

COMPX322-21X – Advanced Web Development

Assignment 3: Node.js and React (20%)

Due date: Monday 7th June 2021 at 8:00 pm

Submission instruction: Your web application should be in a folder called "comp322-assignment3-318XXXX". **REMOVE THE `node_modules` FOLDER BEFORE SUBMITTING.** Compress your application folder (without `node_modules`) in a ZIP file called "318XXXX- comp322-assignment3.zip" where 318XXXX is your ZUCC student ID and upload it to Blackboard.

For this coursework you are required to build a project management application which allows users to manage their projects. Your implementation will require use of the following:

- HTML
- Cascading Style Sheets (CSS)
- JavaScript
- ReactJS to build the front-end user interface
- Bootstrap or Foundation for designing [Optional]

Application description

- This is a simple front-end application for managing projects.
- Users can see all the available projects, add more projects, delete projects, search and sort projects.
- The application lists all the projects from a file as shown in the screenshot [This is just an example, you should write your own CSS to display the projects nicely – you can use Bootstrap or Foundation but that is not required]:

Project manager

Add Project

Sort by

Project Name

 Sort order

Ascending

<div><div>x</div><div>Project 2 CRM System</div></div>	Start Date: 01 Feb 20 - End Date: 30 Sep 20 CRM (Customer Relationship Management) systems help businesses organize relationships with their customers.
<div><div>x</div><div>Project 5 E-Commerce Website for Visually Impaired</div></div>	Start Date: 01 Jan 21 - End Date: 05 May 21 An ecommerce website is developed to assist blind people that automatically recognizes clothing patterns and colours.
<div><div>x</div><div>Project 1 Library Management System</div></div>	Start Date: 01 Jan 21 - End Date: 31 Jan 21 Library management is a project that manages and stores books information electronically according to students needs
<div><div>x</div><div>Project 3 Online Charity Management System</div></div>	Start Date: 01 Feb 21 - End Date: 28 Feb 22 This charity management system will help NGOs to find donors easily.
<div><div>x</div><div>Project 4 Visitor Indicator System</div></div>	Start Date: 09 Jul 19 - End Date: 31 Jan 20 This doorbell cum visitor indicator circuit can give identification of the visitor to your home in your absence.

- User can also delete the project by clicking on the “X” button as shown in the above screenshot
- When a user clicks on the “create a new project” button, a form should appear for adding new projects as shown in the following screenshot [Design may vary depending upon how you want to design your application using your own creativity]

Project Name

Project ID

Description

Start Date

End Date

Submit

- You should make a single page application – using conditional rendering to decide what to display – form or button (see <https://reactjs.org/docs/conditional-rendering.html>)
- Form consists of several input fields for Project Name, Project ID, Project Description, Start Date, End Date and Submit Button as shown in the screenshot above
- On clicking the submit button, a new project should appear in the list of projects. Form validation is not required, i.e. you can assume that the user will input valid data.
- The user can also search (by project name) and sort the projects. Projects can be sorted in ascending or descending order by project name and by start date.

Search Sort by Project Name Sort order Ascending

Search Sort by Project Name Start Date Sort order Ascending

Search Sort by Project Name Sort order Ascending Descending

NOTE: There is no requirement for back-end functionality or a permanent storage. You are not required to save changes to the list of projects to "data.json". Thus, the changes to the project list (add/remove) will not be permanent, and if the app gets reloaded the original list will be fetched again from "data.json".

Implementation

You should create a React App with npm, naming your app "comp322-assignment3-318XXXX" where 318XXXX is your student ID. As we covered, the command to set up the project is therefore `npx create-react-app comp322-assignment3-318XXXX`

You might need to install extra dependencies using "npm install ... --save".

Organise your folders using a **components** folder and a **css** folder inside the **src** folder. In the src folder there should be only the "index.js" file, all the component JS files should be in the components folder, while the css files for their appearance should be in the css folder – each component should have its own css file.

You have been provided with the "data.json" file which contains the project data. Download this file from Blackboard and place it in the **public** folder of your application. Your public folder should just contain the following files: "data.json", "favicon.ico", "index.html" and "manifest.json". You should not need to edit these files. You can delete the others in this folder and you should not need to add other files here.

You are allowed to use third-party npm modules to improve the appearance or functionalities of your application but the minimum functional requirements should be achieved without them.

What to submit and how

All of your material for this assignment must be submitted electronically on Blackboard.

In your computer, all parts of your application should be inside a directory called "comp322-assignment3-318XXXX" where 318XXXX is your ZUCC student ID.

When you have finished your work, REMOVE THE node_modules FOLDER. Compress the directory into a ZIP file called "comp322-assignment3-318XXXX.zip. Upload the compressed file to the Assignment 3 submission link on Blackboard. No other mechanism for submission will be accepted.

DO NOT USE `npm run build` OR `npm run eject` (don't use this, it is very dangerous – you will lose your original files!). These commands are for creating optimised, compressed versions of your app, which is NOT what you should submit. You should submit the development source files (like in the examples shown in class).

Make sure all work submitted is your own, you should be familiar with the University regulations on plagiarism. You are free to discuss with your classmates, but you should not share any code with anyone. This is an **individual** assignment so you should prepare all of the code on your own. A selection of students will be asked to explain their solution or parts of their code as part of the assessment process.

You are encouraged to make a Git repository of your work and upload it to the Gitlab server of the school. This is NOT a requirement for the assignment and you will get full grades even without using Git. However, if you choose to use Git and make regular commits, not only you will have a good system for managing your work on the assignment, but also the commits will show your process of development and likely avoid the need to have you explain your code.

Grading

Weighting	Allocated to
30%	Application meets minimum functional requirements, with reasonable design (CSS) and components organisation
10%	React: Using fetch in a lifecycle method to list projects
10%	React: Delete project functionality
20%	React: Form to add projects and its processing
20%	React: Searching and sorting projects
10%	Folder structure, comments and coding style

NOTE: marks can be deducted in case of missing functionalities or relevant issues in the application.