جامعة أبوظبي
Abu Dhabi University

COLLEGE OF ENGINEERING

COMPUTER SCIENCE & INFORMATION TECHNOLOGY PROGRAM

SPRING 2021

ITE 410: WEB PROGRAMMING

COURSE PROJECT

Faculty: Dr. Murad Al-Rajab

- **Due Date:**
  - ✓ ***Phase One*** *Submission Due on* ***March 11, 2021****. Only submit through* ***Blackboard****.*
  - ✓ ***Phase Two*** *Submission Due on* ***May 29, 2021****. Only submit through* ***Blackboard*** *(detailed instructions are inside).*
  - ✓ ***Phase Three*** *Due on* ***May 30 - June 1, 2021****.* ***Project Presentations***

- **Percentage of Final Grade: 30%**

Sources of Information and Advice

The output of this project will be the work of a group of 2 students for the course project. Your output; the end-to-end website application (Website Development) must be the work of each group only. All sources of information must be properly referenced.

To use the work of others without identifying it is considered as a plagiarism. This is a form of academic dishonesty and will not be accepted, and may lead to dismissal from the Course or a grade of ***0*** *in the Project.*

PROJECT TOPICS AND GUIDELINE

✲ Learning outcomes covered on this assessment:

No.	Learning Outcome
1	Identify the basic concepts of Internet application development and web architectures.
2	Use Typescript scripting languages
3	Apply the MEAN stack frameworks to design web applications with Database.
4	Design and develop web Applications using Java technologies.
5	Use cloud servers to store and retrieve data for web-based applications.

✲ Topics

The centerpiece of this course is a semester-long project, the main objective of this course project is the design and implementation of a **major end-to-end MEAN full stack** website application in a group of **two students** using the key technologies we have practiced throughout the semester. The project must be an ***end-to-end full stack*** interactive website application that you have the freedom to select by yourselves. The idea of the website application should reflect a significant or interesting personal, cultural or social implication. This group project should apply and show what you have learned from the course. Your project should include the following compulsory technologies:

1. HTML5
2. CSS3
3. Bootstrap (optional)
4. JavaScript
5. Node.js and Express JS
6. MongoDB
7. AngularJs (Optional if time permits)

✲ Project ideas

You need to decide what end-to-end full stack website application you would like to build for the project. Here are some suggested guidelines for you to follow when deciding what you want to build:

Think of a Small and simple end-to-end full stack Website application.

- You need to incorporate all the above-mentioned technologies into one consistent website application.
  - Your work will not be graded on things like how interesting and impressive your project idea is, how original your idea is, or how representative your idea is, etc.… (this is not a web design course project!)
  - However, your project should be a complete and comprehensive website application, even if it's a very small or a trivial website application, or a simple one.
- Try to choose an idea that is simple and very focused for the subject of your project (social, educational, commerce, etc.).

- **No more than 4-6 different pages/screen in the entire app.**
- It is recommended that you have **only 1 to 6 screens (pages)** in your website application, 6 at the absolute maximum and it not preferred to go beyond.
- If your project involves more than 6 screens (pages), it is probably too big in scope for the project.
- You should include as one of the screens (pages); a personal web page in your project (called ***About Developers***) which includes a bibliography about each member of the team, and describes your own special talents and interests. I would recommend that you include a recent picture of each member. Please be Warned not to post anything on the page that you don't want to be publicly presented.
- You should include as one of the screens (pages), an HTML5 Ajax-enabled form that will be tested and shows error asynchronously when the user moves to another fields, or any other idea that can implement the same.
- Your project should be bug-free (*error-free*).
  - To ensure that you are using the technology right, then your code must be an error-free.
  - This is why you should implement a **small and simple** project idea.

✲ Style and Technology requirements

HTML5/ CSS3:

- Use different HTML5 tags, e.g. don't use the `<div>` regularly on every page.
- You can apply Bootstrap (optional) and modify the CSS accordingly
- You must write raw HTML5 and CSS3, as you had studied and you are not allowed to use any compiled HTML5.

JavaScript:

- **Must implement at least one Array in your project**
- **Must write object-oriented JavaScript.**
  - Try to avoid using global variables but it is fine to use global variables for constants, instantiating classes, or other realistic situations
  - But you should not declare anything as a global variable which could be better encapsulated in a class
- You must write raw JavaScript code, as you had studied in class.

Server side:

- Your server side must be implemented using the **Node JS and Express** framework libraries
- Must implement **AJAX** and **JSON** in your website and they can be in a structure of an HTML5 FORM or any other idea that you can utilize them.

Backend

- Don't save data to the filesystem: persistent data should be stored in a MongoDB, so your backend must contain MongoDB as the database.
- You need to create MongoDB collections as needed based on your project idea and what data will be stored into the database
- HTTP methods should be used in ways that are compatible with the method definition. For example:
  - Use GET for retrieving data. Do not write data in a GET handler.
  - Use POST for saving data. Do not use POST to display a page.

Notes:

- The goal of this project is to build a website from the ground up.
- You may not refactor or redesign a website that you've built before
- You are to use the technology tools that you had learned in classroom and mentioned earlier.

✲ Publishing your site

Hosting & Deploying:

- The whole project should be deployed and hosted on a cloud server. Some suggestions to use amazon EC2 or Heroku or Google CE (Compute Engine).

