courses, professors, grades, and sessions.
16	3	Admin	Admin shall change any courses/prerequisites as per the demand of the admin panel.
61	3	People	The amount of people shall be defined by the college administration.
5	3	User	The user (student) must be given information on whether a specific course is transferable from a certain school.
9	3	System	System shall send notifications of recommended courses and waitlisted courses for users to add.
59	0	N/A	
2	0	N/A	
60	0	N/A	
38	0	N/A	
39	0	N/A	
40	0	N/A	
54	0	N/A	
55	0	N/A	

56	0	N/A	

Title: ReqCheck

Since most colleges have their own course curriculums for their programs, students find it quite difficult to make these transfers due to discrepancies between their courses. It typically takes about 2 to 3 weeks to finalize/approve a transfer due to demand and a lack of cohesivity between school programs and course codes. This is why we built this software Req Check, where we typically solve this problem by strategically creating value to save time and provide knowledge to both students and school administrators on how to handle transfers.

The Req Check software takes one school's course code and matches it with the other schools' courses. An automated system like Req Check will efficiently map out when a student will graduate as well as ensure they are on the right track as a new transfer student. Req Check will alleviate stress for students and faculty as well as save time and money. At the end of the day, the goal is to help both parties and stakeholders plan and handle the transfers beforehand seamlessly.

URL to your product accessible to instructor, on deployment server:

http://ec2-52-53-211-193.us-west-1.compute.amazonaws.com/

Usability Test Plan

Task 1

Objectives

- We will be testing the ability of a user to plan their courses according to their degree requirements. The user should have an easy way to see exactly what requirements are needed for their degree, and they should be able to quickly switch out a course if there are multiple that satisfy a requirement. No matter what change is made on the courses page, it should always end up in a state that represents a valid degree plan. The page should be intuitive for the user, and it should show plenty of information without being overwhelming. The dialogs should be easy to navigate and leave the user with a clear indication of what they are doing.

Description	Change an editable degree requirement
State	The user must be logged in to be able to see the courses page. If not logged in, the user will be redirected to the home page.
Completion Criteria	A course title appears in place of “No Course Selected”
Intended Users	The courses page should only be accessible to students with registered accounts. This specific feature is useful for students who have degree requirements that can be satisfied by multiple courses.
URL	http://ec2-52-53-211-193.us-west-1.compute.amazonaws.com/courses

Task 2

Objectives

- We will be testing the ability of a user to search for any SFSU undergraduate course. The user should be able to easily search a course to view all relevant information about that course. The search bar should be responsive and never leave the user without feedback. When an ambiguous search is made, the user should be presented with a results page displaying all potential matches to the search.

Description	Search for a course at SFSU
State	User is on any page with the search bar accessible
Completion Criteria	The page displays information about a specific course
Intended Users	Any user who want to know more about a specific course
URL	http://ec2-52-53-211-193.us-west-1.compute.amazonaws.com

Task 3

Objectives

- For this function, a registered user will be able to access the roadmap webpage and be able to set up their course roadmap. How this will work is that the registered user will have already compiled their courses and will be able to list them in the semesters they wish to take in the future. This function will allow the registered user to visually see a roadmap of the courses they can take in the future, semester by semester.

Description	Use the roadmap to plan courses by semester
State	A student user should be logged in and on the roadmap page
Completion Criteria	Roadmap displays all courses organized into semesters
Intended Users	Students trying to plan their courses by semester. This page will not be accessible to non-registered users.
URL	http://ec2-52-53-211-193.us-west-1.compute.amazonaws.com/roadmap

Task 4

Objectives

- For this function, we will be testing to see if there are any course prerequisites to a chosen course. When a registered user is on the course webpage, they will be able to select courses and be notified of whether a course has any prerequisites. This function is purely informative and visual. For example, when a registered user is selecting their courses, and they select csc413, they will be advised that the course requires csc340.

