WEB-ACADEMY: AN ONLINE EDUCATIONAL PLATFORM

Capstone Design

Ismail Mounjid

Supervised by:
Pr. Omar Iraqi Houssaini

Spring 2019
Capstone Report

Student Statement:

I, Ismail Mounjid, assert that I have applied ethics to the design process and in the selection of the final proposed design. I also affirm that I have held the safety of the public to be paramount and have addressed this in the presented design wherever may be applicable.

_________________________________________  Ismail Mounjid

Approved by the Supervisor

Pr. Omar Iraqi Houssaini
ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my supervisor Pr. Omar Iraqi Houssaini for his guidance and insight throughout my capstone in order to help me finish this project.

I am also thankful to AUI for the education it provided me with as well as the whole experience for shaping me into the person I am today.

Lastly, I would like express my indebtedness to my parents, family and friends which always gave me the support and love needed for me to complete this thesis.
Table of Contents
1. INTRODUCTION .................................................................................................................. 1
2. PROJECT SCOPE ................................................................................................................ 2
3. STEEPLE ANALYSIS ......................................................................................................... 2
   3.1. Socio-cultural .............................................................................................................. 2
   3.2. Technological .............................................................................................................. 2
   3.3. Economical .................................................................................................................. 3
   3.4. Environmental ............................................................................................................ 3
   3.5. Political ....................................................................................................................... 3
   3.6. Legal ............................................................................................................................ 3
   3.7. Ethical .......................................................................................................................... 3
4. FEASIBILITY STUDY .......................................................................................................... 3
   4.1. Technical Feasibility .................................................................................................... 4
   4.2. Time Feasibility .......................................................................................................... 4
   4.3. Economic Feasibility .................................................................................................. 4
   4.4. Learning Outcome ...................................................................................................... 5
5. REQUIREMENTS SPECIFICATIONS ............................................................................... 5
   5.1. Functional Requirements ............................................................................................ 6
       5.1.1. Use-Case Diagram .............................................................................................. 6
       5.1.2. As a general user ............................................................................................... 7
       5.1.3. As a student ........................................................................................................ 7
       5.1.4. As a teacher ........................................................................................................ 7
   5.2. Non-Functional Requirements .................................................................................... 8
       5.2.1. Access Requirements ....................................................................................... 8
       5.2.2 Integration Requirements .................................................................................... 8
       5.2.3 Security Requirements ....................................................................................... 8
       5.2.4 Performance Requirements ................................................................................ 8
       5.2.5 Scalability Requirements .................................................................................... 8
       5.2.6 Extensibility Requirements .................................................................................. 9
Abstract:

The motivation behind this project is to provide Moroccan students with an educational platform. This is especially important in the context of this country, firstly because there are no such educational platforms that are particularly adapted to the Moroccan education and curriculum, and secondly, is that the case of Morocco makes the importance of such application more interesting especially knowing the disparity in the access of education between different social classes in the country as well as the significant contrast in the quality of education between the private and public sector of education in Morocco.

To help achieve such a goal, we have designed and developed a web application that provides an online educational platform for both students and teachers. As an educational platform, this web application shall be used by both educators and students in order to access and share educational content. The software product has both a student side and a teacher or instructor side. From the student side, the user is able to access classes which contain lessons and exercises to practice his or her learning. From the teacher side, the user has the possibility to create classes in which he/she is able to put lessons with their respective practical exercises. The web application is looking forward to be quite impactful, specifically in the local context of Morocco.
1. INTRODUCTION

With the appearance of internet and the massive amount of data that is shared and browsed everywhere we go, we should only be more aware of the importance of education nowadays and the impact it can have on our lives. Indeed, education as knowledge is not merely a bundle of concepts and information which one carries around, it is even more. By having the right knowledge of the world around us, through education, we can effortlessly change it into something better. Education helps us enhance and change our perspective of the world and having access to it should be of no issue to anyone.

