A picture containing text

Description automatically generated

ISS396 – Junior Project SE

Final Report

Student Specialty Advisor

Elaborated by:

Aymen Hammami, Youssef Aydi

Aziz Maazouz, Mahdi Klouz

Supervised by:

Dr. Asma Amdouni - MedTech

Tunis, 2021-2022

ABSTRACT

There is no doubt that every student around the world, at some point, is going to face a big question which could define his or her future: “What specialty do I want to pursue? How can I pick the right education for me, and how can it help me build a successful career?”

However, the answer to this is difficult to find when you are young, have no work experience, and lack crucial information.

This is the case with most of the South Mediterranean University’s students who wish to graduate as engineers in the future. The current package of information, figures, and flyers a first-year student (Freshman) or a second-year student (Sophomore) could get from the administration is deemed to be outdated.

The information offered about the courses and specialties is not only outdated but also extremely limited and lacks personalization. And in this current state of the world, where the internet has over 4.5 billion users, it is more reliable to reach the puzzled students using this network.

In this report, we present a free web application “Student Specialty Advisor,” which we believe will be a great and modernized solution when it comes to guiding current and future SMU (South Mediterranean University) students to their suitable specialty.

Student Declaration

*By submitting this report, the students promise on penalty of failure that*

*• they have cited any source from which they used data, ideas, or words, either quoted or  
paraphrased (e.g., parts of the report that is copied/pasted from the Internet, design or  
construction performed by another person, etc.).*

*• they have not received unpermitted aid for the project design, construction, report or  
presentation.*

**Table of Contents**

[1- Project Overview 4](#_Toc103126760)

[**1.1-** **Introduction** 4](#_Toc103126761)

[**1.2-** **Project Description** 4](#_Toc103126762)

[**1.3-** **Problem Definition** 4](#_Toc103126763)

[**1.4-** **Vulnerabilities and Existing Solutions** 4](#_Toc103126764)

[**1.5-** **Proposed Solution** 5](#_Toc103126765)

[**1.6-** **Conclusion** 5](#_Toc103126766)

[2- Product Specifications 6](#_Toc103126767)

[**2.1-** **Introduction** 6](#_Toc103126768)

[**2.2-** **Project Management** 6](#_Toc103126769)

[**2.3-** **Product Backlog** 7](#_Toc103126770)

[**2.4-** **Sprint Backlog** 8](#_Toc103126771)

[**2.5-** **Requirements Specification** 11](#_Toc103126772)

[**2.6-** **Project Considerations** 12](#_Toc103126773)

[**2.7-** **Conclusion** 13](#_Toc103126774)

[3- Product Design 14](#_Toc103126775)

[**3.1-** **Introduction** 14](#_Toc103126776)

[**3.2-** **Sequence Diagrams** 14](#_Toc103126777)

[**3.3-** **UI Design** 17](#_Toc103126778)

[**3.4-** **Conclusion** 25](#_Toc103126779)

[4- Development and Prototype Solution 26](#_Toc103126780)

[**4.1- Introduction** 26](#_Toc103126781)

[**4.2- Development** **Details** 26](#_Toc103126782)

[**4.3- Conclusion** 26](#_Toc103126783)

[5- Tests, Results, and discussions 27](#_Toc103126784)

[**5.1** **Introduction** 27](#_Toc103126785)

[**5.2** **Test Cases** 27](#_Toc103126786)

[**5.3** **Discussion** 28](#_Toc103126787)

[**5.4** **Conclusion** 28](#_Toc103126788)

[6- General Conclusion & Future work 29](#_Toc103126789)

[**References** 30](#_Toc103126790)

1. Project Overview

# **Introduction**

In this chapter, we will discuss the value of our project, the problem we are trying to solve and how our idea is supposed to overcome it. We will start with a brief description of the project, followed by the problem-solution strategy.

# **Project Description**

The project is about a web-based application, called “Student Specialty Advisor” and is a free service offered to the current and future South Mediterranean University’s students. Its goal is to help students pick the right specialty / program for them, so they can excel not only in their studies, but also in their future career.

# **Problem Definition**

Students around the world become puzzled once they must choose which education specialty / specific degree to follow, and that is due to the lack of their experience, information, and fear that this might define their future career, if not their future overall. This is also the case when it comes to our university’s students, and especially MedTech’s students who wish to graduate as engineers.

The current information a student can get from SMU’s website or administration is outdated and consists mostly of the name of the courses or the curriculum. It lacks personalization and the valuable feedback of students who already picked a specialty.