  ➲ http://aws.amazon.com/ec2/
  ➲ https://www.heroku.com/
  ➲ https://cloud.google.com/compute

  - https://github.com/SIB-Colombia/dataportal-explorer/wiki/How-to-install-node-and-mongodb-on-Amazon-EC2
  - https://jasonwatmore.com/post/2018/12/06/deploy-to-heroku-node-mongo-api-for-authentication-registration-and-user-management

- Your website must be successfully hosted and deployed on amazon EC2 or Heroku or Google CE for submission.
  - **Note:** If you use a MongoDB backend; you will need to use the related add-ons (if any required) such as the mLab Heroku add-on.
- **Make sure that you will have enough time before the final project submission deadline to practice on how to deploy and host your website on the suggested cloud servers. Doing so, can work out any issues before the final project submission deadline.**

Browsers:

- You must verify your site works on the latest version of any browser (could be Firefox or Chrome).

✵ Implementation Guidelines and Milestones

The following guidelines and millstones are suggested for your project:

Guideline 1: Implement the frontend UI with minimal-to-no CSS at the beginning

Implement one screen at a time, without CSS

- Write the code for the HTML5 and the JavaScript which are necessary to implement one screen at a time for your website. After you make sure that each screen works probably in an individual manner, you can implement the interactions between the other screens.
- No need to spend too much time writing your CSS3 at this stage, or no need to write the CSS3 at this stage. You can enhance your website application presentation using Bootstrap or the CSS3 later after you have your website application working end-to-end.

Guideline 2: Design your database

- In MongoDB backend, you need to decide how data will be stored:
  - What collection(s) (relations/ entities) are you going to have?
  - What fields (attributes) are you going to store in each object?
- Try to write a non-server NodeJS scripts for the purpose of testing the queries of your database. Thus, write a separate NodeJS script file which is used to query and update your database, before you touch and write your NodeJS server code. This will make sure that your MongoDB is built right and will work fine independently from the frontend or server-side code.
  - This is just a testing script, which will not be connected directly to your project code in any way.

Guideline 3: Implement the backend (server side)

- Write and run your NodeJS and Express server-side code.
- Write the related code for all of your GET and POST handler(s).
- Connect your frontend to your backend.
- Implement your routes, querying or updating your database.

Guideline 4: Finish the implementation of the CSS

- After you have your website running and working fine at the end-to-end level, you can finish the CSS3 website frontend styling.

Guideline 5: Host, Deploy and Test

- Host and deploy your website on the suggested cloud servers.
- Test your hosted and deployed website to make sure it is working as expected.
- Return back to the project requirements mentioned at the beginning of this document to make sure that your website applications had followed exactly what is required for the end-to-end full stack website.

Although you will be working progressively on the project, you will have ***three*** milestones that you have to keep in mind:

- ✓ ***Phase One Submission Due on March 11, 2021. Only submit through Blackboard.***
- ✓ ***Phase Two Submission Due on May 29, 2020. Only submit through Blackboard (detailed instructions are inside).***
- ✓ ***Phase Three Due on May 30 – June 1, 2021. Project Presentations***

✵ Project Management

The "managerial" role between team members must be rotated; the student in charge of the project manager role in this case is responsible to make sure that things inside the project are on track, following the correct sequence; the work is done as planned, etc. Students must peer in mind that the task requires little management skills, instead of all team members as a whole shares responsibility for the work to be done, the management role then is to be shared to assure a high-quality work. Shared work can be a serious problem, when a team member will be unable, not serious or unwilling to take responsibilities, for that the role of a project manager comes up. The project grade will assess not only the project work but also the individual contribution of each team member.

✵ Schedule of Project Deliverables

#	Deliverable	Due Date	What to Submit
1	Proposal submission	March 11, 2021	The idea of project full stack website application (description of the web site 1 – 2 paragraphs) including front end and back end usage, and names of the students in the group.
2	The programming and testing of all end-end scripts, modules, database and visual design.	May 29, 2020	All HTML, CSS, JavaScript, NodeJS, Express, Database files, and other related files such as the multimedia files (All the project files)
3	Projects presentations	May 30 – June 1, 2021	Live Demo

- *Note that the 1-2 paragraph description of your project idea doesn't have to be too long or descriptive, and it doesn't have to long and detailed; I just want to get a sense of what you want to make.*
- Note: **NO late submissions will be accepted for either deadline and will receive a grade of 0**.
- **All submission files will be checked using the plagiarism system checker**.
- You have to work **in Groups of 2 Students each** on this project, and only one submission per group.
- Use a zip utility to bundle your files together and submit them as one attachment through the project submission link on the Blackboard page of the course. Also, include in the zip file one text file call it ***readme.txt*** that includes the group names and the link to the cloud server where I can directly run your project.

✵ Grading

Student's grade for this project will be divided into two components. The first 20% of the grade is the team website end-to-end full stack implementation work including all components discussed earlier in this project document. The other 10% of the grade will be based on the project demonstration and presentation in front of class colleagues ensuring that the website is hosted on the cloud sever, tested and working fine. The presentation will be a 5-7 minutes presentation for each team including Qs & As. Individual team member work will be assessed as well. So, everyone needs to participate fully in their team's efforts. We strongly recommend you build something small and simple.

✵ Contest

A small contest for the final project submissions will take place at the end using the following category:

- **Best Website end-to-end application:** will be given to the most beautifully attractive project with exceptionally well-written code

If your final project is a contest winner, your lowest homework score (grade) will be dropped. You will be also awarded with a certificate of appreciation.

✵ Communication

Use the following notation whenever you email the Instructor regarding your Project:

`ITE410_FirstName_LastName_Project_Subject`

☺ **BEST WISHES** ☺

Copy & Paste of complete pages or articles or programs (codes) from the Internet or other resources are **strictly prohibited**. If any information or data is taken from any resource then a citation and referencing are mandatory.

Dr. Murad Al-Rajab