

University of Nevada, Reno  
Department of Computer Science and Engineering

Master CS  
*Project Part 2: Revised Specification and Design*

**Team 01**  
Crystal Atoz  
Kayla Garin  
Austin Schrage  
Yan Shore  
Zhuqi You

Instructors: Dr. David Feil-Seifer and Dr. Devrin Lee

External Advisor: Professor Ben Brown and Dr. Bill Doherty

February 18, 2022

## Table of Contents

1.	Abstract .....	3
2.	Recent Project Changes.....	3
3.	Updated Specification	
3.1.	Summary of Project Specification Changes.....	4
3.2.	Updated Technical Requirements.....	4
3.3.	Updated Use Case Modeling.....	5
3.3.1.	Use Case Diagram.....	6
3.3.2.	Use Case Descriptions.....	6
3.3.3.	Detailed Use Cases.....	6
4.	Updated Design	
4.1.	Summary of Project Design Changes.....	8
4.2.	Updated high-level and medium-level design.....	9
4.3.	Updated User Interface Design.....	10
5.	Updated Glossary of Terms.....	25
6.	Engineering Standards and Technologies.....	27
7.	Project Impact and Context Considerations.....	28
8.	Updated List of References.....	29
9.	Contributions of Team Members.....	31

## **Abstract**

The team's project is a web application focused on providing education on various computer science topics. This project will provide each user an interactive learning environment tailored to them. The project's importance evolves from the recent pandemic that forced learners to shift to online learning. Our team hopes that by providing an interactive learning experience the user is engaged and motivated. This web application will support many features that will consist of providing user statistics, achievement badge system, topic quizzes, a variety of courses to choose from, autograder, discussion forum, certifications, and an individual improvement plan. This document goes over a revised specification and design of the project, glossary of terms, engineering standards and technologies, project impact and consideration, list of references, and contributions of team members.

## **Recent Project Changes**

The team's progress since project assignment #1 has mostly been restructuring the organization of the project timeline and solidifying features to implement. In order to meet deadlines, the team has created tasks for each member and a timeline for when those tasks need to be completed. The team has decided that each member will complete a feature every 2 weeks. This will help the team get enough features done for the web application to be completed on time. The team also met with Devrin Lee in a progress meeting. There we discussed our roadblocks and required features for our software.

## **Updated Specification**

### **3.1 Project Specification Changes Summary**

The main changes that were made to the requirements include an administrative profile view, profile statistics, course statistics, and a discussion board. These changes were made to ensure the application has enough features. For non-functional requirements, we added utilizing Vue.js, Node.js, and MySQL. This change was made to keep the code more organized and useful.

### 3.2 Updated Technical Requirements Specification

#### *Functional Requirements*

Functional Requirements	
Level	Functional Requirement
1	<p><b>FR1:</b> User will have the ability to create an account.</p> <p><b>FR2:</b> User will have the ability to login/logout of the account.</p> <p><b>FR3:</b> User will be able to see a list of all courses.</p> <p><b>FR4:</b> User will be able to add courses and see their added courses.</p> <p><b>FR5:</b> User will be able to remove courses they are enrolled in.</p> <p><b>FR6:</b> User will be able to watch course videos.</p> <p><b>FR7:</b> Course modules will unlock after successfully completing the previous module.</p> <p><b>FR8:</b> User will be able to take course quizzes and view the correct score after submitting.</p> <p><b>FR9:</b> User profiles will successfully save all progress made.</p> <p><b>FR10:</b> User will be able to view all earned and unearned achievements.</p> <p><b>FR11:</b> User will have an achievement added to their profile when completing specific tasks.</p> <p><b>FR12:</b> User will be able to view the percent of achievements they have earned.</p> <p><b>FR13:</b> The personality quiz will output the correct learning style and recommended courses based on user input.</p> <p><b>FR14:</b> User will be able to view their overall statistics for an individual course.</p> <p><b>FR15:</b> User will be able to register for an available event</p> <p><b>FR16:</b> The application will display the number of available spots in an event.</p> <p><b>FR17:</b> Admin accounts will be able to add and edit courses and events</p> <p><b>FR18:</b> Admin accounts will be able to see overall course statistics on all users.</p> <p><b>FR19:</b> Admin accounts will be able to add achievements</p> <p><b>FR20:</b> User will be able to view their profile statistics.</p>
2	<p><b>FR21:</b> User will be able to choose which badges to display to other students.</p> <p><b>FR22:</b> User will be able to see a ranking system for courses.</p> <p><b>FR23:</b> User will be able to filter through events.</p> <p><b>FR24:</b> User will be able to post to a discussion forum.</p> <p><b>FR25:</b> User will be able to rate modules/courses.</p>
3	<p><b>FR26:</b> User will be able to upload their own videos for approval.</p> <p><b>FR27:</b> User will be able to create their own tests for approval.</p> <p><b>FR28:</b> User will be able to create their own progress badges.</p> <p><b>FR29:</b> User will be able to add other users as friends.</p> <p><b>FR30:</b> User will be able to view the course progress of other users that are friends.</p>

	<b>FR31:</b> User will be able to participate in monthly coding contests and view rank. <b>FR32:</b> User will be able to rate videos
--	------------------------------------------------------------------------------------------------------------------------------------------

*Non-Functional Requirements*

**Non-Functional Requirements**

- NFR1:** Application will use Vue.js, Node.js, and MySQL.
- NFR2:** Application downtime will be limited to 1-hr per day from the hours of 9:00-15:00.
- NFR3:** Interface will be intuitive for non-technical users.
- NFR4:** Gamification features will be engaging.
- NFR5:** Video length will be limited to keep it engaging.
- NFR6:** Account creation process will be simple.

