Project Assessment Rubric

	10-9	8-7	6-5	4-0
Functionality	Applications demonstrate comprehensive & robust evidence on the following: Routes protected with a token value. Frontend application requests data from backend application using Axios. Create, update & delete API data via modal. Incorrectly formatted form fields are handled using validation error messages. View API data in a table using an id and a variety of query parameters. API data paginated across several pages. UI styled with Reactstrap. Applications deployed to Heroku. End-to-end tests thoroughly demonstrate coverage of creating, updating, deleting & viewing API data.	Applications demonstrate clear & detailed evidence on the following: Routes protected with a token value. Frontend application requests data from backend application using Axios. Create, update & delete API data via modal. Incorrectly formatted form fields are handled using validation error messages. View API data in a table using an id and a variety of query parameters. API data paginated across several pages. UI styled with Reactstrap. Applications deployed to Heroku. End-to-end tests mostly demonstrate coverage of creating, updating, deleting & viewing API data.	Applications demonstrate evidence on the following: Routes protected with a token value. Frontend application requests data from backend application using Axios. Create, update & delete API data via a modal. Incorrectly formatted form fields are handled using validation error messages. View API data in a table using an id and a variety of query parameters. API data paginated across several pages. UI styled with Reactstrap. Applications deployed to Heroku. End-to-end tests demonstrate some coverage of creating, updating, deleting & viewing API data.	Applications does not, or does not fully demonstrate evidence on the following: Routes protected with a token value. Frontend application requests data from backend application using Axios. Create, update & delete API data via modal. Incorrectly formatted form fields are handled using validation error messages. View API data in a table using an id and a variety of query parameters. API data paginated across several pages. UI styled with Reactstrap. Applications deployed to Heroku. End-to-end tests does not or does not fully demonstrate coverage of creating, updating, deleting & viewing API data.

		Applications thoroughly demonstrates	Applications clearly demonstrates code	Applications demonstrates code	Applications does not or does not fully
Elegance		code elegance on the following:	elegance on the following:	elegance on the following:	demonstrate code elegance on the
		 Appropriate use of control 	Appropriate use of control	Appropriate use of control	following:
		flow, data structures and in-	flow, data structures and in-	flow, data structures and in-	Appropriate use of control
	a)	built functions.	built functions.	built functions.	flow, data structures and in-
	ຼັ	 Sufficient code modularity. 	 Sufficient code modularity. 	 Sufficient code modularity. 	built functions.
	ă	 Components written as 	 Components written as 	 Components written as 	 Sufficient code modularity.
	<u>e</u>	functional, not class.	functional, not class.	functional, not class.	 Components written as
		 Adheres to client-server 	 Adheres to client-server 	Adheres to client-server	functional, not class.
Code	ğ	architecture.	architecture.	architecture.	 Adheres to client-server
	ပ	 Header & in-line comments 	 Header & in-line comments 	 Header & in-line comments 	architecture.
		explain complex logic.	explain complex logic.	explain complex logic.	Header & in-line comments
		 Formatted code using Prettier 	Formatted code using Prettier	Formatted code using Prettier	explain complex logic.
		& npm script.	& npm script.	& npm script.	Formatted code using Prettier
		 No dead or unused code. 	No dead or unused code.	No dead or unused code.	& npm script.
		DEADAGE (The control of the control	DEADAG (Il. and alama de la constitución de	DEADME (il	No dead or unused code. PEARME Standard and the second Standard Stand
		README file contains thoroughly evidence of:	README file contains clear evidence of:	README file contains evidence of:	README file does not or does not fully contain evidence of:
			URL to applications on Heroku.	URL to applications on Heroku.	
	e e	 URL to applications on Heroku. How to setup the 	How to setup the	How to setup the	URL to applications on Heroku. Llow to setup the
	Usage	 How to setup the environment for 	environment for	environment for	How to setup the environment for
	Š		development, run the tests &	development, run the tests &	
	<u>Git</u>	development, run the tests &	deploy the applications.	deploy the applications.	development, run the tests &
	ø	deploy the applications.	Git branches mostly named with	Same git branches named with	deploy the applications.
		Git branches thoroughly named with	convention & contain the correct code	Some git branches named with convention & contain the correct code	Git branches are not or are not fully
	:≌	convention & contain the correct code			named with convention & do not or
	Ħ	relating to the functional	relating to the functional	relating to the functional	do not fully contain the correct code
	Documentation	requirement.	requirement.	requirement.	relating to the functional
	<u> </u>	requirement.	Git commit messages clearly formatted	Git commit messages formatted &	requirement.
	0	Git commit messages comprehensively	& reflect the functionality changes in	reflect the functionality changes in	requirement.
	Δ	formatted & reflect the functionality	substantial detail.	detail.	Git commit messages are not or are not
		changes in succinct detail.	January McCom.		fully formatted & do not or do not
		onanges in succinct actum.			reflect the functionality changes.
					remede the functionality changes.

Project Marking Cover Sheet

Name:
Date:
Learner ID:
Assessor's Name:
Assessor's Signature:

Criteria	Out Of	Weighting	Final Result
Functionality	10	50	
Code Elegance	10	40	
Documentation & Git Usage	10	10	
	/100		

This assessment is worth 80% of the final mark for the Introductory Application Development Concepts course.

Feedback: