Project Requirement and System Flow
Professor: Nilay Ganatra
Students: Yau On Li & David Tang

Scope

Our web application will be a vehicle review web application. It will allow users to add their model/make/year of choice into the website that would be stored in a database on the backend. In addition, the website would allow users to search for various options, allowing them to retrieve information about said vehicles left by other users. The user will have the ability to leave their own vehicles review on the website as well.

We will consider the following parameters:

- Car Make
- Car Model
- Year of release
- User
- Review
- Ratings
- Like count

All vehicles have a make, year, and milage etc. We can store/retrieve vehicles' details in our database SQL server. We will need to consider parameters such as car_id which will be a unique identifier for the specific model, make, year of release et cetera.

The next part of our application is the ability for users to enter their own reviews for a particular car. If a car does not yet exist in the database, the user should either first add the new car title to the system or they would need to enter a pre-existing car title. Users should be limited by the maximum number of words that they are allowed to use in their review. Some car enthusiasts may need more words to express their opinions so the limit will be adjusted accordingly for their needs. The review system features a simple rating that goes from 1/10 (worst) to 10/10 (best).

We will be using HTML in the front end, allowing users to interact with our car review application. In the backend, we will be using python for the coding logic and we will use MySQL database. Specifically, we need to use python code that would receive requests from the front-end during user form submission (such as submitting a car review or when the users want to add a new car). The code would then need to be run and create some output which would need to be communicated back to the front-end, displaying messages such as “car review added, car added” or providing a list of cars for the user. We will be using a virtual environment created with command:

python -m venv env_name

and Flask will be serving as the connector between our front-end application and our back-end coding. It will be imported and used for this project with routes configured to service each of our webpages on the main page.

System Flow

Our website will feature a navigation bar at the very top with categories:

- About Us
- Home
- Add a car
- View Cars
- Add a car review

This navigation bar allows users to visit any of our other webpages no matter where they are within the website directory structure.

On a main page, it will display all the car reviews submitted by users that have previously visited the website. Users can interact with it: scrolling up and down to have a look at the car reviews.

About Us will feature details relating to the creator of said website and contact information.

With "Add a car, make, model" , users can freely add whatever car, make, model and name of their choice. There won't be any specific limitations, server will be able to handle it and security is good enough that it cannot be exploited.

With "View Cars", we can search for a car based on car title, author name, publisher name, published date etc. So that users can see if their favorite car is in the database. Every car will be stored in a database server and will be accompanied by a picture of said car.

Finally, with the "Add a car review" link, users can add their own car review and have it displayed on the main page with a time stamp. If the user enters a car that has not yet been added to the database, they have to add it first or choose another car title that the database recognize.

Website Design Concept:

Not final, might be subjected to change