### 3.3 Use Case Modeling

#### *Use Case Diagram*

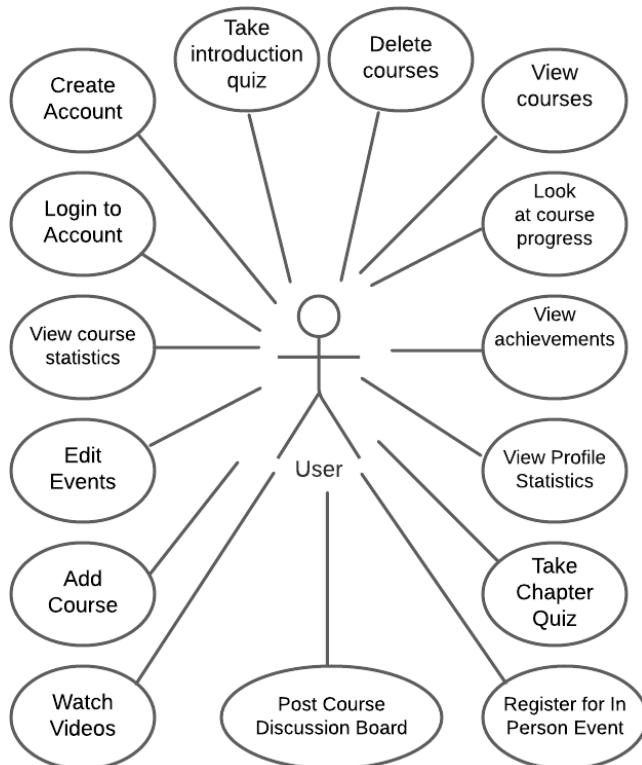


Figure 1: Use Case Diagram for Master CS

### *Use Case Descriptions*

<b>Use Case Descriptions</b>		
UC01	Create Account	On accessing the website, a user is able to create an account in order to view additional content, take courses, and save data. The user clicks on the create new account link and follows the instructions.
UC02	Login to Account	When a user accesses the website, the user is prompted to login with existing credentials. The user then enters their username and password and the server compares what was submitted to what is in the authentication database. The user will then have access to their profile and saved data.
UC03	View Course Statistics	Administrative accounts will be able to see the amount of users attempting the course. The admin will be able to see how many completed the course and statistics on the course such as overall grade. Regular users will be able to see their individual course statistics.
UC04	Edit Events	Users that have an administrative role can add and edit events. The administrator will be able to add events, set the amount of maximum attendees, and edit event details.
UC05	Add Course	The user selects what course they would like to learn. Once selected, the user will be redirected to that course's landing page and will be able to add the course to their courses.
UC06	Watch Videos	The user will be shown a list of available content for the selected course. Each video will be loaded and playable on the website.
UC07	Post on Course Discussion Board	If the user has any questions about the content they have viewed, they will have the ability to post their questions in the social area. The user will be requested to share if the answer they received was helpful.
UC08	Register for In Person Event	The user will be able to view a list of future events. The user will be able to register for an event to attend in person.
UC09	Take Chapter Quiz	After completing a module, the user is directed to a chapter quiz page. Based upon the results of the quiz, the user either advances to the next module or is given additional resources to better understand the content they are having difficulties with.
UC10	View Profile Statistics	The user will be able to navigate to their profile page and see their statistics. They will be able to view statistics on courses

		enrolled in, courses completed, modules completed, and total scores.
UC11	View achievements	The user will be able to view earned achievements and achievements that can be earned on their profile. The achievements they earn will have a badge and description. The user will be able to access their achievements through their profile page.
UC12	Look at course progress	The user will be able to navigate to a course page and view their course progress. They will be able to view the percentage completed, grade for the course, and topics within the course completed.
UC13	View courses	The user will be able to view the courses they have added. The user will be able to click on a course to navigate to the course page.
UC14	Delete courses	The user will be able to delete courses by navigating to the main course list and clicking delete. The course will then be removed from their course list.
UC15	Take introduction quiz	Upon creation of the user's account, the user will be able to take a multiple choice introductory quiz. The quiz results will show the user what courses are recommended to them based on their interests and what learning style they have.

*Requirement Traceability Matrix*

	F R 1	F R 2	F R 3	F R 4	F R 5	F R 6	F R 7	F R 8	F R 9	F R 10	F R 11	F R 12	F R 13	F R 14	F R 15	F R 16	F R 17	F R 18	F R 19	F R 20	F R 21	F R 22	F R 23	F R 24	F R 25	
UC 1																										
UC 2																										
UC 3																										
UC 4																										
UC 5																										

## Updated Design

#### **4.1 Project Design Changes Summary**

The project design of this semester will be in an upgraded state compared to that utilized last semester. For the changes of the project, we will elaborate from two aspects: one is from the features, the other is from the technology. The first is features. This semester we will add more content: User goals / progress box: a box that shows the user's progress in learning and any goals they want to achieve. User statistics: shows how many users have completed a module / course and how long it took them to do so. Provide statistics on courses and grades (i.e. the percentage of users who completed the course). Improvement plan: after a test or assignment, show what needs to be improved. Additional videos or articles are recommended to master this topic. Discussion Forum: Discussion of each module or course. Users can post questions or suggestions. Then, grading systems for video and modules. These aspects will allow users to

improve their skills while immersed in a fully responsive application which will continuously be enhanced through monitored technical updates.

For technical updates, we will use database technology this semester to ensure that users can register and log in. Then, we also updated our website development tools, for the back-end part, database middle ground: Node.js. For example, Node.js using express will be able to run MySQL queries on the database and provide endpoints that Vue applications can access using Axios. For the front-end part, we use Vue, which can make the development of web-based front-end applications more convenient, because Vue has explicit, responsive data binding and component-based development.

## **4.2 Updated high-level and medium-level design**

### **System-level Diagrams**

In the update of this semester, our team will add more content, including content delivery, user participation and personal learning. The context diagram in Figure 1 shows the high-level components of the education website project. The User Registration System is responsible for managing new users and making sure that they meet the requirements set. The User Account System manages the account particulars that are associated with a user in the system. For example: courses enrolled, email, and phone number. The Group Communication System enables users to share information with other users in a convenient area on the website. The Event Scheduling System automatically checks for availability and suggests times that are available. In the future, physical meetings can be scheduled with this module. The Security System is responsible for validating user credentials and granting access to modules that the user has access to. The File Database System stores all text data from the website and user information. The Video Database System stores the course video content.