That is the main concept behind my project Web-Academy, which is an online educational platform shaped around the idea of making education accessible everywhere and for free. Although similar platforms already exist, Web-Academy’s scope would be projected towards the local Moroccan audience by being it adapted to the Moroccan education and curriculum. This is also especially important in the case of Morocco, knowing the disparity in the quality of education between schools, specifically private and public ones. The aim would be to even out the quality of education through a platform for both students and instructors to be able to share and access educational content based upon the Moroccan curriculum.

The project would be first targeting the first level of educations, mainly high school and secondary, then move into higher levels of education with the purpose of making it more global.
2. PROJECT SCOPE

Web-Academy is a web application designed especially for computer devices. This capstone project consists of building an online educational platform which aims at providing a decent and free access to educational content to both students and teachers. The timeframe implementation of such project would be spring 2019 and would act as a fulfilment of the Capstone Design course.

3. STEEPLE ANALYSIS

The STEEPLE analysis is done in the aim to evaluate the application with regards to its influence on external factors which are socio-cultural, technological, economic, environmental, political, legal and ethical.

3.1. Socio-cultural

With the significant increase in the access to internet, and a limitless amount of information, having a similar access to education has never been so important. In fact, in this day and age where information is everywhere, education would be the most significant as it would be a way to provide critical information and way of thinking which would help discern between falsehood and truth. This is especially true knowing how easy it has been proven that masses of people can be easily brainwashed for fault of no ability to think and process information critically due to a lacking level of education and knowledge.

3.2. Technological

It is only thanks to new technologies and the easiness of accessing and learning about them that the implementation of this application has been possible. Indeed, technology provides us more than just some additional tools to be used to make our lives easier, but its impact can be even more significant and life-changing if combined to be utilized in transformative ways. This is the main aim behind this web application; to make technology a mean towards reaching nobler ends such as education.
3.3. Economical
This application’s development was only possible through the availability of open source technologies such as Vue.js and Node.js among many others. Nevertheless, it is expected that this application will have a huge empowering potential through free education enabling seekers to better themselves in life in general, including the economical aspect.

3.4. Environmental
Web-Academy will not have any impact on the environment since it does not make use of any environmental resource.

3.5. Political
No political implication will be the cause or consequence to the implementation of this project. This academic project would have as its main goal the showcase of the material learned throughout my undergrad education as a capstone.

3.6. Legal
The application will follow all legal laws and regulations since it will be using open source technologies as well as it will protect the confidentiality of its users without disclosing any of their private information.

3.7. Ethical
From the ethical perspective, Web-Academy will not result in the violation in any of the ethical regulations which will mainly have to do with its users and the use of their information for other ends and means for providing additional resources.

4. FEASIBILITY STUDY
The aim of this feasibility study is to define the achievability of our project from three main perspectives: technical, temporal and economical. From the technical aspect, there is the need to know the available technologies which will help build the project. Not only that, but actually
finding the available technologies which are most suitable and effective with regards to the project. The next aspect which is time is very important as well as it will determine the amount of time expected or needed for the project’s completion. As for the economical perspective, the concern would be to analyse the amount of funds which would go into the making of this project.

4.1. Technical Feasibility

In the technical aspect, the technology enablers which would be used in our project are: Node.js, Express.js, NPM, HTML5, CSS, Vue.js, Vuetify, MongoDB and Mongoose. Part of the decision that went into choosing these technologies was the previous experience and familiarity with some of them. Another part was that these were found to be the most efficient and suitable to the project. Yet, another reason was that these technologies are basically widely used in the field and have good feedback, which means better reliability. Most of these technologies were neither used nor learned in any of my classes here at AUI. However, the concepts, material and similar technologies learned in these classes provided me with the essential understanding needed to be able to learn how to use them.

4.2. Time Feasibility

From a temporal perspective, the feasibility of this project seems possible in 3 months, despite needing a lot of time. One important aspect would be the amount of learning that will have to be done on my own, which may take a significant length of time. In fact, although I’ve experienced with some of these technologies, I will still have to do more digging on my own in order to master all of them, but more importantly, to be able to use almost all their potential for this project.