# **Vulnerabilities and Existing Solutions**

We live in a hyperconnected world where we can almost find any kind of information we seek on the internet. Therefore, there is no doubt that multiple websites exist out there which offer general descriptions about the study programs, future careers, and some useful testimonials from both students and professionals.

However, we believe that because we will target specifically SMU students, our project would be considered the better solution for them, as it is customized to their exact needs.

# **Proposed Solution**

Our web-based application will offer multiple ways for a lost student to decide on what specialty to pursue. These features consist of:

* A high-quality program compatibility quiz, inspired by the work of other American universities, and related statistics to help.
* An up-to-date package of information about all the courses, subjects, and probable future careers, followed with brief videos from professionals and professors.
* An assistance chat-bot that can understand written questions and answer them well.
* A system that helps students reserve meetings with advisors (senior/final students and professors).
* A small forum where students can exchange more information and express their opinions about the different specialties.

# **Conclusion**

In this part, we went through our motivations and goals. We also defined the problem we are trying to solve by the features we proposed.

In the next chapter, we will discuss in depth the product specifications, requirements, and methodology.

1. Product Specifications

# **Introduction**

In this chapter, we will translate the needs of our web application’s end user into technical requirements and specifications. We will also describe the development steps we took while adopting Scrum and discuss the constraints & standards applicable to our project.

# **Project Management**

During the development of the project, we applied agile software development practices to gradually evolve the project and improve our work.

* + 1. **Methodology**

We adopted the Scrum framework to manage our project in an agile and efficient way. We were supervised by the product owner Ms. Asma Amdouni, and assigned the roles (Scrum master, development team) between the members of our group.

Throughout the scrum journey, we learned how to cooperate and efficiently work within a team, how to divide tasks properly and how to produce new constructive ideas.

* + 1. **Timeline**

We have divided the project into features, where each feature has a set of user stories assigned to it. We did this to have a logical representation of the project and to define its priorities.

We split the development period into 6 2-weeks sprints. In this subsection, we present how we conducted the Scrum ceremonies.

|  |  |  |
| --- | --- | --- |
| **Sprints** | **Features** | **Status** |
| Sprint 1 | Account System  Profile Management | Done |
| Sprint 2 | Program Compatibility Quiz | Done |
| Sprint 3 | Specialty videos page  Specialty in depth information page | Done |
| Sprint 4 | Meetings with Advisors System | Done |
| Sprint 5 | Admin Dashboard  Assistance Chat Bot | Done |
| Sprint 6 | Community Forum | Done |

# **Product Backlog**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | As a | I want to be able to | So that | Priority |
| 1 | Visitor | Sign up / Create an account | I can use all the website features | MUST |
| 2 | User | Log into my account | I can use all the website features | MUST |
| 3 | User | Change my profile information | I can keep my profile up to date | SHOULD |
| 4 | User | Change my password or email | I can control access to my account | SHOULD |
| 5 | User | Take a program compatibility quiz | I can know which specialty suits me | MUST |
| 6 | Admin | Generate random answers to the quiz | I can improve the quiz algorithm and collect information | COULD |
| 7 | User | Display videos about the different specialties | I can get an idea about them | SHOULD |
| 8 | User | Display in-depth information about specialties such as description, curriculum, future opportunities... | I can know what to expect if I pick that specialty | MUST |
| 9 | User | Request a meeting with an advisor (student or professor) | I can ask my questions about the specialty to the advisor | MUST |
| 10 | User | Display the advisor contact information | I can contact them myself through another way | SHOULD |
| 11 | User | Display the advisor availability | I can get an idea about the time of the meetings | MUST |
| 12 | User | Apply to become an advisor (student) | I can help other students with my experience | COULD |
| 13 | Admin | Display statistics about the program compatibility quiz from the dashboard | I can improve the quiz algorithm and collect information | MUST |
| 14 | Admin | Manage the list of advisors and their scheduled meetings from the dashboard | I can easily update the list of advisors and the times of the meetings | MUST |
| 15 | Admin | Manage the list of videos in the specialty videos section from the dashboard | I can easily update the list of the published videos | SHOULD |
| 16 | Admin | Manage the list of the signed-up users from the dashboard | I can have better control over who is using the website | SHOULD |
| 17 | User | Display the community forum | I can participate in the threads and discussions there | MUST |
| 18 | User | Post a comment in a forum's discussion | I can share my opinion about a specialty or a subject | MUST |
| 19 | User | Delete a comment I posted in a forum's discussion | I can control what comments to leave | COULD |
| 20 | User | Display the comments inside a forum's discussion | I can see what other users think about a specialty or a subject | MUST |
| 21 | Admin | Moderate comments | I can remove offensive or misleading comments | SHOULD |
| 22 | User | Suggest improvements and send feedback | I can contribute to the website's usability and content | SHOULD |
| 23 | User | Report a bug that happened to me | I can contribute to it getting fixed | SHOULD |
| 24 | Visitor Or User | Chat with an assistance bot | I can get help with the website or a specialty | MUST |

# **Sprint Backlog**

We used a web-based list-making application called “Trello” to organize and manage the tasks within the sprints. We divided the board into four sections: TASKS, IN PROGRESS, TO VERIFY, and DONE and we moved each task through them according to its status.