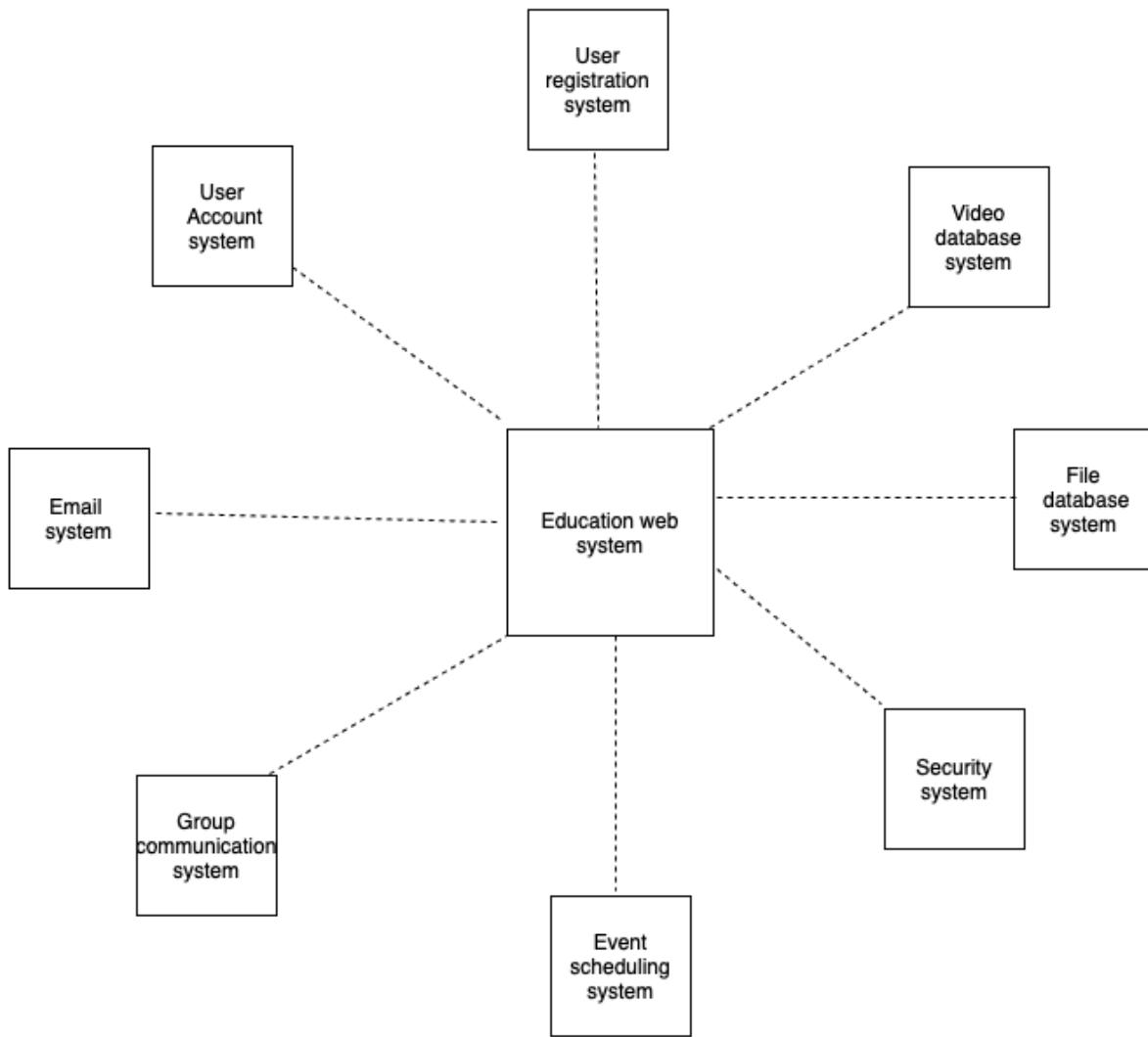


Figure 1: The system level diagram of the education web system.

## Software in program units

Table 1: The table lists the program units under the Account module.

<b>Account</b>	This module contains all information related to user accounts.
<b>getUser()</b>	This member gets the user table from the database containing all the users registered to the site.
<b>checkUser()</b>	This member checks if the user is registered.

	If the user is not, they are directed to create an account.
<b>createAccount()</b>	This member creates an account and stores the information into the user table.
<b>getPassword()</b>	This member gets the password from the user table.
<b>checkPassword()</b>	This member checks if the password inputted is correct. If it is incorrect, it will prompt the user to try again.
<b>getPhoneNumber()</b>	This member gets the phone number from the user table.
<b>getEmail()</b>	This member gets the email from the user table.

Table 2: The table lists the program units under the CreateAccount module.

<b>CreateAccount</b>	Members create an account for login the website
<b>setUserName()</b>	The user sets up a specific username to access the website
<b>setPassword()</b>	The user sets up a specific password for identifying account login.
<b>setEmail()</b>	The user sets up an email to retrieve their username or password.
<b>setPhoneNumber()</b>	The user sets up a phone number to login or retrieve password.

Table 3: The table lists the program units under the Quiz module.

<b>Quiz</b>	This module contains all information related to quizzes.
<b>checkCompletion()</b>	This member checks if the lesson is completed. If so, the quiz can be displayed.

<b>getQuestions()</b>	This member gets questions.
<b>getAnswers()</b>	This member gets answers in relation to what quiz was selected.
<b>checkAnswers()</b>	This member checks if the answers are correct.
<b>getStatistics()</b>	This member gets the statistics from the quiz and stores them.
<b>createSummary()</b>	This member creates a summary of the quiz based on the statistics.
<b>displaySummary()</b>	This member displays the summary of the quiz to the user.

Table 4: The table lists the program units under the Video module.

<b>Video</b>	This module contains all information related to videos.
<b>displayVideos()</b>	This member displays all the videos to the user based on course selection.
<b>getVideo()</b>	This member gets the video that was selected by the user.
<b>checkCompletion()</b>	This member checks if the video is watched.

Table 5: The table lists the program units under the Badge module.

<b>Badge</b>	This module contains all the information related to the badges gained by the user.
<b>getBadge()</b>	This member gets all the badges.
<b>checkRequirements()</b>	This member checks if the user meets all requirements to have the badge.
<b>checkBadge()</b>	This member checks what badges the user has. If they do not have it, it is greyed out.
<b>displayBadges()</b>	This member displays all the badges regardless if the user has it or not.

Table 6: The table lists the program units under the event schedule module.

<b>EventSchedule</b>	This module contains all the information related to the user's schedule.
<b>getUserSchedule()</b>	This member gets the schedule of the user.
<b>displayUserSchedule()</b>	This member displays the user's schedule.
<b>checkSchedule()</b>	This member checks if the schedule is correct in relation to the user.
<b>removeFromSchedule()</b>	This member removes anything the user selected from the schedule and removes it from the user's account.

Table 7: The table lists the program units under the Course module.

<b>Course</b>	This module contains all the information related to courses.
<b>getCourses()</b>	This member gets all the courses from the course table.
<b>displayCourses()</b>	This member displays all the courses to the user.
<b>getUserCourses()</b>	This member gets all the courses that the user is working on.
<b>displayUserCourses()</b>	This member displays all the courses to the user that they are working on.
<b>checkCompletion()</b>	This member checks if the user has already completed the course.
<b>addCourses()</b>	This member adds whatever course the user selected to their account and stores it.
<b>removeCourses()</b>	This member removes whatever the user selected from their account.

Table 8: The table lists the program units under the Security module.

<b>Security</b>	This module contains all the information related to the security of the website.
<b>authenticateUser()</b>	This member authenticates the user before they can log in to the website.
<b>verifyAccess()</b>	This member verifies the user access if they navigate to a new page or they are about to become inactive.

## Data Structure

Our project will be employing Node.js as our base back-end language, so our primary data structures will be the classes that are described above. For the structures that will contain a high number of records: Videos, Accounts, Quizzes, Events and Achievements. We plan to employ a database to handle each individually. Figure 1.1 shows an example entry for the video database. The entry consists of an ID that can be used as a primary key. A category that explains the video content, the video name, and the length consisting of long, medium or short. Figure 1.2 shows an example entry for the quiz database. ID is once again used as a primary key. Category also corresponds to the content of the quiz (same as video), with quiz name identifying the individual quiz and difficulty once again falling into three categories of Easy, Medium, and Hard. Finally, Figure 1.3 represents an example account database entry. Since we plan on handling most of the account information inside the class itself, this entry only requires the ID primary key and the name of the account holder. Figure 1.4 is about events. Users can choose to register according to the time of the event and the information of the instructors, and users can also choose to cancel the registration. Figure 1.5 is about achievements, when users complete the course or reach a certain level of learning progress, they will be rewarded with badges, which will encourage users to learn.

