

## ID607001: Introductory Application Development Concepts

### Project 1: Node.js REST API Assessment Rubric

	10-9	8-7	6-5	4-0
<b>Functionality</b>	<p>The REST API contains thorough evidence of the following requirements:</p> <ul style="list-style-type: none"> <li>Developed using Node.js &amp; can run locally without modification.</li> <li>Appropriate number of collections &amp; fields of data with different types.</li> <li>Relationships between collections.</li> <li>Separate controller &amp; route files that contain CRUD operations for each collection.</li> <li>Custom validation for each field.</li> <li>Return a response containing success &amp; data when performing CRUD operations.</li> <li>HTTP headers are secured.</li> <li>Cross-origin HTTP requests are enabled.</li> <li>Deployed to Heroku.</li> <li>Data is stored appropriately.</li> </ul> <p>In addition, the REST API contains comprehensive evidence of the independent research requirements.</p>	<p>The REST API contains clear evidence of the following requirements:</p> <ul style="list-style-type: none"> <li>Developed using Node.js &amp; can run locally without modification.</li> <li>Appropriate number of collections &amp; fields of data with different types.</li> <li>Relationships between collections.</li> <li>Separate controller &amp; route files that contain CRUD operations for each collection.</li> <li>Custom validation for each field.</li> <li>Return a response containing success &amp; data when performing CRUD operations.</li> <li>HTTP headers are secured.</li> <li>Cross-origin HTTP requests are enabled.</li> <li>Deployed to Heroku.</li> <li>Data is stored appropriately.</li> </ul> <p>In addition, the REST API contains adequate evidence of the independent research requirements.</p>	<p>The REST API contains evidence of the following requirements:</p> <ul style="list-style-type: none"> <li>Developed using Node.js &amp; can run locally without modification.</li> <li>Appropriate number of collections &amp; fields of data with different types.</li> <li>Relationships between collections.</li> <li>Separate controller &amp; route files that contain CRUD operations for each collection.</li> <li>Custom validation for each field.</li> <li>Return a response containing success &amp; data when performing CRUD operations.</li> <li>HTTP headers are secured.</li> <li>Cross-origin HTTP requests are enabled.</li> <li>Deployed to Heroku.</li> <li>Data is stored appropriately.</li> </ul>	<p>REST API does not, or does not fully contain evidence of the following requirements:</p> <ul style="list-style-type: none"> <li>Developed using Node.js &amp; can run locally without modification.</li> <li>Appropriate number of collections &amp; fields of data with different types.</li> <li>Relationships between collections.</li> <li>Separate controller &amp; route files that contain CRUD operations for each collection.</li> <li>Custom validation for each field.</li> <li>Return a response containing success &amp; data when performing CRUD operations.</li> <li>HTTP headers are secured.</li> <li>Cross-origin HTTP requests are enabled.</li> <li>Deployed to Heroku.</li> <li>Data is stored appropriately.</li> </ul>

