CHECKLIST: Dynamic Web Server (50 pts)

To earn 38/50 points (grade: C)

DONE Package.json Completed by Cody

- Fill out the author and contributors sections in package.json (author should be whoever's GitHub account is used to host the code, contributors should be all group members)
- Fill out the URL of the repository
- Ensure all used modules downloaded via NPM are in the dependencies object
- Ensure the "node_modules" folder is **not** included in the GitHub repository

DONE Dynamic Web Pages: Completed by Martin and Charlie

- Create at least 3 dynamic routes for viewing the data from different points of view (for example, if choosing a data set on energy consumption, the routes may be to view the data by state, by year, or by energy type)
  - Write HTML template files that data can be inserted into
- Include appropriate representation of the data
  - Should include some text-based data (e.g. headers, paragraphs, tables, etc.) - dynamically populate text content
  - Should include some visual based data (e.g. images, videos, etc.) - dynamically populate src and alt

DONE Site Navigation: Completed by Cody

- Create a standard navigation to enable a user to navigate between dynamic routes
- Create a home page (OK to select one dynamic route as the default home page)

To earn a grade of A or B

DONE 4 pts: create dynamically populated 'previous' and 'next' links in HTML template files that link to the previous or next page for its data type: Completed by Martin

- Link can either be disabled or circle around when at the first/last item

DONE 4 pts: send a proper 404 error if a requested source does not exist in the database: Completed by Cody

- Can be plain text, but should be customized to the request (e.g. "Error: no data for state FB", or "Error: no data for year 2030")

DONE 4 pts: create a chart/graph to visualize the data: Completed by Charlie

- Tip: I would suggest using an existing library if generating the graphs on your HTML pages. Some possible choices are:
  - Plotly (https://plotly.com/javascript/Links to an external site.)
  - ChartJS (https://www.chartjs.org/Links to an external site.)
  - AnyChart (https://www.anychart.com/Links to an external site.)
  - CanvasJS (https://canvasjs.com/Links to an external site.)

Total Expected Score: 50 / 50