Figure. 1.1

ID	Category	Video Name	Length
----	----------	------------	--------

Figure. 1.2

ID	Category	Quiz Name	Score
----	----------	-----------	-------

Figure. 1.3

ID	Email address	First name	Last name
----	---------------	------------	-----------

Figure 1.4

Events	Time	Instructors	Description
--------	------	-------------	-------------

Figure 1.5

Achievements	Title	Description
--------------	-------	-------------

### 4.3 Updated User Interface Design

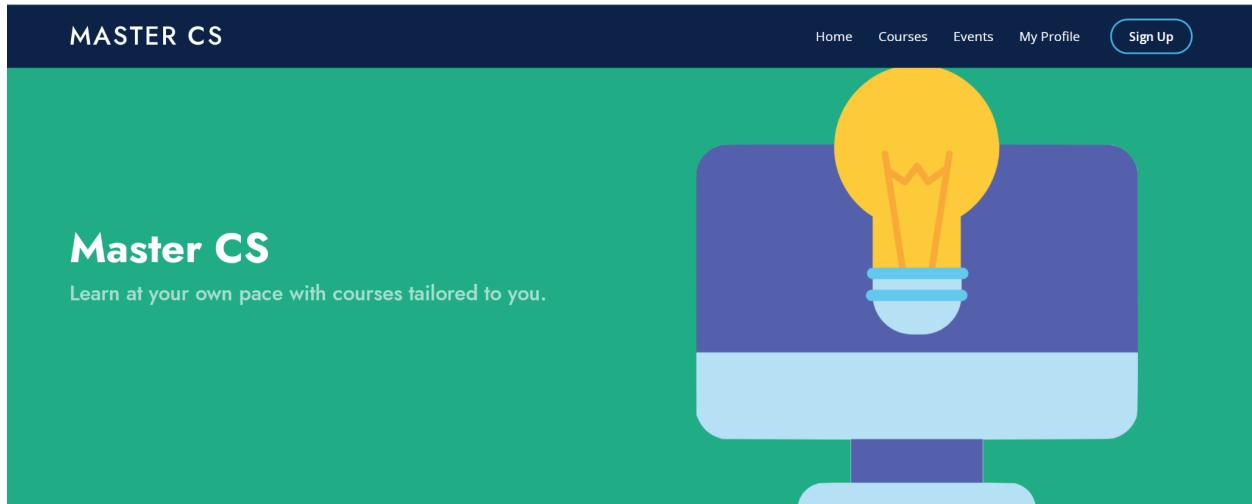
At present, the prototype development functions we have completed include five points:

- Home Page
- Courses:
  - Video
  - Quiz
- Event Registration
- Achievements Page
- Sign-Up + Login Page

#### Home Page

In the dynamic home page, our navbar includes “Home”, “Course”, “Events”, “My Profile”, and “Sign Up”. In the middle of the home page is “About”, which shows the purpose of the website.

The following is an introduction to all our Team 1 team members and teaching staff.



The image shows the "ABOUT" page of the Master CS website. At the top, there's a section titled "ABOUT" with a short description: "Generate your own personalized learning plan with our courses." Below this is a bulleted list: "• Compete with other users while learning", "• Choose to learn topics online or in person", and "• Take an introductory quiz to personalize your experience". To the right, there's another section with text: "Master CS is a platform dedicated to helping users learn. Course layouts are designed to fit with your schedule and learning style. We made learning fun and competitive through educational games and achievements you can earn and show off. Have fun learning with Master CS!" Further down, there's a section titled "TEAM 1 MEMBERS" featuring six team members in a grid. Each member has a circular profile icon with a stylized letter 'N', their name, and the word "test" below it. The names listed are Crystal Atoz, Kayla Garin, Austin Schrage, Yan Shore, Zhuqi You, and Advisors.

Figure 1: Home page

## Courses

This is the course homepage page. We have set up 6 important CS subject courses. The main function of the course homepage is to display all of the courses so the user can navigate to the course of their choosing.

The screenshot shows the 'COURSES' section of the Master CS website. It features four course cards:

- Cyber Security:** Features a hand interacting with a digital interface showing padlocks and keys. Includes a green 'Start Video' button and a yellow 'Quiz' button.
- Data Science:** Features a hand interacting with a digital interface showing gears and charts. Includes a green 'Start Video' button and a yellow 'Quiz' button.
- Big Data:** Features a hand holding a smartphone with various data visualization icons (bar charts, pie charts, etc.) floating around it. Includes a green 'Start Video' button and a yellow 'Quiz' button.
- Machine Learning and AI:** Features a blue brain icon integrated into a circuit board. Includes a green 'Start Video' button and a yellow 'Quiz' button.

Each course card includes a brief description and a call-to-action for both video and quiz access.

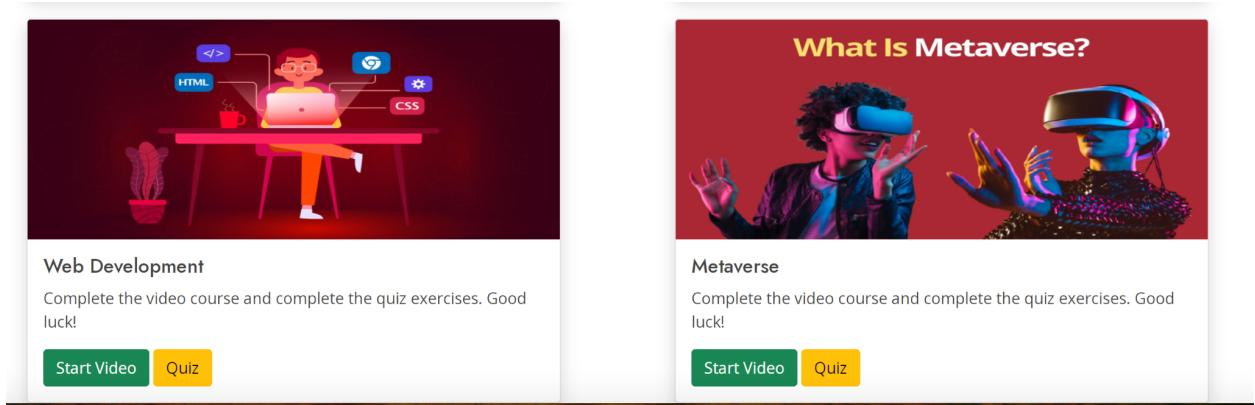


Figure 2: Courses page

## Course Videos

This is the playback function of course videos. This semester, we insert a listed video. This semester, we will implement the course videos by embedding the links using the Youtube API.

