Champlain College - Lennoxville
Final project: Proof of concept

PROGRAM:	420.B0 Computer Science Technology	
COURSE:	Transactional Web Applications 1	
COURSE CODE:	420-430-LE	
WEIGHT:	7% of the final score (7% out of 29% for the final project)	
SEMESTER:	Winter 2023	
INSTRUCTOR:	Francis Gauthier fgauthier@crcmail.net	Office C-239

Context

You are a small team of consultants specialized in web applications. You are known for producing web applications managing a lot of data.
Last week, you were approached by a large governmental firm called GreenLab that specializes in greenhouse gas emissions, and they are currently developing a technology to reduce the CO_2 emissions of many large industry companies. They are about to open a new branch focused on applications to visualize the benefits of their new technology. They will offer a high-paying contract for 2 to 3 consultants for the next 3 years. Your last contract is about to expired, therefore your team will apply for the contract.

An interview and demonstration will happen on May 9th. In the meantime, you have a few weeks to prepare a solid web application that will impress the deciders of GreenLab that you are the best team for the job.

Technology stack

By talking to some employees of GreenLab and their HR representatives, they mention that they are using some technologies and would like to see an application that uses the same technology, as they already have a few employees that are developing with those.

The jobs offer mentions specifically these technologies:

- MERN stack, REST API
- Python, ETL
- Cloud computing with AWS

Proof of concept

In this assignment, you must first come up with a **proof of concept** of what you will show to the GreenLab interview.

> *A proof of concept -- evidence, typically derived from an experiment or pilot project, which demonstrates that a design concept, business proposal, etc., is feasible.*

Your proof of concept will have to provide a web application with these four different views:

- A registration page
- A login page
- A home page
- A graph view page

Registration and login

The registration and login page must be publicly accessible.

The registration page should ask for this information:

- Email
- Password
- Confirm password
- Date of birth
- Field of work (dropdown)
    - Should contain minimally *Education, Information Technology, Health and Social Services, Manufacturing*

The login page should ask for:

- Email
- Password

The home page

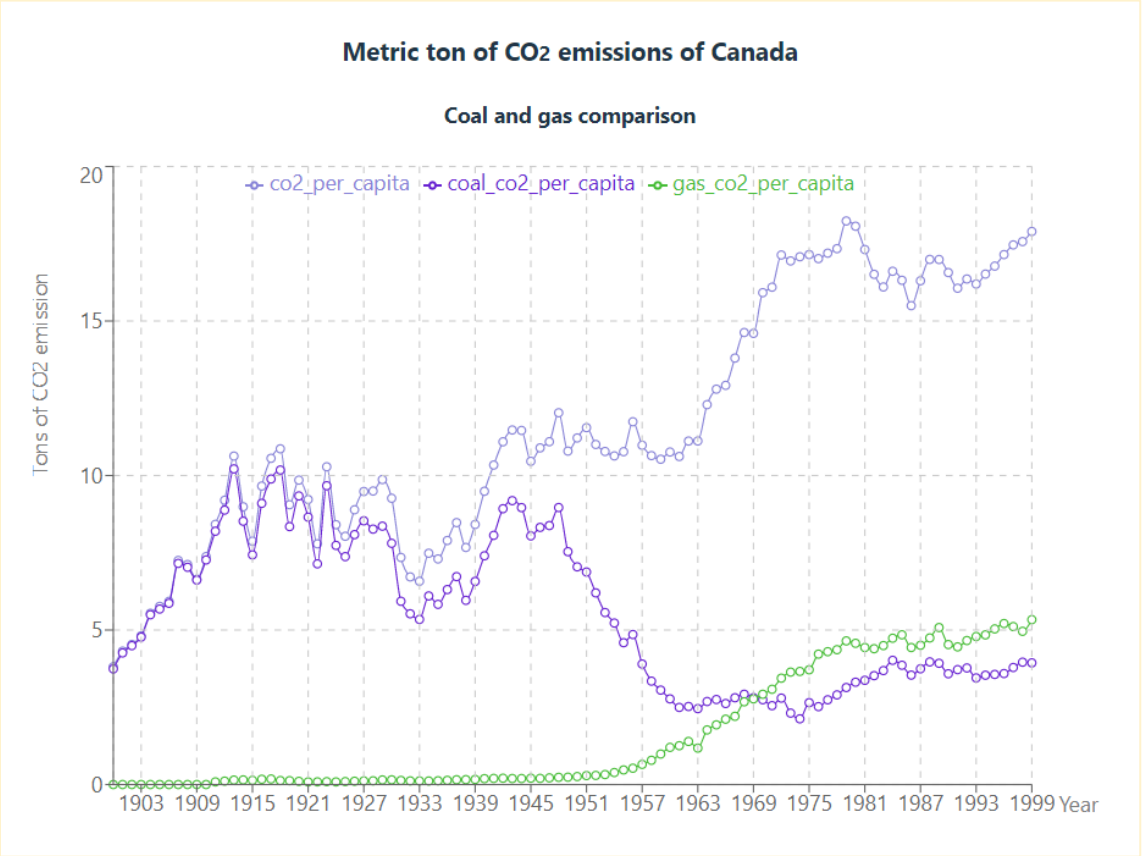
The home page should only be accessible to logged in users.
The home page should provide a simple way to access the graph view page.

The graph view page

The graph view should display one graph displaying some information about greenhouse gas emissions.
The graph should contain:

- A title
- Some labels (x and y axis)
- Some data plot

Here is an example of a resulting graph (shown in browser):



*Many options will be needed later. For example, ability to filter by country, by gas type or count type. These are not required for the proof of concept part.

Splitting the work

All companies' employees will earn a specific title when landing a job, which also determines their obligations and responsibilities within their team.
You will have to assign yourselves titles and responsibilities. Here is an example on how to split the work between team members:

Example for a team of two:

1. Web application developer
    a. Client-side and server-side generalist
    b. Focuses on User Interface and User Experience
    c. Synchronizes the work between the front-end, back-end and database side
2. Data analyst
    a. Specializes in data manipulation
    b. Focuses on extract, transform and load the data

Both team members are responsible for architecture and deploying the application

Example for a team of three:

1. Front-end specialist
    a. Client-side framework specialist
    b. Focuses on User Interface and User Experience
2. Back-end specialist
    a. Server-side framework specialist
    b. Responsible for architecture and deploying the application
    c. Ties the knot between the front-end and database side
3. Data analyst
    a. Specializes in data manipulation
    b. Focuses on extract, transform and load the data

Deliverables

Analysis (POC) – April 25th

- An analysis document that demonstrates the consultants have explored the technologies available in the market to perform the web application. First, the analysis should list all the technologies that will be used for the application. Second, the analysis should explain in detail how the data will be transformed from an initial CSV up to a displayed graph on a web browser. Cloud solutions are welcome!

- A proof of concept of a web application that meets the following criteria:

- Offers a registration page and a login
- Redirects to a home page upon successful login
- Ability to go from the home page to a graph page
- Ability to see one graph in the graph page

Final submission and presentations - May 9th - More details next week…

Submission

Submission will be made through a Github repository.
Create a private repository and add the members of your team. Also, add me as a collaborator (**frangauthier** username)
Your project must contain many files, but these files are mandatory:
At the root of the project:

- A Analysis.docx or Analysis.md file explaining your graph workflow (from DB to front-end)
- A readme.md file specifying exactly how to start the client and the server of the application.
- Your source code for your application, **organized in different folders**

The deadline for the proof of concept is **Tuesday April 25th 2023**, End of Day.