CS 465 Module Four Assignment Guidelines and Rubric

Overview

You are working towards building a full stack application, specifically called the MEAN stack. The acronym contains letters representing some of the critical software you will utilize to develop the app (MongoDB, Express, Angular, and Node.js). You have reached a critical stage in the full stack project. You will create a database and then connect the database to the server which, in turn, will deliver the data to the browser for the customer who uses the Travlr Getaways website. In this module, you will create MongoDB databases. You will provide structure, models, and schemas to the database to establish the groundwork for the API. Finally, you will add mock data to test pulling live data from the database.

Prompt

Follow the **NoSQL Database, Models, and Schema** section of the CS 465 Full Stack Guide and address the following:

- **Database Access Module:** Create a JavaScript module to access the NoSQL database using the Mongoose package library.
- **API Integration:** Create the logic necessary to define the trip schema with validation and retrieve the trip data in JSON format, using the Mongoose node package library.
- **Populate Database:** Use the Seedgoose node package library to populate the Trips collections with sample data suitable for Travlr Getaways' software requirements.
- **Testing:** Test the database by using a database tool (Robo 3T) to verify that the data has been loaded correctly.

What to Submit

Submit your updated zipped file folder, travlr.zip. The trips.json file should be included as part of the zipped file folder. This will demonstrate that when you open the database, it shows the live data. The JSON file will live within the travlr.zip folder.

Please note: You will "push" your local Git repository module4 branch to your GitHub repository after each module iteration. This procedure will become second nature to you as you move through the course modules. In the end, you will have a full stack web application with branches that represent key stages of building a full stack application leading up to full project completion.

Module Four Assignment Rubric

Criteria	Proficient (100%)	Needs Improvement (70%)	Not Evident (0%)	Value
Database Access	Accesses a NoSQL database by creating a JavaScript module with proper error-handling logic	Shows progress toward proficiency, but with errors or omissions	Does not attempt criterion	35
API Integration	Defines the schema and successfully retrieves data in JSON format to allow for API integration	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include using Mongoose functions to access MongoDB	Does not attempt criterion	35
Populate Database	Populates the database model using a node package library	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include inserting seed data into MongoDB	Does not attempt criterion	20
Testing	Tests the database to ensure data exists in the database in the proper format and is returned as JSON	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include ability for database to retrieve data or database returning data in improper format	Does not attempt criterion	10
			Total:	100%