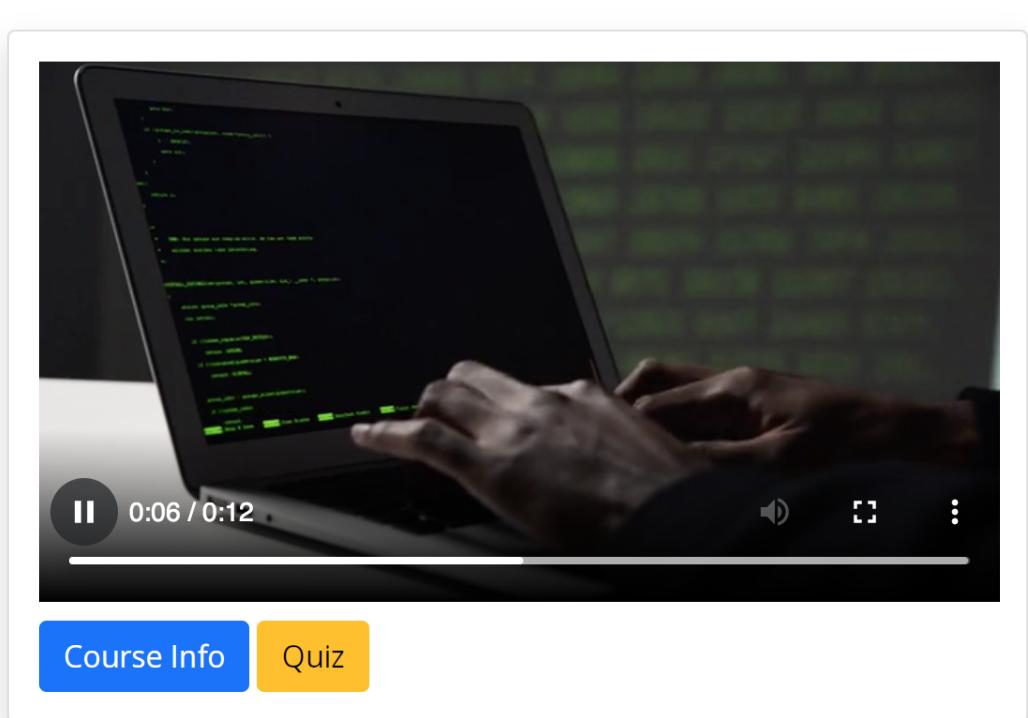


Figure 3: Video

## Course Quiz:

This is the course quiz function. We have designed a sample quiz with five questions, which mainly shows three aspects.

The screenshot shows a web-based quiz interface. At the top, there's a dark header bar with the text "MASTER CS" on the left and navigation links "Home", "Courses", "Events", "My Profile", and a "Sign Up" button on the right. Below the header, the main content area has a title "Simple Quiz About Course" and a subtitle "Test your knowledge !". The quiz consists of five numbered questions with multiple-choice answers. Question 1 asks about the HTML element for JavaScript code, with options A, B, C, and D. Question 2 asks about an external JavaScript file, with options A, B, C, and D. Question 3 asks how to write "Hello" in an alert box, with options A, B, C, and D. Question 4 asks if JavaScript is related to Java, with options A and B. Question 5 asks about the start character for variables, with options A, B, C, and D. At the bottom of the quiz area is a "Submit Answers" button. The footer of the page contains the text "Master CS © 2021, all rights reserved".

1. In which HTML element do we put in JavaScript code?

Oa. <js>  
Ob. <script>  
Oc. <body>  
Od. <link>

2. Which HTML attribute is used to reference an external JavaScript file?

Oa. src  
Ob. rel  
Oc. type  
Od. href

3. How would you write "Hello" in an alert box?

Oa. msg("Hello");  
Ob. alertBox("Hello");  
Oc. document.write("Hello");  
Od. alert("Hello");

4. JavaScript is directly related to the "Java" programming language

Oa. True  
Ob. False

5. A variable in JavaScript must start with which special character

Oa. @  
Ob. \$  
Oc. #  
Od. No Special Character

Master CS © 2021, all rights reserved

**Fig 4: Quiz**

When users submit a quiz, the score will automatically show.

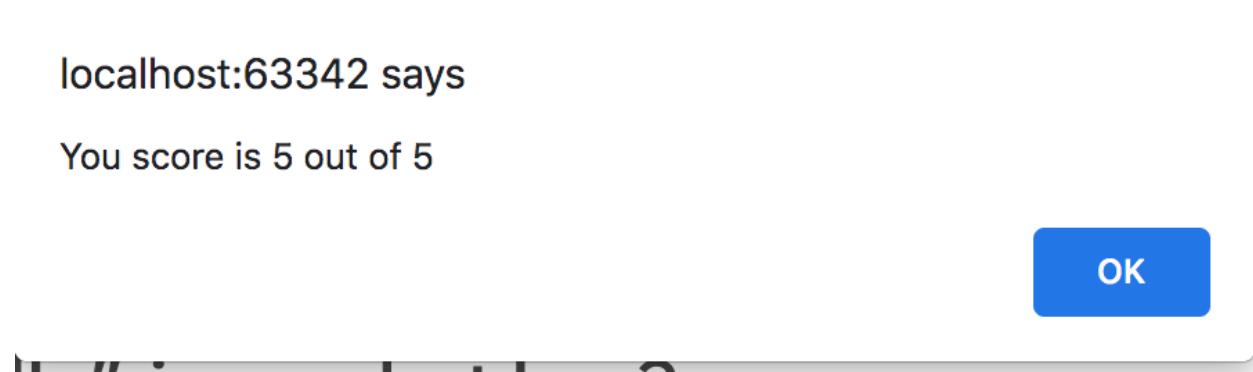


Fig 5: Quiz score display

When users do not select any questions to submit a quiz, the system will display that you have not chosen to answer all the questions.

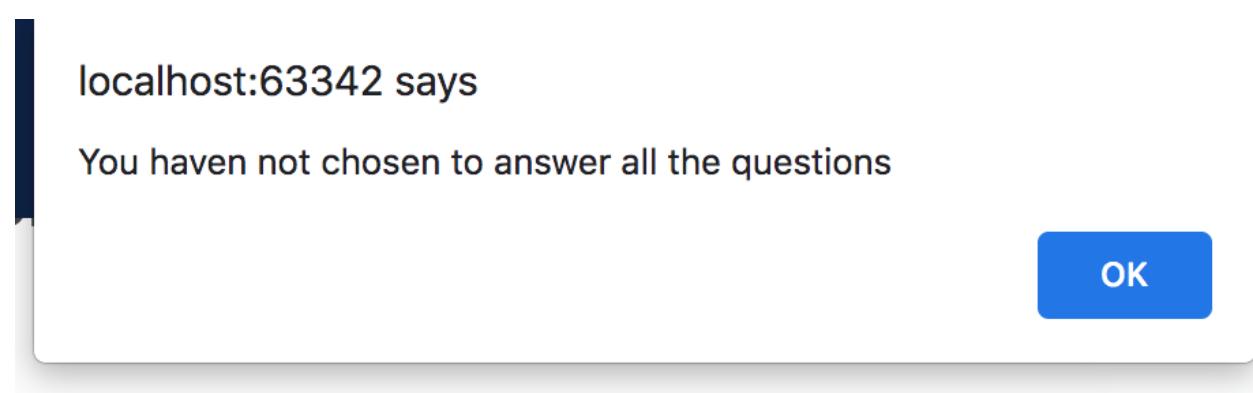


Fig 7: Quiz score display

When users choose one of the questions to submit and do not choose to answer all the quiz questions, the system will remind users again that you have not chosen to answer all the questions.

### **Event Registration**

This is the event page. Users can register according to their favorite events that they want to attend in person.

The screenshot shows a web-based events registration system. At the top, there's a dark header bar with the text "MASTER CS" on the left and navigation links "Home", "Courses", "Events", "My Profile", and a "Sign Up" button on the right. Below the header is a green banner with the text "UPCOMING EVENTS". The main content area displays six event cards, each featuring a blue square icon with a white 'N' logo, the event title, professor information, date, and a "Register" button.