Sprint 1: Account System & Profile Management

|  |  |  |
| --- | --- | --- |
| **Tasks** | **Member** | **Complexity** |
| Creation of API endpoints | Youssef | L |
| Creation of MongoDB models | Youssef | L |
| Implementation of password hashing | Aziz | M |
| Implementation of authentication | Aymen | M |
| Implementation of registration | Aymen | L |
| Implementation of controller responsible for updating account information | Aziz | M |
| Implementation of JSON Web Token | Aymen | M |
| Creation of sign-in and sign-up forms | Aymen | M |
| Creation of home page | Mahdi | L |
| Creation of navigation bar | Mahdi | M |
| Creation of profile page | Youssef | L |

Sprint 2: Program Compatibility Quiz

|  |  |  |
| --- | --- | --- |
| **Tasks** | **Member** | **Complexity** |
| Creation of API endpoints | Youssef | L |
| Creation of MongoDB models | Youssef | L |
| Implementation of controller to process quiz answers | Aymen | M |
| Implementation of controller to store results as statistics | Youssef | L |
| Implementation of quiz question component | Aziz | M |
| Implementation of quiz questions generator | Aziz & Mahdi | M |
| Implementation of sending the answers to the backend server | Mahdi | L |
| Implementation of an admin only button that allows to simulate random answers | Aymen | M |

Sprint 3: Videos & Information sections

|  |  |  |
| --- | --- | --- |
| **Tasks** | **Member** | **Complexity** |
| Creation of API endpoints | Mahdi | L |
| Creation of MongoDB models | Youssef | L |
| Implementation of controller to send all the video links | Youssef | L |
| Implementation of videos page | Aymen | M |
| Implementation of component that contains a video | Aziz | M |
| Implementation of program information page | Aymen | L |
| Saving in a document all the information to display in the program information page | Mahdi | L |
| Retrieval of the video links to show to the user | Aziz | L |

Sprint 4: Meetings with Advisors System

|  |  |  |
| --- | --- | --- |
| **Tasks** | **Member** | **Complexity** |
| Creation of API endpoints | Youssef | L |
| Creation of MongoDB models | Youssef | M |
| Implementation of controllers to manage schedule | Aymen | M |
| Implementation of controllers to manage reservations | Aymen | H |
| Implementation of controllers to manage advisors | Aziz | M |
| Implementation of controller to send emails to advisors | Aymen & Youssef | M |
| Creation of an “About” page to introduce the concept | Mahdi | L |
| Creation of an “Advisors” page to introduce the advisors | Aziz | L |
| Creation of a “Schedule” page to request / reserve meetings | Aymen & Aziz | L |
| Implementation of the reservation system in the “Schedule” page | Aymen | M |

Sprint 5: Admin Dashboard & Assistance Chat Bot

|  |  |  |
| --- | --- | --- |
| **Tasks** | **Member** | **Complexity** |
| Creation of API endpoints | Youssef | L |
| Implementation of controllers to manage advisors | Youssef | M |
| Implementation of controllers to manage meetings | Aymen | L |
| Implementation of controllers to manage videos | Aziz | L |
| Implementation of controller to retrieve statistics about the quiz | Mahdi | L |
| Creation of the dashboard page layout | Aziz | L |
| Creation of the management items displayed on the dashboard page | Aymen | M |
| Creation of graphs and charts that display statistics in the dashboard page | Youssef | M |
| Implementation of assistance chat bot | Aymen | M |

Sprint 6: Community Forum

|  |  |  |
| --- | --- | --- |
| **Tasks** | **Member** | **Complexity** |
| Creation of API endpoints | Aziz | L |
| Creation of MongoDB models | Youssef | M |
| Implementation of controllers to update existing threads and comments | Aymen | H |
| Implementation of controllers to store or retrieve the discussions/threads | Youssef | M |
| Implementation of controller that allows an admin to moderate the discussions/threads | Mahdi | L |
| Integration and implementation of email verification once an account is created | Youssef | M |
| Creation of the community forum page and display the discussions/threads that it includes | Aymen | M |
| Creation of the page that will display a thread / discussion and its user posts / messages | Aymen | H |
| Implementation of controllers to manage threads in admin dashboard | Youssef | M |