Description	Check prerequisites for an upcoming course
State	For the system setup of the state, the user must be a registered user. If the user is not registered, then they will not see this webpage. If a user is registered, they will see the course webpage; they will see all the course requirements as well as whether a course is in progress or completed. Furthermore, each course section requirement sill allow the user to be given information on whether a course has prerequisites. This way, they will be notified if an upcoming course will require prerequisites.
Completion Criteria	When a registered user has chosen a course, they will be notified of any prerequisites. A pop up notification will be both informative and visual in order to emphasize the prerequisite requirement.
Intended Users	This will be only for registered users; a non-register user will not see this option.

URL	http://ec2-52-53-211-193.us-west-1.compute.amazonaws.com/courses

Task 5

Objectives

- For this function, we will be testing to see if courses from another school are equivalent to a course at San Francisco State University. If they are equivalent, then you will be given credit for the course taken at the other school. The reason why is if you are a transfer student, you should have taken previous courses that will have allowed you to transfer to SFSU in order to receive course credit for the Bachelor's Degree of Computer Science. This function will thus allow you to test whether a certain course is equivalent to a course at SFSU.

Description	Check a course equivalency; will test the equivalency of a course taken at another school to a current course at San Francisco State University.
State	For the system setup of the state, any user can check in to see the course equivalency. You can choose the school, and then choose the course applicable.
Completion Criteria	After choosing the courses, you will be notified if the course from the transfer school is equivalent to the SFSU course. If it is, you can add the SFSU course to receive credit.
Intended Users	This will be for both registered and unregistered users. You can access the course equivalency website to find an equivalent course.
URL	http://ec2-52-53-211-193.us-west-1.compute.amazonaws.com/equivalencies

Usability Metrics: Effectiveness

Task	% Completed	Errors	Comments	% Time Completed
Change an editable degree requirement	80%	Course didn't change after selecting	The pencil icons made it very easy to distinguish and alter the editable degree requirements.	
Search for a course at SFSU	0%	Pressing enter or clicking the search icon does not do anything.	The search bar location is ideal. The prompt text is very helpful.	
Use the roadmap to plan courses by semester	0%	Page under construction.	Does not support roadmap.	
Check prerequisites for an upcoming course	100%	None	Checked prerequisite easily.	
Check a course equivalency	0%	Page under construction.	Does not support equivalencies.	

Usability Metrics: Efficiency

Task	Time	Effort	Content/Design
Change an editable degree requirement	13s	Low	3 clicks
Search for a course at SFSU	12s	Low	1 click
Use the roadmap to plan courses by semester	N/A	N/A	1
Check prerequisites for an upcoming course	12s	Low	1
Check a course equivalency	N/A	N/A	1

Questionnaire: Scale 1 (Strongly Disagree) to 5 (Strongly Agree)

1. It was simple to pick a course for a degree requirement.
2. When choosing a course, the dialog was easy to navigate.
3. When choosing a course, there was plenty of information about each one.
4. It was easy to understand which requirement I was editing.
5. The course search provided me with plenty of information about any course.
6. The search feature was responsive and felt natural to use.
7. The search feature gave me enough freedom to search courses the way I wanted.
8. The roadmap was simple and easy to use.
9. The roadmap made it easy to organize my remaining semesters.
10. The roadmap was flexible enough to handle every scenario I needed it to.
11. It was easy to see what prerequisites I needed for my remaining courses.
12. The information on the prerequisite dialog was easy to understand.
13. It was clear when prerequisites were needed for any of my upcoming courses.
14. The course equivalency page was easy to use.
15. The course equivalency page had sufficient options for courses and schools.
16. The course equivalency page displayed information clearly.

Google Form Link: https://forms.gle/QrA18fcUXc7JXvVG9

QA Test Plan

In order to test the product, we will pick the following five nonfunctional requirements.