- Git and Github**  
Professors: Sergiu Dascalu and Vinh Le  
January 1st 2022 @ 2:00p.m.  
[Register](#)
- Agile Software Development**  
Professors: Sergiu Dascalu and Vinh Le  
January 2nd 2022 @ 2:00p.m.  
[Register](#)
- System Modeling**  
Professors: Sergiu Dascalu and Vinh Le  
January 3rd 2022 @ 2:00p.m.  
[Register](#)
- Software Evolution**  
Professors: Sergiu Dascalu and Vinh Le  
January 4th 2022 @ 2:00p.m.  
[Register](#)
- Project Planning**  
Professors: Sergiu Dascalu and Vinh Le  
January 8th 2022 @ 2:00p.m.  
[Register](#)
- Project Management**  
Professors: Sergiu Dascalu and Vinh Le  
January 9th 2022 @ 2:00p.m.  
[Register](#)

Fig 8: Events registration system

If users register, the system will notify them that they successfully registered.



If users want to cancel after registering, they can click unregister and the system will show that they successfully unregistered

A screenshot of an event registration page. It features a logo with a stylized 'N', the title "Git and Github", and details about professors and date. Below the title is a large "Unregister" button. A success message dialog is overlaid on the page, showing "localhost:63342 says" and "Successfully Unregistered!" with an "OK" button.

Fig 9: Events registration system

### Achievements Page

This is the achievements page, which is an incentive and reward mechanism. When users complete some courses, they will receive badges. This is to make all users have a better experience and give users more motivation to learn.

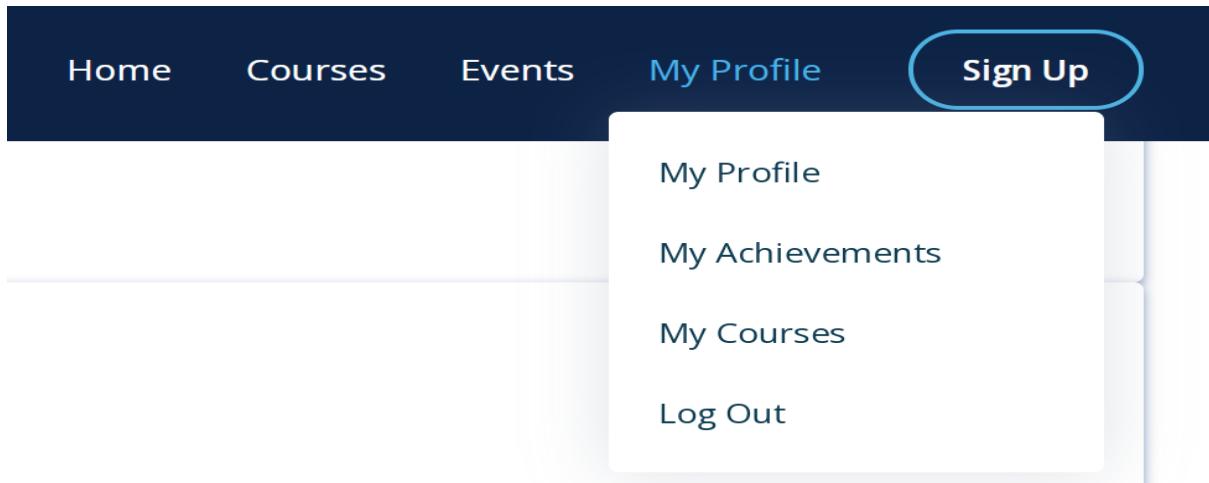


Fig 10: My profile

A screenshot of a mobile application titled "MASTER CS". The top navigation bar includes "Home", "Courses", "Events", "My Profile", and "Sign Up". Below the navigation is a green header bar with the title "ACHIEVEMENTS". A sub-header text reads: "Here you can see what badges you have earned. Unlock achievements for various tasks." Below this, there is a horizontal progress bar with a green segment labeled "0%".

ACHIEVEMENT	DESCRIPTION
	<b>Welcome to the Class!</b> Earned when you first register to create an account
	<b>E-Learning Newbie!</b> Earned when you complete your first lesson
	<b>Baby Step!</b> Earned when you complete your first topic

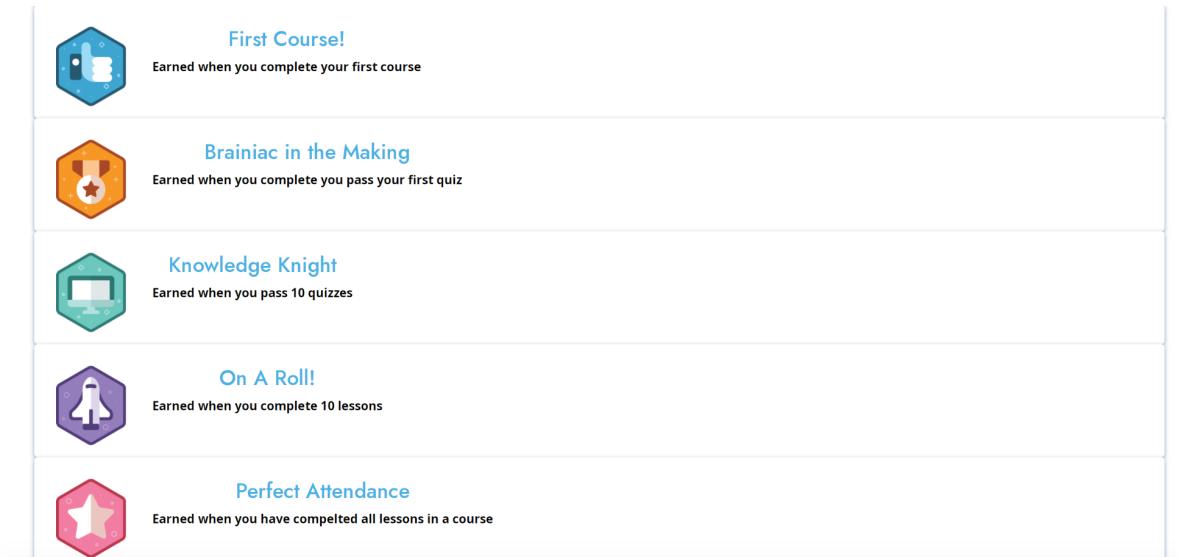


Fig 11: Achievements page

## Sign-Up & Login Page

This is the sign up page. Its main function is the login interface. If users are new, they can register according to the above information.

