Web Science

Quiz 1: February 27, 2024

Enter your answers directly into this document (with the exception of #2 and #3).

All answers should be In Your Own Words, using complete sentences with proper spelling and grammar.

Save this document as either a Word docx or a PDF. For all questions other than #2 and #3, you will not receive any credit for answers not placed in this document.

When finished with the quiz, put everything you wrote (this document, all code, etc.) in your personal GitHub repo in a folder named **quiz1**. Local development only for this quiz, but everything you want graded must be on GitHub in your **quiz1** folder.

1. **Short answers** (25 points): (Answer in complete sentences unless otherwise specified, explain your answers)
   1. (8) Fill out this table as to what should happen for each verb at each endpoint **according to the letter of the law.** (No need for complete sentences.)

| **Resource** | **POST** | **GET** | **PUT** | **DELETE** |
| --- | --- | --- | --- | --- |
| **/news** | Create new news (bulk) | get/receive all news (bulk) | update news (bulk) | Delete/remove all news (bulk) |
| **/news/123** | update data specific news123 article | get news123 article | if exists update news 123  if not error | Delete/remove news #123 |

* 1. (7) What is a package.json file? What is it used for? How is it created?

Package.json file contains project properties (name, version, description, author) as well as minimum version of dependencies (tools, frameworks, utility libraries). Also the package.json file is used to help version management by specifying the version of Node.js and what it's compatible with to ensure that the project runs consistently across different environments. It serves as a manifest for the project, providing information about the project, its dependencies and various configuration settings. The ‘package.json’ file is essential for Node.js developers to define project details and automate various tasks related to project management. In the terminal you use the command ‘npm init’ which will then generate the package.json file you fill out the descriptions such as name, version, description, main, scripts.

* 1. (10) Describe **in exacting detail** the sequence(s) of transaction(s) for a frontend to request data from some external entity via Node.

First a frontend request is set which is often implemented using HTML, JavaScript and CSS, which initiates a request for data. Next is a HTTP request from frontend, JavaScript code in the frontend makes an HTTP request to a backend server implemented in Node.js Node.js Server receives request, the node.js server receives the request from the frontend. This involves configuring a route in the Node.js applications, which handles specific endpoints or URL paths that the frontend requests. Backend processing, node.js server executes the corresponding server-side logic. The logic involves interacting with a database (making API calls). External Entity Interaction node.js server interacts with that external entity to retrieve the necessary information. Data processing and transformation, once the data is retrieved the server may process or transform it as needed before sending it back to the frontend. HTTP response from node.js contains the processed data and status code. Frontend receives response, which involves JavaScript code processes the HTTP response received from the Node.js server. Data handling in frontend code extracts and processes the data from the HTTP response. User interaction frontend may enable user interactions based on data received , allow users to submit forms or click buttons.

1. **Coding question**: (60 points) Here is a free API that does not require any API keys: <https://www.boredapi.com/> – the documentation for which can be found here: <https://www.boredapi.com/documentation>  
     
   Create a new GET endpoint in Node/Express that accepts the type of activity (and optionally additional information in a query string–check the documentation!) from your frontend and returns the API’s information about that activity back to the frontend. If the user fails to input a valid activity, return the /api/activity endpoint (which returns a random activity).  
     
   Creativity matters; try your best to integrate this new information into your frontend. Try as hard as you can to make it feel like it is a meaningful, conscious, intentioned feature of your app. How you do that is up to you. Don’t make it look like some random afterthought. Go beyond the minimum (but remember that creativity doesn’t have to be visual). If you need to, write a short README.md file that tells me what I should consider for creativity. (creativity: 20 points of the 60 available for this question)  
     
   ***You may use any and all open source libraries you want for this coding question, so long as you cite them in a README.md file.***

20 points for frontend (could be React, doesn’t need to be React)

20 points for backend (extending your API to include this new info, serving it to your frontend)

20 points for creativity

1. (15) Ensure the package.json file for Q2 has no errors when I run npm install.
2. **Extra credit (+5)**: Before moving into its current building, which building on campus was the home of the Rensselaer Student Union?

The Student Club House, with a terrace on the back of the building overlooked the '86 Field. The Club House was razed in 1932 when a [new students' club](https://www.rpi.edu/dept/library/html/Archives/buildings/student_clubhouse.html) was completed.