Course Title: Fundamentals in Frontend Development

END OF MODULE PROJECT (CAPSTONE PROJECT)

Project Title / Description	:	Implement a React SPA
Project Objective(s)	:	Create a React single-page application (SPA) that has at least a listing page and a details page showing data from API. *Browse public API at https://github.com/public-apis/public-apis and choose one that seems interesting to you. To minimize complexity, avoid choosing those that require OAuth. If you're comfortable you can choose those with apiKey (which usually requires you to register an account with them). If you are not sure which one to choose, here are some personal favourites: https://api.artic.edu/docs/ (if you are feeling artsy), https://breakingbadapi.com/documentation (for Breaking Bad fans).* The application should allow users to bookmark items (like a favourite button) and show the bookmarked items in the home page. The bookmark data can be saved in localStorage. While creating the application, you should develop the style guide for your application using React Styleguidist. You're free to use any styling technology (plain CSS, TailwindCSS, Bootstrap) etc, but your application must be styled. You're **NOT** allowed to use third-party component libraries (like Material UI or React Bootstrap) but you're free to refer to them for inspiration.
Project Duration	:	**121 hours**

Mode of Delivery	:	Individual Consultation + Individual Presentation

MARKING RUBRICS FOR THE END OF MODULE PROJECT (CAPSTONE PROJECT)

	Score		
Category	**0-1**	**2-3**	**4-5**
Problem and Solution (x2)	▪ **0-1** - No to very minimal attempt made ▪ **2-3** – The app is unusable.	▪ **2-4** Students implement only 1-2 of the requirements.	▪ **7-8** Students implement 80% of the requirements.

		▪ **5-6 -** Students implement half of the requirements.	▪ **9-10** Students implement the complete set of requirements.
Documentation	▪ **0** - No code / attempt documentation written. ▪ **1** –The code is undocumented.	▪ **2 -** There are some comments explaining the code, however, provides no extra context to the code readers. ▪ **3 -** There are some comments explaining the code that the readers can't understand without.	▪ **4 -** The code is self-documented and easy to understand **or** there's external README file that's really helpful ▪ **5** – The code is written well, easy to understand. The app is documented in a README that someone can get up and running with
Presentation and Communication	▪ **0** - Student do not present clearly on the work and/or unable to answer questions related to the work. ▪ **1** – Student could briefly present the	▪ **2 -** Student present the work and are able to answer some questions related to the work. ▪ **3 -** Student present the work and are able to	▪ **4 -** Student communicates well during presentation and able to answer most of the questions related to the work. ▪ **5** – Student communicates well

	work and/or unable to answer questions related to the work.	answer most questions related to the work.	during presentation and able to answer all the questions related to the work.