TEMPLATE ENGINE

Web Application Framework (WAF) – Fall 2024
Lab Task

Part 1. Create and implement database schema for user signup.

Part 2. Create and implement database schema for the user profile part of a web application. The following shall be the minimum profile data criteria:

- ✓ a. Name of the user
- ✓ b. Gender
- ✓ c. City
- ✓ d. Date of birth
- ✓ e. Occupation
- ✓ f. Any other information you find suitable.

→ Model
→ Controller
→ Ejs (Views)
} CRUD Operations
↓
Part 1
Part 2

→ Normalization (Schema)

Make sure that your database schemas are normalized. For example, you shall not add city data directly in the user collection (table) but make a separate collection for city names and city codes and use those city codes as a reference in the user table.

This means that city names & codes are already inserted in the collection which are used by the city doc in the profile schema as option.

✓ 3. Create the Model part (MVC) of the application for parts 1 and 2 named '*yourName*Model.js'. The model part shall include data structures corresponding to your designed/implemented database schema and methods providing data manipulations in the database.

4. Create a controller part (MVC) for parts 1 and 2 named as '*yourName*Controller.~~py~~js'. The controller part shall include data validations and business logic (if applicable). Interaction with the database shall be through the model part, not directly (you can use import for now).

5. Create necessary views in a separate folder named views using EJS template engine. Make views reasonably interactive and attractive sing CSS. (1) The database schema should be in a form and the user should be able to post data from the form to the relevant MongoDB collection. (2) The user should be able to query and perform CRUD operations to render data from the database collection. Your task is to create a small express oriented application using EJS as frontend view and express js as backend server interacting with MongoDB collection to implement server side rendering.

(1) POST
Form → DB

(2) CRUD (query)
render Data

→ Server Side Rendering

Tasks
→ Data from views to database.
→ CRUD Operations on data.