

CSC 443 – Web Programming Project 2 –Backend (Node JS, Express JS, Mongo DB, Rest API usage)

Date assigned: Thursday 12th of November 2021 - 03:00 PM

Date Due: Monday 6 December 2021 at 11:40 PM.

Please read the requirements slowly and very carefully.

Objective:

To test the students' knowledge of backend development using Node JS, Express JS, usage of Rest APIs and using Mongo DB database.

Groups

1. This is a **group project of 3 students**. So, I will share with you an Excel sheet where I write down group divisions (examples: Group 1, 2, 3...). Please finalize putting yourselves in groups as fast as possible and write your full names in one of the empty groups' cells in the Excel sheet. Groups should be from the same section of the course. I need this to be finished in a maximum of one or two days not more than that. If a student decides to do it alone, this also is fine. In the case of one student doing the project, he/she has to put one name in the excel sheet cell near one of the group numbers available.
2. All members of the group should submit to BB all the project files zipped including an MS Word Document stating the group number and detailing everything used from elsewhere. Everything that is not the creation of the group (images, videos, audio, code, algorithms etc...) should be cited adequately. If you do not include citations & adequate credit, you will get a zero grade. I need a separate MS Document called "**Contributions**" submitted by each group member to BB. It could be in the zip or as separate where in it you write the exact & detailed contributions of each member in the making of the project.

Tasks:

Please follow the requirements detailed in the following points. Don't forget that you have to submit an MS document containing all the sources and citations of anything that is not your own creation. The project is an opportunity to learn new things so please impress me! Make sure you do a good project so you can impress prospective employers and put it in your GitHub profile/portfolio.

Your project **MUST** adhere to the following requirements to get the full grade – each point is assigned has a certain weight of the total grade:

1. You will use one of the group's project 1 front-end (HTML and CSS) website (try to choose the best) and add to it backend logic that follows **exactly** the requirements detailed in this document. There are no grades assigned to any improvement using any HTML or CSS on the front-end. Thus, do not waste time on this. You might need to do

few front-end tweaks to accommodate more backend logic (Example: add forms or buttons etc..). This is absolutely fine.

2. Your backend logic i.e. the web server logic **MUST** be implemented in **Node JS and its modules**. No other web server technology will be accepted and if you will use anything other than **node js** you will get a zero grade on the project. Do not use ruby on rails or PHP or Python or anything else that is similar.
3. You **MUST** use **Express JS** for **routing** and **handling clients' requests** (get, post, delete, put etc...). You should have **different routes** that **do different tasks** according to your project 2 logic. You could have different kind of functionality: register users, add products to a shopping cart, create orders, read, write & delete from the database etc...
4. Your **database** has to be **MongoDB**. You can use [mongodb Node JS driver](#) or [mongoose](#) (to create your DB schema) or any other npm module that facilitates the creation and communication with your MongoDB database. You **MUST** show me that you create, read, update, delete (**CRUD**) in your database to take the grade of this requirement. Examples: create orders, create/delete users, delete products etc...
5. Please make sure that any npm modules are installed locally in your project directory (i.e. using `npm install <modulename> --save`)
6. You **MUST** use AJAX (preferably the Fetch API or other modern AJAX APIs) to make your client-side logic communicate with the server or with API endpoints & vice versa depending on the needs of your website topic. I have to see good amount of usage of AJAX (preferably using the Fetch API or other modern AJAX APIs). If your project 1 has forms, you can use the methods explained in the lectures and labs to send data of any kind to the server or you can use the WebAPI [FormData](#) if you like.
7. You **MUST** use **JavaScript DOM manipulation code** to control your website layout and UI on the client side. You have to show me that you have used DOM JS to control **numerous** things in your website (Examples: maybe showing/removing/moving divs, doing JS animations, adding HTML components, filling data in HTML components from the server or from consuming API data etc..). You **can** use jQuery or any *similar* JavaScript library (i.e., any JavaScript Library that controls the DOM like jQuery) in your project. Vanilla JS always impresses me more than libraries and frameworks. We want you to learn how to do DOM manipulation using the low-level foundational JavaScript/DOM features. jQuery makes DOM manipulation easier, so it is ok if you want to use it.
8. Your JavaScript code **MUST** be clean. You **MUST** follow **JS coding standards** and **best practices** (appropriate naming, appropriate use of classes, no global variables other than constructor functions, etc.). You **MUST** make sure your code is **readable**. Make sure also that you code is **performant** i.e., make sure that you follow performance tips explained in the lectures and tips from articles from your own research.
9. You **MUST** use a **templating JS library or engine** that works well with **Express JS** to create dynamic HTML pages from the server side. There are many candidates which you are free to use. We explain in the lectures/labs one templating library which you can use. Please consider this [link](#) and this [link](#) for few recommendations. Famous Template engines: Moustache ([Link1](#), [Link2](#)) or [Handlebars](#) or [EJS](#).