4.3. Economic Feasibility
A last aspect is the economical one of this project. This aspect would be the least concerning and significant of all the other ones. Indeed, the feasibility of the project economically is very possible as almost all the resources would be free and available to use. Yet, the trivial cost which would be incurred in order to run it and build it such as the need for an internet connection, either through Ethernet cable or Wi-Fi, should still be considered. This should not be a challenge as having regular access to internet would be very feasible.

### 4.4. Learning Outcome

This project is expected to be a great positive and learning experience, to say the least. In fact, on my part, the amount of work that I would put on it would be quite significant as I will try to give my best to finish the project with all features implemented. This is especially challenging as the project is expected to have many functional requirements. Nevertheless, this would teach me valuable lessons. The project is also expected to benefit its users, specifically the Moroccan ones, by providing them with a platform which would enhance their education. Indeed, users will be able to learn at their own pace online rather than having to follow a teacher’s pace such as it is the case in a traditional classroom, which is critical to grasping concepts. Asides from that, this project could even be a start for a web application which may go into production one day and turn out to be a successful business which may become profitable. However, that would a far-fetched goal to consider at this point in the project, yet still a great one to keep in mind as the work goes along.

### 5. REQUIREMENTS SPECIFICATIONS

We would like to design and develop an online educational platform that shall fulfil the following functional and non-functional requirements:
5.1. Functional Requirements

5.1.1. Use-Case Diagram

The Use-Case Diagram is made of three user cases; the user can either be logged in as a student, teacher or administrator. For each case specific functionalities are enabled to the user. For instance, a teacher can create and upload different content to his or her class whereas the student can grade other students’ in order to get his or her own grades. For the manager or administrator of the web application, it is available to him or her the functionalities of managing the content to be put on the main page as well as to manage to manage the users of the web application by browsing through users and classes and having the choice to delete such content if deemed appropriate by him or her for instance.
5.1.2. As a general user

a) Users shall be able to browse lessons (videos) by level of education and subject;

b) Users shall be able to register either as a student or a teacher.

5.1.3. As a student

c) The user shall be able to watch different lessons and videos;

d) The user shall be able to practice and correct exercises.

e) The user shall get his grade on his assignment. His assignment will have to be graded by three students and his grade will be the average of the three corrections;

f) The user shall be able to register in classes.

g) The user shall remove himself/herself from classes.

5.1.4. As a teacher

h) The user shall be able to create classes.

i) The user shall also be able to invite or remove students to his/her class;

j) The user shall be able to add lessons. The lessons can also be removed;

k) The user shall be able to add videos and recommended content for each lesson. This content can also be removed by the user;

l) The user shall be able to add practices. These practices can also be removed by the user;

m) The user shall be able to grade his students’ assignments. The user shall also have his/her students grade each other’s works.
5.2. Non-Functional Requirements

5.2.1. Access Requirements
End users shall be able to use the application through one channel: Web. The Web-Academy application shall be developed as a web application offering asynchronous access to the end user through her web browser.

5.2.2 Integration Requirements
The application shall be able to integrate and bring together the component sub-systems into one system and ensuring that the subsystems function together as a system.

5.2.3 Security Requirements

5.2.3.1 Confidentiality Requirements
The confidentiality of all operations performed by users shall be preserved. In particular, network traffic confidentiality shall be protected.

5.2.3.2 Integrity Requirements
The integrity of all operations performed by users shall be preserved. In particular, users must be assured that they are connected to the right application.

5.2.3.3 Availability Requirements
The application shall be accessible when needed, with no single point of failure (SpoF). Moreover, the history of operations performed by each user shall survive system reboots and power failures.

5.2.4 Performance Requirements
Operations must be performed in less than one second.

5.2.5 Scalability Requirements
The application response time must remain under one second even with increasing loads. In particular, the initial number of users shall be around 10.000 with 100.000 operations a day. It is expected to reach 100.000 users in one year with 1 million operations a day.
5.2.6 Extensibility Requirements

Depending on the evolution of user and market needs, the application shall be extensible to support additional features such as displaying recommended classes per student level and the ability that the student can keep track of his progress on the platform.

Finally, the application has a preference for JavaScript technology, along with Node.js as a run-time environment on the server side.