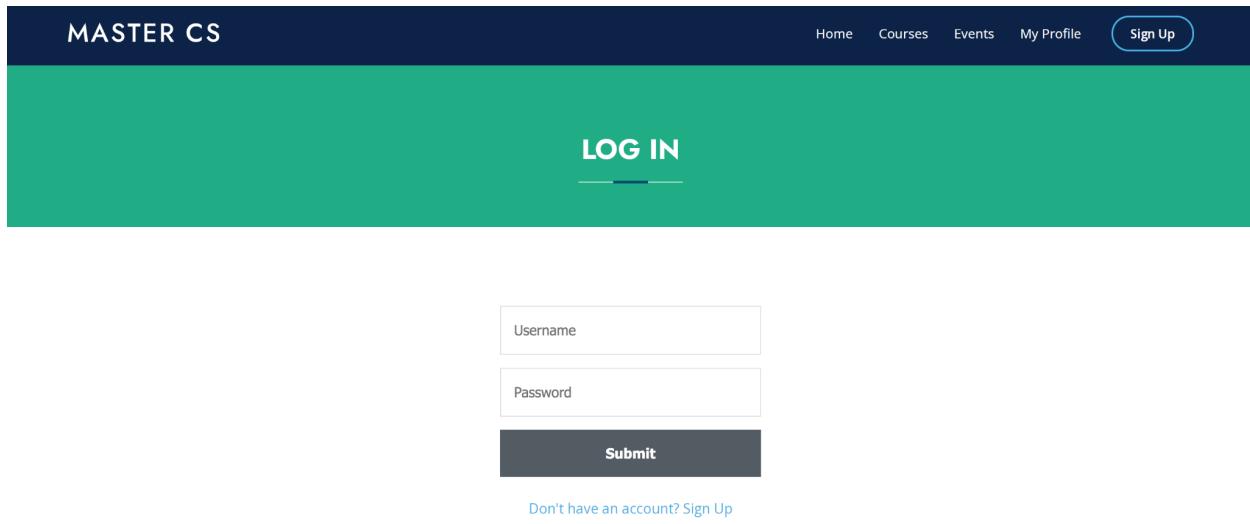
The sign up page features a dark blue header with the text "MASTER CS" and navigation links for Home, Courses, Events, My Profile, and a prominent "Sign Up" button. The main area has a teal background and displays the word "SIGN UP" in white capital letters. Below this, there are five input fields for user information: "First Name\*" and "Last Name\*" (each with a separate input field), "Your Email\*" (with one input field), "Your Phone\*" (with one input field), and "Password\*" (with one input field). At the bottom of the form, there are links for "Sign Up" and "Already have an account? Log In".

Fig 12: Sign up

If users already have an account, they can click log in.

[Sign Up](#)  
[Already have an account? Log In](#)

This is the log in page.



The screenshot shows a web application interface for 'MASTER CS'. At the top, there's a dark header bar with the text 'MASTER CS' on the left and navigation links 'Home', 'Courses', 'Events', 'My Profile', and a 'Sign Up' button on the right. Below the header is a large teal-colored section containing the word 'LOG IN' in white capital letters. Underneath this, there are two input fields: one for 'Username' and one for 'Password', both with placeholder text ('Username' and 'Password'). Below these fields is a dark grey 'Submit' button with white text. At the bottom of the teal section, there's a small link 'Don't have an account? Sign Up'.

Fig 13: Log in page

## Glossary of Terms

1. **Achievement:** The user completes an event, project, course, etc.
2. **Auditory Learner:** Auditory learners usually learn faster by listening and hearing.
3. **Backend:** Server side processes that are opaque to end users.
4. **HTML:** A popular front-end design framework that enables easy construction of webpages.
5. **Browser:** Devices used for accessing web content. Popular browsers include Google Chrome and Internet Explorer.
6. **CS related courses:** Learning modules that focus on the many branches of Computer Science such as networking, algorithms, and back/front end development.
7. **Database:** A collection of structured data that can be accessed from a computer system.
8. **Standards:** A set of guidelines that inform developers on ways to avoid harm to consumers and customers throughout the development process.

9. **Education platform:** A single portal that contains tools and resources for both learners and educators. Platforms typically contain all information to conduct education. Popular products include Google Classroom and Blackboard.
10. **Environment:** A programming environment is where the code is executed and data is sent to users.
11. **Free learning:** Educational material that is made available without cost to the end user.
12. **Frontend:** What the user sees and interacts with on a website. Technologies included are CSS, JavaScript, and HTML.
13. **Gamification:** The theory that most learners excel when they are having fun.
14. **Kinesthetic Learner:** This type learner requires the learner to manipulate or touch material to learn.
15. **Login:** The process of submitting credentials to an authentication server or service provider. Submitted credentials are compared with what is stored on a database. Depending on the result, access is granted to the client or denied.
16. **Subscriber:** An individual that has successfully completed the enrollment procedures and is able to access content defined by the service owners.
17. **UI/UX:** (User Interface/User Experience) UI is how the user interacts with the interface that the web service offers. UX is how easy the navigation is and the efficiency of page layout.
18. **Video Player:** A software program that is able to play digital video from a remote server. The digital video is encoded with a codec that reduces its size and is capable of being streamed over the internet with reduced bandwidth requirements.
19. **Visual Learner:** This kind of person learns things through pictures, video and story descriptions. They are able to learn quicker by seeing content.
20. **Administrator:** A user that has the ability to view statistics on other user's accounts. This type of account would be assigned to a teacher overseeing student progress.
21. **API (Application Programming Interface):** Allows programs to interact with each other without being directly controlled by the user.
22. **Persistence:** The ability for a site to track and save what a user has been doing so they can continue their activity if they get disconnected.
23. **User Statistics:** A set of numerical indicators which informs a student of their overall

progress and performance.

24. **Responsiveness:** A measure of how closely a site behaves to the desired behavior of the user. This applies to both mobile and browser based navigation.
25. **Components:** A set of objects created in Vue.js that enable easy reuse of design principles throughout the application.

## **Engineering Standards and Technologies**

1. **PHPMyAdmin:** This software allows for easy manipulation of the same database by multiple users. It provides an interface through which databases can be created and their structures modified more simply than through direct SQL commands. We plan to implement this technology to manage our database.
2. **Vue.js:** Vue is a front-end framework that enables the quick creation of new pages. Common page elements in Vue are created once and then shared across all pages on the site. This allows developers to reuse objects and also increases visual continuity for the user. We are using this framework to implement our front-end.
3. **Visual Studio:** This is an Integrated Development Environment (IDE) that is used for code development. It provides the functionality to manage, build, and run projects in a variety of languages. We are using this technology for our development process. Another IDE that is being utilized is WebStorm.
4. **XAMPP:** This software provides the ability to host an instance of a MySQL database on a web server. The server is not live, but it is accessible with the correct IP and login information. We plan to use this software to implement the online functionality of our database.
5. **Trello:** Trello is a project planning site that allows an entire group to track the progress of a project. Tasks are assigned into cards which can be placed into categories which denote various states of progress. This allows multiple users to work on the same project without any of their progress overlapping. We are using this application to track and manage our project progress.
6. **Discord:** Discord is a communication application that provides both Voice and Text communication functionality. This has become our main method of communication. Last semester we had a text chain and met through Zoom. But this semester we have found it is easier to center all of our communication in the same place.
7. **IEEE Standard 2089 - IEEE Standard for an Age Appropriate Digital Services**

**Framework Based on the 5Rights Principles for Children:** This standard provides a guideline for implementing online services that can be accessed by children under the age of 18. Children under 18 represent a vulnerable group on the internet and as such need to be protected in the design process. Our group plans to use this standard to ensure that our site design cannot be exploited to take advantage of this vulnerable group.