1. Every student shall get the information about the subject and their prerequisite courses and also the student shall see if there are any second level prerequisite courses.
   Test Objectives: - To test the feature we have to pass the right credentials of the user to check the all the courses and prerequisite courses are coming correctly
   HW and SW setup: - Window 11, Mac Monterey, Chrome, Firefox
   Feature to be tested: - All the courses, prerequisite courses, second level prerequisite courses.
   QA Test Plan: -

#	Description	Input	Output	Result
1	Check for all courses	Access token of login	CSC848…etc.,	PASS
2	Check for the prerequisite courses	CSC848	CSC413	FAIL
3	Check for the second level prerequisite courses	CSC848	CSC317	FAIL

2. Students can see all the basic knowledge about the course while going with the particular course.
   Test Objectives: - in order to test the objective is to check while a student enters a course it should come with right course information.
   HW and SW setup: - Window 11, Mac Monterey, Chrome, Firefox
   Feature to be tested: - Course details
   QA Test Plan: -

#	Description	Input	Output	Result
1	Undergraduate level course details	CSC 413	Units Hold, Description, Professor List	PASS
2	Graduate Level Course details	CSC 317	Units Hold, Description, Professor List	FAIL
3	Prerequisite Course	CSC 848	Units Hold, Description, Professor List	PASS

3. At a time one subject can be a prerequisite for one or more courses.
Test Objectives: - The objective of the test case is that one course can be prerequisite for one or more courses
HW and SW setup: - Window 11, Mac Monterey, Chrome, Firefox
Feature to be tested: -
QA Test Plan: -

#	Description	Input	output	Result
1	Check for one course	CSC 413	CSC 848. CSC 868	FAIL
2	Check for one course	CSC 415	CSC 780, CSC 845	FAIL

4. All the students who are accepted are eligible to access the tool and can view their course curriculum.
Test Objectives: - The objective to test this to verify the outsider or those students have not accepted can't access the portal.
HW and SW setup: - Window 11, Mac Monterey, Chrome, Firefox
Feature to be tested: -
QA Test Plan: -

#	Description	Input	output	Result
1	With Accepted student	Username, password	Logged in with session	FAIL
2	Without Accepted admission student	Username, Password	Not able to Access the portal	FAIL
3	Outsider with false	Username, Password	Not able to Access the portal	FAIL

5. Students can use the degree planner feature that shall give them ideas about the time expected to complete the degree.
Test Objectives: - The objective of the test case is that it is showing a perfect degree roadmap
HW and SW setup: - Window 11, Mac Monterey, Chrome, Firefox
Feature to be tested: - Degree Planner
QA Test Plan: -

#	Description	Input	Output	Result
1	To test undergraduate degree planner	All the selected courses for degree	4 years	FAIL

2	To test graduate degree planner	All the selected courses for degree	2 years	FAIL

Code Review

Part A

In our product we used web development with javascript as a primary language and used nodejs as a runtime.For frontend we used react js library which gave flexibility using html as a JSX.About the code structure is built on the atom structure where atom is component that can be used by other component which are molecules can be used as a template to make html page which is running on single div.In backend we used bob martin architecture where our request will be served with less time and maintainable code that can used further to break down microservice divide the module across the product.

Part B

Code Files that underwent review:

https://github.com/sfsu-joseo/csc648-848-06-sw-engineering-fall-2022-Team06/blob/master/Milestones/M4/BetaPrototype/client/src/components/CourseSelectionDialog.js

https://github.com/sfsu-joseo/csc648-848-06-sw-engineering-fall-2022-Team06/blob/master/Milestones/M4/BetaPrototype/client/src/components/PrerequisiteDialog.js

1. Team 7 Review
   === CourseSelectionDialog ===
   Needs comments!
   Generally you should not use inline styling excessively. Instead, use a .css file for styling because it lets you reuse styles.