6. Software Development Methodology

The methodology followed in the building of this web application was the Agile Method. “The Agile Method is a particular approach to project management that is utilized in software development. This method assists teams in responding to the unpredictability of constructing software. It uses incremental, iterative work sequences that are commonly known as sprints” [5].

This method was particularly significant because it fit my capstone experience as I had to continually learn about new technologies and adjust my design accordingly which meant adapting by going through sprint steps of the development in order to move forward.
7. Technology Enablers

The next phase will be deciding on the technological enablers for the project. This part of the project is important as it will show the programming languages, frameworks and run-time environments which will be used to develop the web application. These tools would be important as they will determine the quality of both the code and user experience.

[7] MEVN Stack

(MongoDB Express.js View.js Node.js)

7.1. JavaScript as a programming language

JavaScript has been used as the main programming language of both the front-end and back-end. This meant less burden in having to learn new programming languages as the same language could be used in coding both the server and client side. According to the Mozilla Developer Network (MDN), “JavaScript (JS) is a lightweight interpreted or just-in-time compiled programming language with first-class functions” [11]. Although it is most popular as a “scripting language for Web pages”, it can also be used by “non-browser environments, such as Node.js, Apache CouchDB and Adobe Acrobat” [11]. JavaScript is also considered “a prototype-based, multi-paradigm, dynamic language, supporting object-oriented, imperative, and declarative (e.g. functional programming) styles” [11].
7.2. Front-end Technologies

The technologies that have been used for the client-side are Vue.js as a JavaScript framework and Vuetify as a CSS framework. The choice of these two technologies has been mainly thanks to the fact that they are related, as they both share packages, as well as they are both popular frameworks for the front-end.

7.2.1. Vue.js

On the front-end the project will be built using a framework which is becoming more and more nowadays popular that is Vue.js. Vue.js is a progressive JavaScript framework which means that it can either be added only as part of an already existing server-side application which needs a richer and more interactive user experience or be utilized from the get-go to build more business logic on the front-end thanks to its core libraries and ecosystem which enable scalability. One of the few libraries which will be discussed in more detail include Vue-router and Vue-x.

Vue.js has three main characteristics which make it appropriate to be used: approachable, versatile, and performant [3]. Vue.js is approachable in the sense that its clearly written guide on their website makes it almost possible for everyone to start building the front-end of a web application even with only a basic and limited knowledge about HTML, CSS and JavaScript. Secondly, Vue.js is versatile thanks to its “incrementally adoptable ecosystem that scales between a library and a full-featured framework” [3]. And finally, Vue.js is also quite performant because of it high 20 Kb runtime compression file system using GNU zip software algorithm which makes it for a fast virtual random rendering that turns out quite useful in development mode [3].

7.2.2. Vuetify:

The CSS framework used in the designing the user experience is Vuetify. “Vuetify is a progressive framework that attempts to push web development to the next level” [13]. The decision to choosing this CSS framework over many others came down to four main reasons.
Firstly, Vuetify has one of the most vibrant communities as a CSS framework. Indeed, as they say it on the Vuetify website, “Developing with Vuetify means you will never be alone” [14]. This is especially important in the case of this software product knowing that I do not have much experience with CSS frameworks. Therefore, this meant that solutions to almost any problem faced could be found, by reaching out to this active community.

Secondly is that “Vuetify is built from the ground up to be semantic” [14]. This makes it “easy to learn and to remember” [14]. Again, this is particularly interesting and fitting to this project as there is a lack of familiarity with CSS frameworks and how they work. This meant a great learning curve.

Thirdly, Vuetify has regular updates and renovations. It has a committed “development team” that always strives towards providing “the best experience” a user can have. Therefore, this allows for easy bug fixing and continuous enhancement [14].

Last but not least, and which ties back to all the previous points, is the fact that Vuetify benefits from an “amazing support” [14]. Indeed, “Vuetify is patched weekly, responding to community issues and reports at breakneck speed”. This entitles its users to feel confident “knowing” that their applications “will be supported and updated frequently with fixes and new features” [14].