# **Requirements Specification**

* + 1. **Functional Requirements**

The following USE CASE Diagram describes the functional requirements of the project:Diagram

Description automatically generated

* + 1. **Non-functional Requirements**
* Usability: the project should be easy to use and easy to access even for non-technical users. The user should be able to quickly achieve the target and navigate freely in the features of the website.
* Security: the user information should be kept private and safe, and the password given on account creation should be hashed.
* Performance: the web-based application should be able to handle errors, load in a decent time and handle HTTP methods efficiently.
* Responsiveness: the web-based application should be responsive on both mobile and desktop devices and ensure a good overall user experience.

# **Project Considerations**

* + 1. **Project Constraints**
* Scope constraint: There is a possibility for scope creep to occur due to our adoption to Scrum as a development framework.
* Time constraint: The deadline of the project is the second week of May.
* Cost constraint: The project has a low budget.
* Experience constraint: We do not have prior experience of building a mobile UI, and no prior experience of building a scalable account-based web application.
  + 1. **Project Limitations**
* Adopting Scrum as a framework for development.
* Using JavaScript as the coding language and Node.js as the runtime environment.
* Building the frontend of the project with React.js for desktop and mobile devices.
* Building the backend of the project using Express.js and a NoSQL database (MongoDB in the case of our project).
  + 1. **Project Delimitations**
* Due to the covid-19 pandemic, some team members had to stop contributing to the project for health concerns and that slowed down our progress. We also had to work remotely for a few weeks because of a national quarantine.
* Since the internet is a vast network visited by millions every minute, we have to build and deploy the project on the web for it to be accessible and to reach the SMU students.
  + 1. **Project Standards**

We tried to apply the following software development standards while working on the project: IEEE/ISO/IEC P15288 - ISO/IEC/EEE Draft Standard - Systems and Software engineering -- System Life Cycle Processes:

“This International Standard establishes a common framework of process descriptions for describing the life cycle of systems created by humans. It defines a set of processes and associated terminology from an engineering viewpoint. These processes can be applied at any level in the hierarchy of a system’s structure. Selected sets of these processes can be applied throughout the life cycle for managing and performing the stages of a system's life cycle. This is accomplished through the involvement of all stakeholders, with the ultimate goal of achieving customer satisfaction.” (ISO.org, 2015)

* + 1. **Business, Social and Ethical Considerations**

The goal of “Student Specialty Advisor” is to reach the SMU community, and to be a free tool for the students that provides them with every information they need to decide which specialty suits them best. We believe that will impact the distribution of the students between the different specialties and will strengthen the social community inside of SMU.

# **Conclusion**

In this chapter, we discussed the technical requirements and specifications of the project, as well as the steps and tasks we took during the development phase.

In the next chapter, we will discuss all the aspects of the product design.

1. Product Design

# **Introduction**

In this chapter, we will go through the design decisions of the project, and we will showcase the UI of some sections that can be found in our web application.

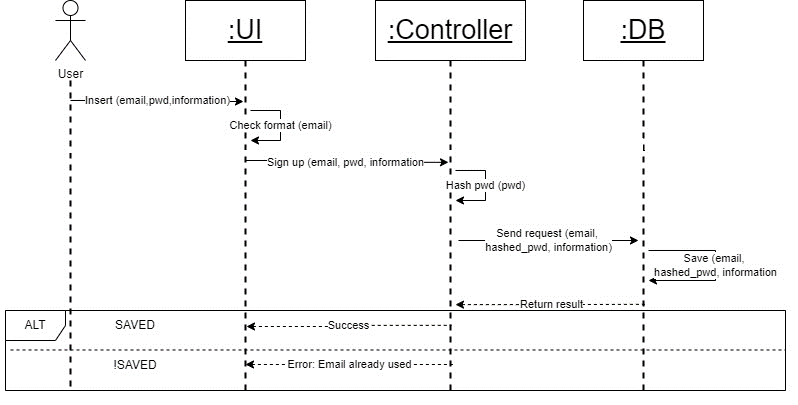
# **Sequence Diagrams**

Authentication

Diagram

Description automatically generated

Sign Up



Assistance Chat Bot

Diagram

Description automatically generated

Program Compatibility Quiz

Diagram

Description automatically generated

# **UI Design**

Below, we will showcase some of the sections of “Student Specialty Advisor” by taking screenshots from the project.