   Line 21 could use a JSDoc comment describing the purpose of this function.
   e.g. /** ... */

   The ternary operators in this file should be replaced with a function or something because it is hard to read.
   For example:

```
function line78() {
        if (showDetails) {
                return (
                        <Button onClick={} ...>
                                Back
                        </Button>
                );
        } else return undefined;
}
```

And just call this function on line 78 instead of doing all your conditional checks in the return() block.
Same goes for line 89's very long ternary statement.

Line 94-102: What's going on here?
Line 138: This line is way too long. You should break it up into multiple lines.

=== PrerequisiteDialog ===
Needs comments.
Generally you should not use inline styling excessively. Instead, use a .css file for styling because it lets you reuse styles.

Line 15: CourseListItem could use some comments.

Need header files and in-line comments!

2) Team Member Vivek

The code in the file PrerequisiteDialog.js and CourseSelectionDialog.js is written with the ES6 modules and maintained with the latest react version which is 17.1. In the code arrow function is used to return JSX multiple times but with the dynamic values. JSX are well formed divided. The component is made on the functional base component which is the good thing to have.

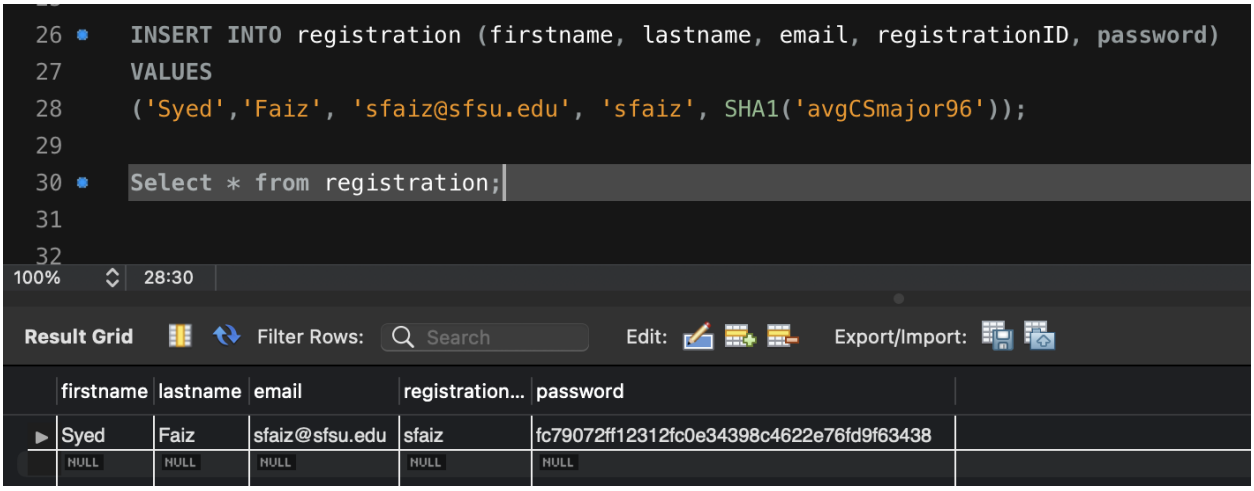
3) Code Review

The reason why I chose these 2 files is because the majority of our use cases and priority will deal with this code, and I wanted to get feedback on them. At the moment, the code has the data hardcoded, so it doesn't really help. However, the overall usability and functions will be important and I wanted to review these codes.