### 7.3. Back-end technologies

On the back-end, the technologies that have been chosen to be used are also JavaScript based: Node.js and Express.js. Part of choosing them was the familiarity that has been created with JavaScript by using it on the front-end side, which makes it not mandatory to use or learn a new language. Another reason was that these technologies are also widely used and popular nowadays which meant better reliability and documentation.

#### 7.3.1. Node.js

Node.js is a “JavaScript runtime built on Chrome's V8 JavaScript engine” [1] for the server side. Node.js is especially performant and scalable through this simple illustration of “hello world”
In the code example, “many connections can be handled concurrently. Upon each connection the call-back is fired, but if there is no work to be done, Node will sleep. This is in contrast to today's more common concurrency model where OS threads are employed. Thread-based networking is relatively inefficient and very difficult to use. Furthermore, users of Node are free from worries of dead-locking the process, since there are no locks. Almost no function in Node directly performs I/O, so the process never blocks. Because nothing blocks, scalable systems are very reasonable to develop in Node” [1]. Moreover, the fact that Node.js is only implemented without threads doesn’t make it impossible for the utilization of multiple cores in an environment. Indeed, the child process can be created using the “child_process.fork() API”. In addition to that, the communication between different child processes is designed to be feasible [1].

7.3.2. Express.js

On the other hand, Express.js is “a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications” [4]. As it was the case with other frameworks, Express.js is also high in performance in the sense that “Express provides a thin layer of fundamental web application features, without obscuring Node.js features” [4] that
are widely used and efficient. The creation of robust APIs is also easy and quick with Express.js as it makes at the user’s disposal “a myriad of HTTP utility methods and middleware” [4]. Last but not least is that many of the most popular framework are based upon Express [4].

### 7.3.3. MongoDB

As for the database side, there has been the choice to make use of MongoDB. The decision has come down to the fact that MongoDB is also one of most popular document-based databases. It is also the “best way to work with data” in the sense that “MongoDB’s flexible document data model makes working with data intuitive”, whether it is for the building of an app from scratch or updating an existing one [2]. MongoDB is also able to run anywhere making it feasible for a flexible data migration with no deployment hurdles [2].

### 8. Software Architecture

![Software Architecture Diagram]

The architecture of this web application includes two main sides: the back-end and the front-end side. For the backend, or the server side, it includes the Web container and the Business
container. The Web Container consists of the policies, controllers, middlewares and routes which are each structured in different folders in the project. The policies and controllers are the logic or code that it is executed at each web service endpoint or route. Middlewares, on the other hand, are a list of functions “which contribute to adding necessary side-effects to an application flow such as making database calls, asynchronous API calls, logging, tracing, etc..” [12]. For the Business Container, it is made of the models and schemas which represent the server’s data. The Object Data Modeling (ODM) for our schemas is constructed using Mongoose which is a “library for MongoDB and Node.js ”[8] that “manages relationships between data, provides schema validation, and is used to translate between objects in code and the representation of those objects in MongoDB” [8].

The back-end technology used for our project is Node.js which acts as a runtime that enables for using JavaScript code on the back-end. The database used was MongoDB which is a non-relational document based database.

For the frontend side, there has been the use of Vue.js which is a popular frontend framework. Within Vue.js, there are different folders which each serve a different purpose for the
application. Firstly, there are Components which are “reusable Vue instances with a name” [9]. Commonly, a frontend Vue.js app is made of a bunch of components which are usually used separately throughout the app. Secondly, there are Services which consist of calls to a specific web service endpoint through an HTTP request in order to either get data or perform any of the requests following the HTTP protocol. The bridge of HTTP requests between front-end and back-end is used by applying the Axios API which is “a promise-based HTTP client that works both in the browser and in a node.js environment. It basically provides a single API for dealing with XMLHttpRequests and node’s http interface” [10]. Another important package that has been installed is the Vuex store. This module is particularly interesting because it allows us to save the state of the application through the use of different actions and mutations which we would want to execute whenever there is a need to save some added data to the state. For instance, a popular use is the persistence of the user’s state once he or she is logged in. Through this persistence the app is made aware of a particular user logged in and is thus able to display the interface to him or her accordingly. And last but not least are methods which allow for the dynamic adjustment of data or interface that is displayed actively to the user once he or she clicks on a button for example.