**8. IEEE Standard 1857.6 - Standard for Digital Media Content Description:** This standard describes the proper method for applying descriptor tags to digital media. The goal of this standard is to increase the ease of indexing and searching media within a site which in turn reduces the load on the network. We will use this standard to inform our use of descriptors on videos accessible through our own site.

### **Project impact and context considerations**

In this era of rapid economic and educational development, online education is more important than ever. It has decoupled education from being tied to a campus, enabling various educational resources to overcome the limitations of spatial distance, and become accessible to anyone in the world with an internet connection. There are many advantages of online education, open teaching scheduling, flexible learning time framing, and resource advantages regardless of physical location. Using the top educational talent with the best teaching methods available increases its effectiveness. Openness and learner choice has become synonymous with online education. It is precisely because of these that online education is not limited by age, location, time and less significantly, budget. Whether users want to continue their programming studies, enhance the probability of a promotion, ignite their passion for the programming hobbies, etc., the user can discover a suitable course and improve their skill set. Online education is not limited to local or distance education of computer and network technology. Since the recent Coronavirus pandemic, it has been proven in countries all over the world that a hands-off virtual form of education is possible and arguably more effective than a face-to-face form of education. Since online education is not limited to physical locations, users have the opportunity to contact others with the same enthusiasm from all over the world, allowing users to absorb different cultures and beliefs; Students and scholars from different countries creating a cultural bridge of understanding. There will be different opinions on economic, geographical, historical, political, and other issues. Social impacts are important to consider. Programming skills bring the

opportunity to help impact the direction of how society will change. With more individuals having the knowledge of programming, the more decentralized the knowledge will become. For example, users of the web platform will not need to drive to school, thereby reducing carbon emissions. Power bills for an empty classroom will be eliminated, helping reduce education related emissions. High-quality employees will also maximize the company's interests by shortening downtime commuting between the work environment and the learning platform. This will also allow more time to be used either studying or creating stronger social bonds. A higher skilled workforce provides economic prosperity for the entire world. Online education has become an indispensable educational channel in the 21st century.

## **List of References**

### **Domain Book:**

Beaird, Jason, Alex Walker, and James George. *The principles of beautiful web design*. Sitepoint, 2020.

Website development is not only to write good code, but also to understand what good design is from discovery to implementation. Because with good design and good code, we can show a website with full experience for all users. This book effectively puts forward many suggestions. For example, website developers know how to effectively use colors, formulate color schemes, create color palettes, and then design symmetry to create a pleasant layout. Here's how some typographical techniques can make a website look great.

### **Articles:**

Clarke, Irvine, et al. "Teaching the Visual Learner: The Use of Visual Summaries in Marketing Education." *Journal of Marketing Education*, vol. 28, no. 3, Dec. 2006, pp. 218–226, doi:10.1177/0273475306291466.

This article analyzes how visual learners are able to learn quickly. For example, learners prefer to enhance their learning through a range of programs such as pictures, diagrams, movies, and videos. Over 40% of college students are more effective at learning visually than the traditional lecture format. The article later points out that there is a disconnection between traditional teaching formats and how modern students learn in the classroom.

Ms.Rajwant kaur,Ms.Kawaljit Kaur,Preeti. "Research on HTML5 in Web Development". *International Journal of Advanced Science and Technology*, Vol. 29, no. 10s, May 2020, pp. 2412-8.

HTML5 is the latest iteration of the HTML standard introduced by the W3C consortium. It allows additional capabilities that were not available in previous versions and can support new dynamic features. Most notably, HTML5 supports dynamic content without the need for 3rd party plugins to properly display content. It also allows real-time collaboration by utilizing standard protocols between dissimilar services.

Garett, Renee et al. "A Literature Review: Website Design and User Engagement." *Online journal of communication and media technologies* vol. 6,3 (2016): 1-14.

This article on how to properly and purposefully design a good website. The author summarizes seven important factors: graphic representation, organization, navigation, content utility, purpose, simplicity, and readability. This will be of substantial help in visualizing and conceptualizing a website.

Ustimenko, Stanislav. "FOSS Project Spotlight: the Codelobster IDE--a Free PHP, HTML, CSS and JavaScript Editor." vol. 1, 2018, p. 2. *Linuxjournal*.

Code Lboster is a free php, html, css, and javascript editor. Notable features include the integrated autocomplete for functions, and an up to date help file query system that is provided by official sources. Editing HTML templates directly in the IDE is possible with integrated css inspector makes editing web files a breeze. Code lobster has support for over 17 user-interface languages and can run on any modern os including Windows 7-10, macOS, and multiple flavors of linux.

### **Websites:**

"Difference Between a Web Developer and User Interface Developer." *Codecondo*, 2020, <https://codecondo.com/difference-between-a-web-developer-and-user-interface-developer/>.

One of the key elements of website development is user interface development. The concept of web development is not only to write good code and complete the specified design, but also to interact with users and make them have a better experience. This part of design is basically included in front-end development, including HTML, CSS, JavaScript, UI and so on.

"EducationCat." *Wixsite*, <https://ashleyzhuqi.wixsite.com/website-1>.

We simulated and completed the initial snapshots of the user interface using Wix. This sample website shows a simple website concept. It includes a login interface, users can choose courses, an interactive forum interface (users can ask questions), and if they have any questions about the course, they can contact the staff at any time. In addition, the website will also provide events

regularly, so that users have more opportunities and experiences, and they will fully participate in more activities. We will further design in future projects.

“Kinesthetic Learning Style.” *Houghton*,  
<https://www.houghton.edu/current-students/center-for-student-success/academic-support-and-accessibility-services/study-advisement/general-study-information/kinesthetic-learning-style/>.

Online education website learning is also one of the topics for the exploration of users' learning methods. We take into account the different ways of students' learning and are committed to applying more strategies in our website development, such as which color needs to be marked for the courses mentioned in the article, so that students can focus, color selection and timeline, and charts are very important tools.

*Wales, Michael. "Web Dev Careers Decoded: Front-End vs Back-End vs Full Stack." Udacity official website 8 (2014).*  
<https://www.udacity.com/blog/2020/12/front-end-vs-back-end-vs-full-stack-web-developers.html>

Hope our group fully understands what front-end development and back-end development is. This website details the work security of developers and how they are responsible for coding, building, analyzing, and maintaining all these websites. For example, as web development trends and best practices change with the seasons, there is no shortage of developers. Web development work is very critical. Here we will design to the front end, back end, and complete stack. Developers need to understand their special and differences.

## **Contributions of Team Members**

Crystal Atoz: 2 hours

She worked on the abstract and project changes sections. Additionally, helped with editing and review.

Kayla Garin: 3.5 hours

Kayla worked on the Updated Specification section and revised the document.

Austin Schrage: 1.5 hours

Austin worked on the Glossary of Terms and Technology and Standards sections. He also helped make revisions on the document.

Yan Shore: 2 hours

Yan worked on Project impact and context considerations, List of References and helped with editing the document.

Zhuqi You: 2.5 hours

Zhuqi worked on Updated high-level and medium-level design. She also helped with editing the assignment.