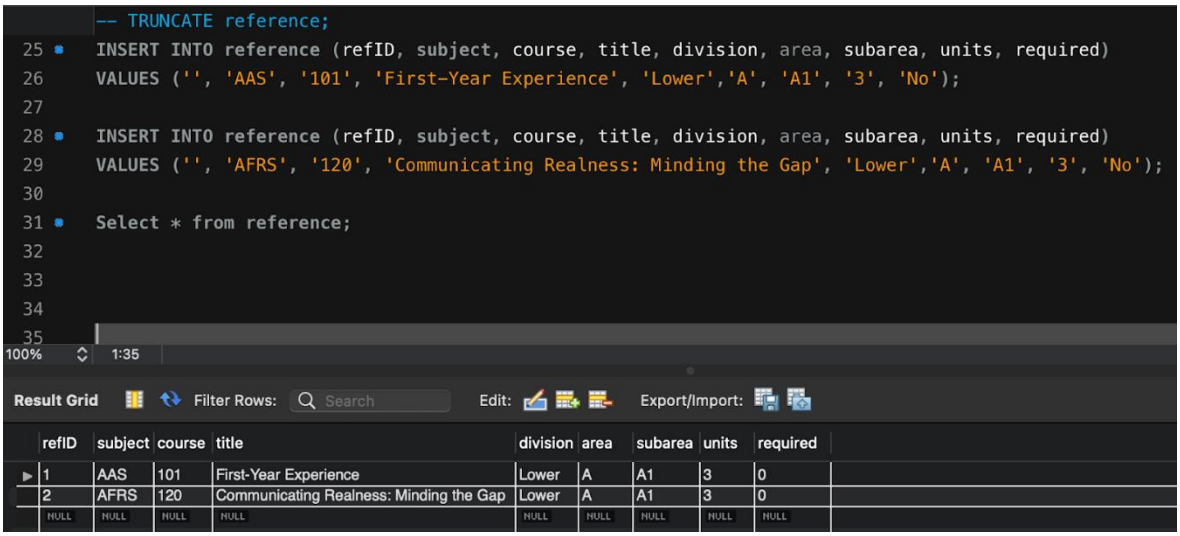
Security Practices

The main asset we are protecting for our application is the user login credential more specifically the password. To use the roadmap feature, users must enter their login which requires their registration ID and password. To protect the user password in the database, we will use the SHA1 hash function for encryption.

To show password encryption in the database, I added an insert to the registration table to register a user and use SHA1 for password. As a result, when I run the select * from registration, it returns the user account I created and the hashed encrypted password.



Similarly, for the search feature I used insert statements to add course data into the reference table. There will be over 800 courses to allow for ample search opportunities and filtering. Search could be filtered by subject, course number, title, units, division, area, subarea, and whether they are required or optional.



Non-functional Requirements

Non-Functional Specs	Status (Done, On-Track, or Issue)	Comments
All types of users will have the credentials that are provided by their university in order to access the req checker.	On-Track	**We are currently undergoing this function in our current build of the website.**
All the professors have to be authenticated by the university to use this tool as a university internal tool with the role of professor.	Issue	This will not be one of the main focus since it is priority 2.
All the admins have to be authorized by the university admin office in order to access this tool.	Issue	The function will not be implemented at this time for being a second priority.
All students(users) after the accepted admission are eligible to access the tool and can view their course curriculum.	On-Track	This is currently being worked on and will be completed int the future.
Every student shall get the information about the subject and their prerequisite courses and also the student shall see if there are any second level prerequisite courses.	Done	Each course shows the prerequisites need to be taken.
Every student can choose the subject and can send the request to the professor to let them in their courses.	Issue	Professor page will not be implemented due to being priority 2.
Every student can send the 2 types of requests to the professor which shall let them in the course, or they can request to eliminate prerequisite courses to eliminate them as per the student experience.	Issue	Professor page is not implemented for messages.
Students can share additional information about the work he has done with approval requests to the processor.	Issue	Professor page is second priority.
Students can also request to eliminate prerequisite courses if they have enough work done related to the course.	On-Track	This is going to be part of course equivalencies.
Students have to give mandatory notes with the reason why they want to take the particular subject.	Issue	Requires a professor page to receive message.
Students shall have ideas about all pending requests that they have sent to the professor.	Issue	Professor cannot receive request.
Students shall see the result of acceptance and rejection from the professor and can distinguish between the professor if a student has applied the same course from the two professors.	Issue	Request cannot be made due to professor page is not implemented.
Professor shall have the idea about all the requests through the total request table.	Issue	Professor table is priority 2 and will not be implemented.
Professors can directly accept or reject student requests with both types of requests.	Issue	Professor page cannot not view request.
Professor shall see the additional details along with student requests.	Issue	Professor page is second priority.
Professor can see all the numbers like waiting students, enrolled students and more.	Issue	Professor page cannot view student information.
Professor has all the rights to accept or reject the application of the student according to the details provided by the student	Issue	Professor online page still not implemented
Students can use the degree planner feature that shall give them ideas about the time expected to complete the degree.	On-track	This function is currently being implemented and have a prototype.
After matching all the credits and all the subjects' planners shall show the roadmap to take the courses and can save those to use letter on.	On-Track	We have all credits and subject's roadmap is under construction.
Students can see the list of their approved rejected courses by the professor.	Issue	Professor page is not implemented.
Users will only choose the website if they want to make the request for the approved courses and can see the degree planner.	On-Track	At the moment this is possible but there are still some tweaks that need to be made.