10. You **MUST** use **modern JavaScript syntax** in your project i.e., you **MUST** use ECMAScript 2015 and higher versions. You can use also pre ES6 style if you want but I need to see **good** amount of syntax usage from ES2015 and higher versions in your project (like using let, const, class definitions, arrow functions, for/of loops, modules, destructuring, rest/ spread operator etc...) in order for you to take the grade of this requirement.
11. There are many **Rest APIs** available online that pertains to any topic you can imagine. There are many free APIs for weather, forex, Spotify, Twitter and other social media, GitHub, medical, Google, COVID 19, museums etc... Showing that your application consume API data will greatly improve your grade. There are thousands of free APIs to use pertaining to any topic you can imagine.
12. **All group members MUST demo their project in front of the lecturer otherwise you will get a zero grade** as I have to check that **it is actually your own work not someone else work**. If you do not understand your code, this means you did not write it or this means you did not do the project or part of it. Don't let me think this way. You can learn to do things from the web. Nobody reinvent the wheel but do not use something in your project that you do not understand. There are no pedagogical benefits in doing this anyway. If you do this, it will affect negatively your grade. **You MUST cite adequately anything & everything that is not your own creation (images, videos, audio, code, algorithms, techniques etc...) in an MS word document submitted with the project.** Bear in mind that a true creator always knows well his/her own creation.

Important: Please provide a .txt file where you mention how to run your code, how your code is organized between front-end and back-end files and what npm modules you are using. This is in order for me to install them, or you can include this information in the MS Word Document.

Optional: You can package your node js application and/or deploy it on the web to [Heroku](#) or other similar paid or free services. The website must appear at the deadline or after it. If we have time, I will show you how to do that or you can learn it on your own. Remember deploying the site is optional.

General Rules

- Late submissions are not allowed.
- Cheating or copying other students' work will get you a zero grade for all involved students. I do not care if someone copied your work or who copied from whom. The project should be unique. It is statistically impossible that two groups would end up doing the same project or have the same code. Remember projects should not be about politics (foreign or especially local) nor about religions or matters of faith/belief. So, your backend should NOT generate or consume such content. I just want to avoid the projects becoming political propaganda. This is the only reason. I am a strong proponent of freedom of speech, expression & belief but please Web Programming is not the right place to express this.

- Any form of plagiarism or academic misconduct is prohibited. Copying code found elsewhere or changing it and claiming it to be yours will get you a zero grade immediately. Using anything which you did not create yourself, must be cited and/or credited adequately in the MS Document.
- Your website (front-end and back-end) should not appear elsewhere, nor should be a previous submission to a different or same course assignment nor a project done elsewhere in the industry or for a client or for any other reasons. If we discover this is the case, you will get a zero grade immediately.
- Please refer to the student code of conduct in the syllabus.

Please zip all your project files including the MS Word Document in one .zip file. Name the zip file with your full name (the way it appears on Banner), make sure it is working i.e. we can extract the files and run your server code and client side code and please submit it to Blackboard before the deadline. In case Blackboard server is down for whatever reason, kindly send the project zip file to my LAU email before the deadline.

NB: If the project size is large & cannot be uploaded to BB, please upload it to onedrive/Google Drive or similar services and share the link only with the lecturer in the MS Document (submitted to BB) before the deadline. **Don't leave submission to the last hour or two. We live in an unstable country with unstable internet and electricity so ideally you should not finish the last minute.** You have a lot of time to finish this project.

If you have any questions, do not hesitate to contact me.

Good luck!