The left section of a figure represents the display of a mobile device.

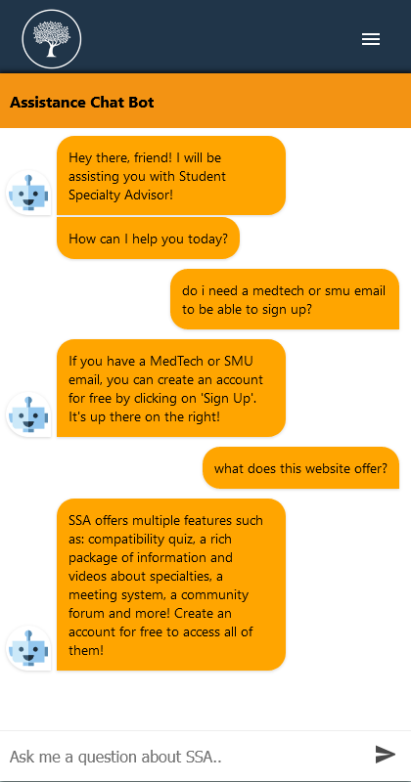
The right section of a figure represents the display of a desktop device.

Timeline

Description automatically generatedTimeline

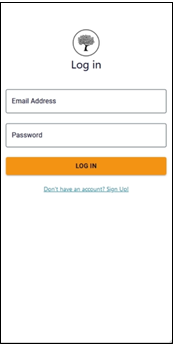
Description automatically generated

Figure 1: Home Section (not logged in)

Graphical user interface, application

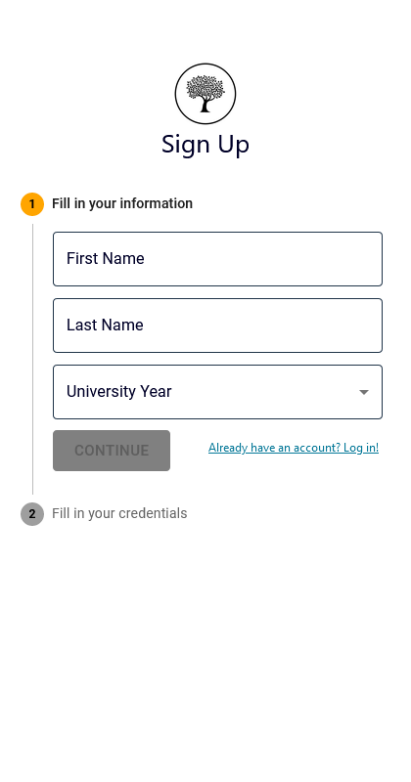
Description automatically generated

Figure 2: Assistance Chat Bot

A picture containing text, screenshot, computer

Description automatically generated

Figure 3: Login Section

A picture containing graphical user interface

Description automatically generated

Figure 4: Sign Up Section

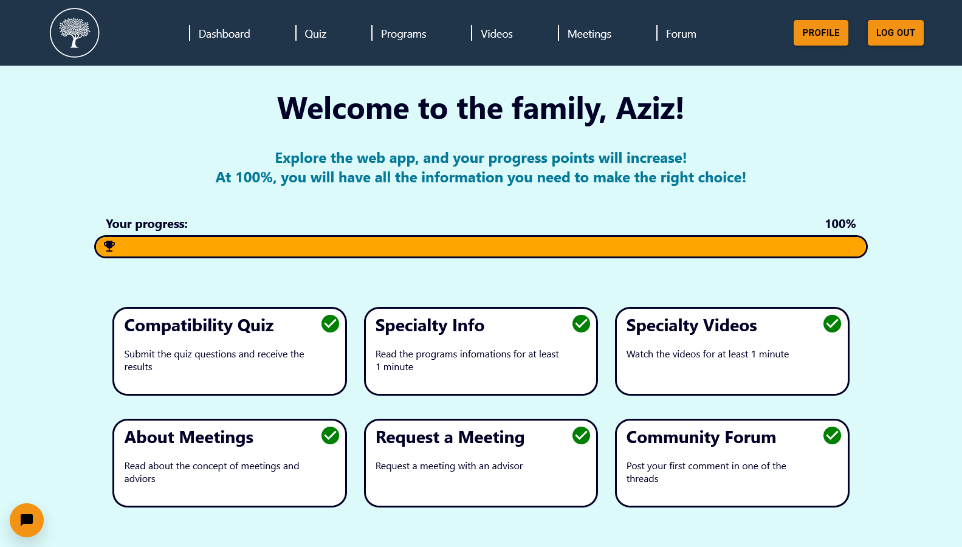
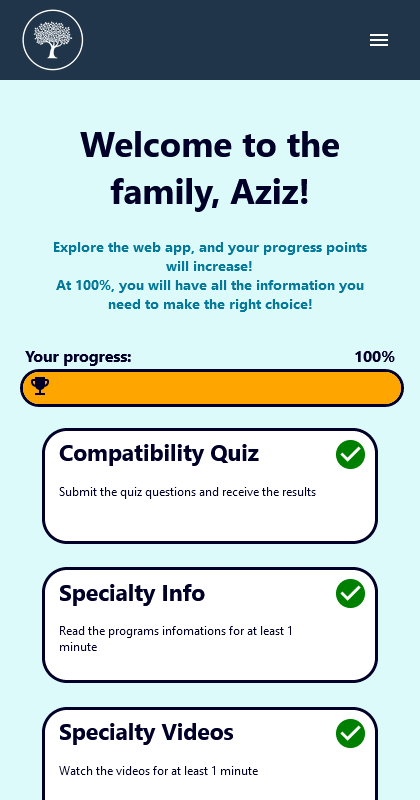