Non-Functional Specs	Status (Done, On-Track, or Issue)	Comments
All types of users will have the credentials that are provided by their university in order to access the req checker.	On-Track	**We are currently undergoing this function in our current build of the website.**
All the professors have to be authenticated by the university to use this tool as a university internal tool with the role of professor.	Issue	This will not be one of the main focus since it is priority 2.
All the admins have to be authorized by the university admin office in order to access this tool.	Issue	The function will not be implemented at this time for being a second priority.
All students(users) after the accepted admission are eligible to access the tool and can view their course curriculum.	On-Track	This is currently being worked on and will be completed int the future.
Every student shall get the information about the subject and their prerequisite courses and also the student shall see if there are any second level prerequisite courses.	Done	Each course shows the prerequisites need to be taken.
Every User will have to agree on the term and services with following conditions that will be shown in the format.	Done	This has been implemented in the login/register section.
To report any issue on the website, the user will fill the contact us page with their little summary about their concern.	On-Track	The contact page is currently being built.
Outside users can have to face legal action against the respective university.	On-Track	We are adding a legal disclaimer to the website at the bottom link.
In order to use the website student users must have their past academic info in their portal otherwise It can fail to give you the expected data.	On-Track	Most features are implemented but data is still being worked on.
Students shall be able to see a roadmap of all existing and future courses that must be taken.	On-Track	The roadmap is currently being worked on by devs.
Student shall pick their courses directly instead of manually inputting a course number	On-Track	This is currently implemented but still needs to be polished.
Students shall be able to switch between Fall, Winter and Summer courses.	On-Track	Semesters will be implemented for the course page.
Students shall be able to see the list of the subject that is approved by the professor.	Issue	A list of classes has been created but the professor page has not been created.
Student shall be authenticated after he gets admission in the university.	On-Track	Authentication is being worked on.
Students shall have ideas about all the subjects and grading methods.	On-Track	All courses will be given a description of grading options.
Students can see all the basic knowledge about the course while going with the particular course.	Done	Courses display all information inf the course section of the website.
Professor can see the list of the students who took the course and also track the record of getting dropped students.	Issue	The professor page is a second priority so it cannot be implemented.
At a time one subject can be a prerequisite for one or more courses.	Done	Prerequisite courses have been created for each course in the course section.
Students can ask more details about the ideal course and course structure to the professor before beginning with classes.	Issue	Messages cannot be sent to professor due to the professor page being second priority.
In the list students can have more idea about the courses once they decide to choose in different frames in the tool.	Done	This has been implemented in the course section of the website.
Professor shall also provide the details about the TA and other general info about their work which attracts students to take the course.	Issue	Professor page is not implemented.

Team Contributions

To be filled by team lead only:

Team Member Name	Role	Rating
Syed Faiz	Backend Lead	9/10
Victoria Wilson-Anumudu	GitHub Master	9/10
Eric Falk	Frontend Lead	9/10
Vivek Santoki	Frontend	8/10
Erik Rodriguez	Frontend	9/10