9. Software Design

The following phase will be the architecture of the application. This phase is pivotal as it will show the whole structure of the app by using modelling tools in order to build diagrams which will illustrate the relationships between different nodes in the structure.
9.1. Class Diagram

Looking at this Class Diagram (CD), we can easily notice the different relationships that are present between different classes in this project. For instance, one interesting one would be the nested relationship of the Class and how it is related to the Unit class which is related to Lesson class that in turn is related to another one of Exercise. Having this class diagram has been particularly useful and fitting for this project since we used a non-relational database; this meant that an entity relationship diagram (ERD) would not be inappropriate since in a non-relational database there is no dealing with entities which merely consist of attributes such as it is the case of an SQL relational database.
9.2. Sequence Diagrams

9.2.1. Teacher Adding Classes

This sequence diagram represents all the actions and steps that the user as a teacher has to go through in order to be able to successfully add a class. The teacher will receive a notification letting him or her know of a successful class creation.
9.2.2. Student Accessing Classes

The sequence diagram above shows the sequence of actions or steps that the user as a student has to go through in order to access and add classes. An important requirement that should be noted is that the student has to be logged in prior to trying to adding any class otherwise he or she will be requested to log in, as depicted in the sequence diagram.
10. Software Implementation Results

Landing page of the web application designed with Vuetify as a CSS framework. The Navigation bar at the top would allow an easy navigation to different functionalities of the application such as Log In and Sign Up.

The same Landing.vue component when scrolling down. This Vue allows the user to see all available classes by Subject. It also has a filtering option by level of education, enabling the user to only see classes fitting his or her level of education.
Login.Vue where the user can Log In if he is already registered. Otherwise can choose to sign up by click on the sign up instead button.

Register.Vue form where the user can register by entering his or her credentials accordingly. The user can also choose to register as either an instructor or student in this form, which each would enable the user different features and functionalities.
This button appears on the Landing.vue when the user either registers or logs in as an instructor. By clicking on it the user as an instructor would be able to get to a Vue Component for creating a class.

The green button enables the user to navigate to a Vue component which would have all the classes respective of the subject with descriptions and class structures.

### Create Class

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class Title*</td>
<td></td>
</tr>
<tr>
<td>Subject*</td>
<td></td>
</tr>
<tr>
<td>Class Subject*</td>
<td></td>
</tr>
<tr>
<td>Level of Education</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td></td>
</tr>
</tbody>
</table>

The form for allowing the user logged in as an instructor to create a class with its needed information: title, subject, level of education and description.
The Vue component to allow the user to see all the classes of a given subject with their respective description as well as a button to enable the user to navigate to the class structure.

The Vue component which would enable the user as an instructor to see his or her created classes. The instructor can also click on the button next the title of the class to be able to access the whole class structure and description.
The Class.Vue component for allowing the user as an instructor to access the whole class structure. In this component, the user can edit the class by adding units and lessons, with their respective learning and practice parts.

In this form, the user as an instructor can edit the lesson by adding a learning part which consist of a title and URL link to a video. In the practice part for lesson, the user can choose the type of practice: file practice or multiple choice answer.
For the file practice, the user as an instructor has to fill in the title and practice link if available. In this option, the user has also to upload a file for the practice which would basically consist of the assigned problem for the lesson.

In this form, the user as an instructor can create the multiple choice quiz for the lesson. The red button would allow him or her to delete the respective chosen option. The blue add button would enable to add more options as possible for the multiple choice question. Finally, he or she can choose the correct option by entering its respective index.
The form for creating a unit for the class would enable the user to fill in the title and description for it. Upon creation of the unit, a message would be displayed to inform the user of the successful creation of the unit. This message is also displayed to the user when successfully creating or adding any other data to the class such as lessons, practices or learning parts.