Code Elegance	<p>REST API thoroughly demonstrates code elegance on the following:</p> <ul style="list-style-type: none"> <li>• Use of intermediate variables, idiomatic control flow, data structures, in-built functions &amp; sufficient modularity.</li> <li>• Functions &amp; variables are named appropriately.</li> <li>• Efficient algorithmic approach.</li> <li>• Resource groups are named with a plural.</li> <li>• File header &amp; in-line comments using JSDoc.</li> <li>• npm script which seeds the collections.</li> <li>• No dead or unused code.</li> <li>• Database configured for development &amp; production environments.</li> <li>• Environment variables are stored appropriately.</li> </ul> <p>In addition, the REST API contains comprehensive evidence of the independent research requirements.</p>	<p>REST API clearly demonstrates code elegance on the following:</p> <ul style="list-style-type: none"> <li>• Use of intermediate variables, idiomatic control flow, data structures, in-built functions &amp; sufficient modularity.</li> <li>• Functions &amp; variables are named appropriately.</li> <li>• Efficient algorithmic approach.</li> <li>• Resource groups are named with a plural.</li> <li>• File header &amp; in-line comments using JSDoc.</li> <li>• npm script which seeds the collections.</li> <li>• No dead or unused code.</li> <li>• Database configured for development &amp; production environments.</li> <li>• Environment variables are stored appropriately.</li> </ul> <p>In addition, the REST API contains adequate evidence of the independent research requirements.</p>	<p>REST API demonstrates code elegance on the following:</p> <ul style="list-style-type: none"> <li>• Use of intermediate variables, idiomatic control flow, data structures, in-built functions &amp; sufficient modularity.</li> <li>• Functions &amp; variables are named appropriately.</li> <li>• Efficient algorithmic approach.</li> <li>• Resource groups are named with a plural.</li> <li>• File header &amp; in-line comments using JSDoc.</li> <li>• npm script which seeds the collections.</li> <li>• No dead or unused code.</li> <li>• Database configured for development &amp; production environments.</li> <li>• Environment variables are stored appropriately.</li> </ul>	<p>REST API does not or does not fully demonstrate code elegance on the following:</p> <ul style="list-style-type: none"> <li>• Use of intermediate variables, idiomatic control flow, data structures, in-built functions &amp; sufficient modularity.</li> <li>• Functions &amp; variables are named appropriately.</li> <li>• Efficient algorithmic approach.</li> <li>• Resource groups are named with a plural.</li> <li>• File header &amp; in-line comments using JSDoc.</li> <li>• npm script which seeds the collections.</li> <li>• No dead or unused code.</li> <li>• Database configured for development &amp; production environments.</li> <li>• Environment variables are stored appropriately.</li> </ul>
---------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Documentation & Git Usage	REST API documented in succinct detail using Postman.	REST API documented in substantial detail using Postman.	REST API documented in detail using Postman.	REST API not or not fully documented in detail using Postman.
	<p>README file contains thorough evidence of:</p> <ul style="list-style-type: none"> <li>• URL to the documented REST API on Postman &amp; Heroku.</li> <li>• How do you setup the development environment?</li> <li>• How do you deploy the REST API to Heroku?</li> <li>• How do you seed the collections?</li> <li>• How do you format the code files using Prettier?</li> </ul> <p>Comprehensive use of Markdown syntax, i.e., headings, bold text, code blocks, etc.</p> <p>Thorough spelling &amp; grammar correctness.</p> <p>Git commit messages are comprehensively formatted &amp; reflect the functionality changes in succinct detail.</p>	<p>README file contains clear evidence of:</p> <ul style="list-style-type: none"> <li>• URL to the documented REST API on Postman &amp; Heroku.</li> <li>• How do you setup the development environment?</li> <li>• How do you deploy the REST API to Heroku?</li> <li>• How do you seed the collections?</li> <li>• How do you format the code files using Prettier?</li> </ul> <p>Substantial use of Markdown syntax, i.e., headings, bold text, code blocks, etc.</p> <p>Clear spelling &amp; grammar correctness.</p> <p>Git commit messages are clearly formatted &amp; reflect the functionality changes in substantial detail.</p>	<p>README file contains evidence of:</p> <ul style="list-style-type: none"> <li>• URL to the documented REST API on Postman &amp; Heroku.</li> <li>• How do you setup the development environment?</li> <li>• How do you deploy the REST API to Heroku?</li> <li>• How do you seed the collections?</li> <li>• How do you format the code files using Prettier?</li> </ul> <p>Use of Markdown syntax, i.e., headings, bold text, code blocks, etc.</p> <p>Spelling &amp; grammar correctness.</p> <p>Git commit messages are formatted &amp; reflect the functionality changes in detail.</p>	<p>README file does not or does not fully contain evidence of:</p> <ul style="list-style-type: none"> <li>• URL to the documented REST API on Postman &amp; Heroku.</li> <li>• How do you setup the development environment?</li> <li>• How do you deploy the REST API to Heroku?</li> <li>• How do you seed the collections?</li> <li>• How do you format the code files using Prettier?</li> </ul> <p>Does not or does not fully demonstrate use of Markdown syntax, i.e., headings, bold text, code blocks, etc.</p> <p>Does not or does fully demonstrate spelling &amp; grammar correctness.</p> <p>Git commit messages are not or are not fully formatted &amp; do not or do not reflect the functionality changes.</p>

## ID607001: Introductory Application Development Concepts

### Project 1: Node.js REST API Marking Cover Sheet

Name:

Date:

Learner ID:

Assessor's Name:

Assessor's Signature:

Criteria	Out Of	Weighting	Final Result
Functionality	10	40	
Code Elegance	10	45	
Documentation & Git Usage	10	15	
Final Result			/100
This assessment is worth 30% of the final mark for the Introductory Application Development Concepts course.			

**Feedback:**

**Functionality:**

**Code Elegance:**

**Documentation & Git Usage:**