Figure 5: Home Section (logged in)

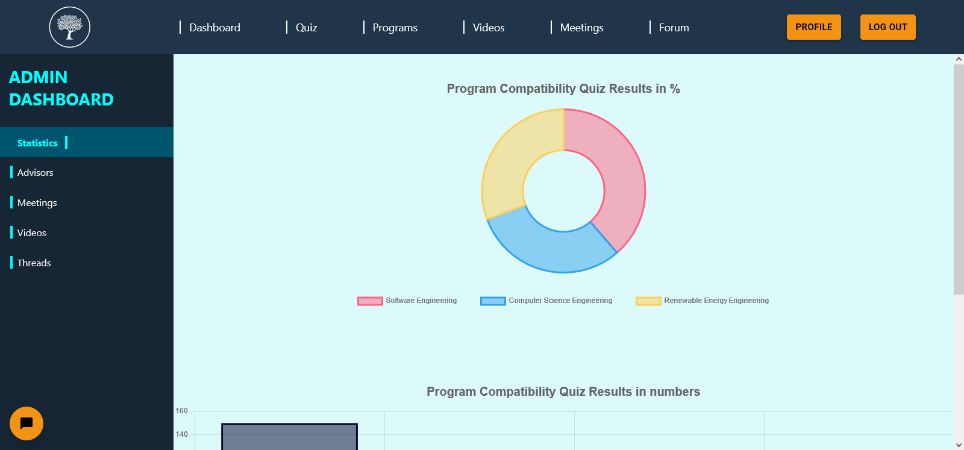
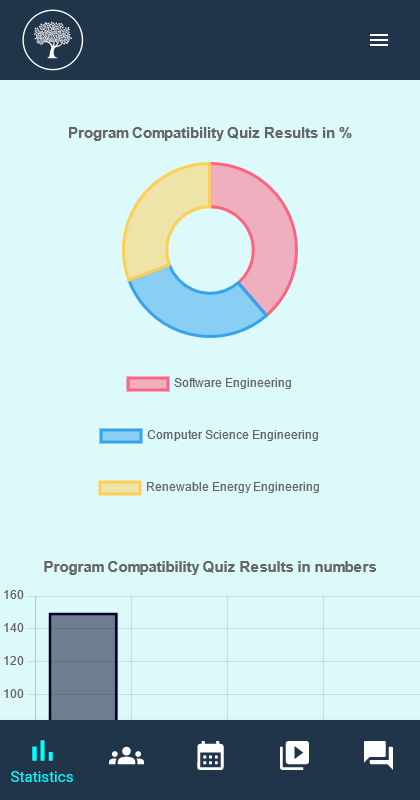


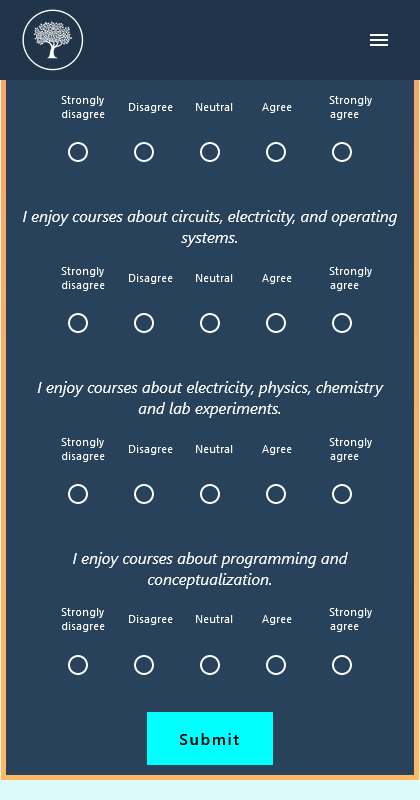
Figure 6: Admin Dashboard Section

Text, letter

Description automatically generatedGraphical user interface, text, application, email, website