The form for allowing the user as an instructor to create a lesson for the unit of a class. The user should fill in the title and description of the lesson. The instructor would be notified of the successful creation of the lesson once he or she clicks on the ‘CREATE LESSON’ button.

When logged in as a student, an additional button ‘MYANSWERS’ is shown on the header to allow the user to view his or her answers. The student will also have a ‘MYCLASSES’ button which would enable him or her to all the classes that he or she has registered in.
The Unit.vue component for units which allows the user to see all the units of the class, after clicking on the class he or she has chosen from the Landing.vue component of the main page. Each unit can also have a particular picture on top of its title which would act as a description or representation of its content.

The Lesson.vue component for the class lists all the lessons of the unit with their respective learning and practice parts. All lessons will be listed on the side of the component.
The modal component for viewing the lesson. The Vue-Youtube-Embed package was used in order to enable the display of the video lesson within the web application, instead of having the user access it through a Youtube link which would have him or her exit the web application.

A modal that would allow the user as a student to answer file quizzes by enabling to upload his or her file as an answer. The student can also choose to type in the answer as an alternative. Once his or her answer submitted, the student will be asked to grade similar quizzes in order for him or her to get tokens which would qualify him or her to get his or her own grades afterward.
Once the student has answered, he or she will be provided with the option of grading other students' answers in order to get tokens which would allow to get his or her own grades. This component will have the task as well as the right answer for it to allow the user to be able to grade the answers rightfully.
The StudentAnswers.vue component for the user as a student to view his or her answers with the option to see their grades which would require the use of a token by the user. The student can also see if he has already seen the grade and how many of the three grades are already available.

In the StudentAnswer.vue component the student can see one of his or her answers in detail; with how much is his or her grade so far, the task for the practice, the right answer for it as well as his or her uploaded answer.
In the StudentClasses.vue component the student can see all his or her classes in which he or she is enrolled in. A button is available to the user in order to see further information about the class.
11. Conclusion

This project has been both a great learning experience and a chance to showcase the knowledge and skills that I learned throughout my education at AUI. The capstone also allowed me to get further hands on experience in the field of software development. In fact, through this project I was able to learn more about new technologies such as Vue.js and Node.js. Not only that but I was also able to apply related techniques and practices that I learned about them in the context of a project. Moreover, other skills had to also come into play such as a needed eager and adversity to learn and face the challenges throughout my building of this web application.

Following an Agile Incremental Model meant that I had to deal with changes and adapt myself to new requirements and design adjustment almost every step of building the software product. However, adopting this Agile methodology was rewarding in terms of flexibility and coming up with new adaptations to arising issues.

All in all, this capstone project has been one of the best learning experiences I had at AUI. Indeed, I had to apply many of the skills that I learned throughout my journey at the university as well as to challenge and adapt myself to new ones which I had to acquire in order to successfully complete this project. Finally, as I expect to follow up on working on this project in the future, this positive learning experience can only extend and further help improve my software development skills and knowledge.
12. Future Work

Being developed in less than three months with the added challenge of learning new material and a lacking experience in the field, this project could be significantly improved and enhanced if given more time and dedication in the future. Indeed, following are some of the functionalities and enhancements that could be added in the future:

- The ability for the student to communicate with his or her peers as well with the professor could be of significantly added value. This would strengthen the bond and forging of a community that would be dedicated to helping each other to learn and educate about new material.

- Another added feature would be to enable the users to rate and create reviews for the classes they enrolled in. This can help direct new users towards better content as well as encouraging the instructors for improving and putting more work into the creation of their classes, as they will eventually know that their classes would be rated and reviewed by users. This will motivate the community to strive towards excellence.

- Another option that can be added to encourage quality content to be displayed on the main page is the inclusion of a paid feature for the teachers which would allow them to have their created classes displayed on the main page. The profit made off the paid features can be used to better enhance the application or be put into a noble cause such as a program for supporting education.

- And last but not least is to enhance the user experience. This would ask for an analysis of the issues and hurdles that the users face while using the software product and then change the interface accordingly.

Ultimately, the list of ideas can only be further stretched and improved along the production and deployment step of this application which would be the next phase.
13. References