Description automatically generated

Figure 7: Program Compatibility Quiz Section

Graphical user interface, application

Description automatically generated

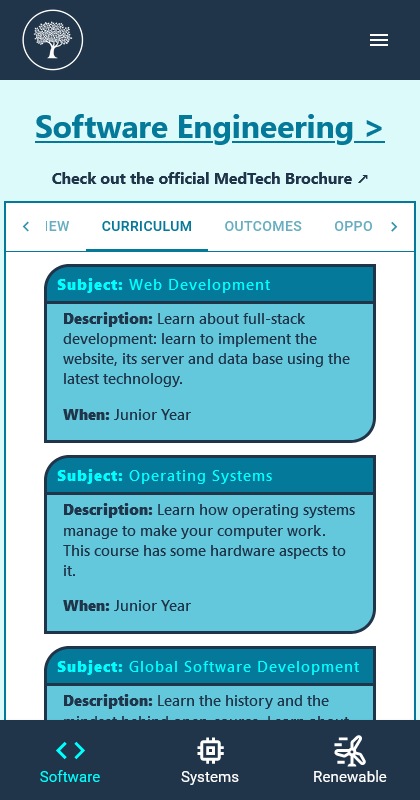
Figure 8: Program Compatibility Quiz once started

Graphical user interface, text, application

Description automatically generatedA screenshot of a computer

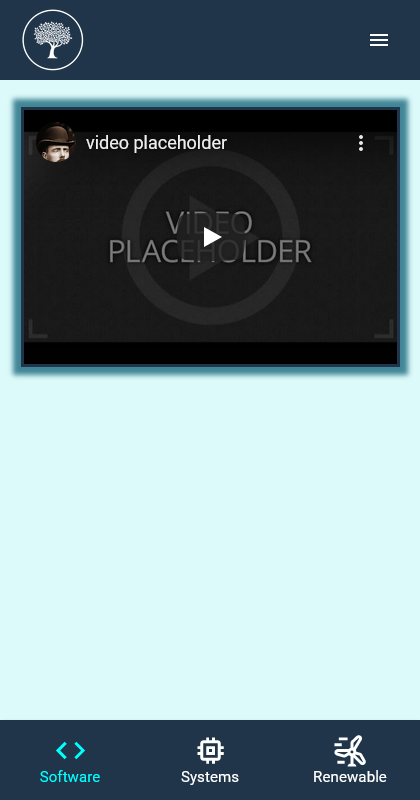
Description automatically generated

Figure 9: Program Compatibility Quiz results

Graphical user interface, website

Description automatically generated

Figure 10: Programs Info Section

Graphical user interface, website

Description automatically generated

Figure 11: Videos Section

Graphical user interface, text, application

Description automatically generatedGraphical user interface

Description automatically generated with medium confidence

Figure 12: Meetings Schedule Section

Graphical user interface, text, application

Description automatically generatedGraphical user interface, text, application, email

Description automatically generated

Figure 13: Community Forum Section

Graphical user interface, text, application

Description automatically generatedGraphical user interface, text, application, email

Description automatically generated

Figure 14: Inside a forum thread (top)

Graphical user interface, text, application

Description automatically generatedGraphical user interface, text, application

Description automatically generated

Figure 15: Inside a forum thread (bottom)

# **Conclusion**

In this chapter, we discussed the design aspect of the project and went through some footage of the project’s current UI.

In the next chapter, we will discuss in depth the development details and support of the project.

1. Development and Prototype Solution

# **4.1- Introduction**

In this chapter, we will discuss the development details, the development support, and different frameworks and libraries we used during the implementation of the project.

# **4.2- Development** **Details**

The project is composed of three systems: the frontend / the client, the backend / the server, and the database.

For the development of the client, we adopted some of the popular web technologies and coding languages as the basis of the project. We used Node.js as the runtime environment and built the client on React JS.

*“React is a free and open-source front-end JavaScript library for building user interfaces based on UI components. It is maintained by Meta and a community of individual developers and companies.”* (Wikipedia, 2022)

Since we wanted the project to be usable on both mobile and desktop devices and directly browsable from the internet, we decided to adapt the web application to mobile using different viewports and media queries, instead of building the project from scratch using an alternative such as React Native.

For the backend, we used ExpressJS alongside Node.js to profit from the robust features already implemented within this framework.

*“Express is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications.”* (ExpressJS, 2022)

To be able to store all the information necessary for the project to work, we decided to go with MongoDB, the world’s most popular and trusted NoSQL Database.

# **4.3- Conclusion**

In this chapter, we discussed the development details and support of the project in depth.

In the next chapter, we will discuss the test cases and the issues we faced during the development of the project. We will also give an assessment of the results and limitations.

1. Tests, Results, and discussions

# **Introduction**

In the next chapter, we will discuss the test cases and the issues we faced during the development of the project. We will also give an assessment of the results and limitations.

# **Test Cases**

Below are some of the test cases of the project that can be considered as crucial bugs:

|  |  |  |  |
| --- | --- | --- | --- |
| **Description** | **Expected Result** | **Actual Result** | **Is the bug fixed?** |
| Visit the sign up or login page while already logged in using the URL directly | Inform the user that he or she is already logged in, and that he or she needs to logout first if they want to sign up or log into another account | The web application does not realize that the current user is already logged in | Yes |
| Pressing enter after typing your account credentials in the login section | Pressing enter simulates a click on the Login button, and submits the credentials | Pressing enter redirects the user to the Sign-up section | Yes |
| In the dashboard, delete an advisor that already has meetings posted | Do not allow the advisor to be deleted unless all the meetings related to it are deleted first | Allows the advisor to be deleted, and the web app crashes | Yes |
| Change profile information | Changing profile information should update it in the database and the user’s local storage | Changes only the profile information in the database. The user must re-log into account for the changes to be reflected | Yes |
| Request a meeting from the meetings schedule | The meeting becomes reserved in the database and an email is sent to the advisor | The meeting becomes reserved in the database but sending the email works only half of the time | Yes |

# **Discussion**

Throughout all the sprints that we scheduled, we managed to deliver the design and the implementation of the features mentioned in chapter 1 and chapter 2.

So far, the web application is functional, and the features are bug free.

However, the “Videos” section of the project still does not have any videos listed there. The process of shooting a professional video, directing it, editing it and uploading it takes a long time, and due to the covid-19 preventative measures and the schedule changes in the month of Ramadhan, it became even more difficult for us to handle recording these videos.

This has also affected the number of the listed “Advisors” that a user can request, which is currently minimal mostly composed of students, since we did not find enough time to propose the position to professors and other experts.

It is also possible to improve on the Assistance Chat Bot feature and integrate AI into it.

The Community Forum is currently functioning as intended, but the users are represented with a letter avatar, instead of a profile picture. This could be added to the user’s profile as an option that enable more personalization.

# **Conclusion**

In this chapter, we discussed the results and the issues of the project.

In the next chapter, we will conclude this report and talk about our vision for the future of the project.

1. General Conclusion & Future work

Student Specialty Advisor is a web-based application aimed at SMU students, specifically MedTech students who have difficulties choosing the right engineering specialty for them.

We want to provide the students with all the information they need in one place and offer them a space where they can express their opinions and rate the specialties freely.

To achieve our goal, we developed a project of multiple features that the users can enjoy once they create their accounts. The current features include:

* A high-quality program compatibility quiz, inspired by the work of other American universities, and related statistics to help.
* An up-to-date package of information about all the courses, subjects, and probable future careers, followed with brief videos from professionals and professors.
* An assistance chat-bot that can understand written questions and answer them well.
* A system that helps students reserve meetings with advisors (senior/final students and professors).
* A small forum where students can exchange more information and express their opinions about the different specialties.

So far, we have managed to deliver all the features in a functional and bug free state.

We want the project to remain useful for the future generations of MedTech, and we want to gradually expand its scale, and that is why we have decided to make Student Specialty Advisor an open-source project that any interested student can contribute to.

We think that going open source will bring more traffic to the web application, will be a great addition to the Global Software Development course, and will help improve the quality and development of Student Specialty Advisor.

# **References**

ExpressJS. (2022). *ExpressJS*. Retrieved from ExpressJS: https://expressjs.com/

ISO.org. (2015, 05). *ISO/IEC/IEEE 15288:2015*. Retrieved from iso.org: https://www.iso.org/standard/63711.html

Wikipedia. (2022). *React\_(JavaScript\_library)*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/React\_(JavaScript\_library)