# Lab: Spring Essentials

# MobiLeLeLe Application

This is an extended, better version of your last Lab – West Compass. As you can see, the database is the same and you can focus on working with the controllers. Again, we'll give you the html templates, styles and other content, but you need to write the logic of the app in controllers and services. If you have the DB from the previous lab, skip the first part.

## Data

This is the data layer of the application. There are some data object for you to implement.

### Brand

Create a Brand class, which holds the following properties:

* id – a **uuid or number**.
* name – a **name of brand**.
* **created** – a **date and time**.
* modified – a **date and time**.

### Model

Create a Model class, which holds the following properties:

* id – **uuid or number**.
* name – a **model name**.
* imageUrl – the **url of image**.
* startYear – a **number**.
* endYear – a **number**.
* **created** – a **date and time**.
* modified – a **date and time**.
* brand – a **model** **brand**.

### Offer

Create a Model class, which holds the following properties:

* id – **uuid or number**.
* description – some **text**.
* engine – **enumerated** value.
* imageUrl – the **url of image**.
* mileage – a **number**.
* price – the **price of the offer**.
* transmission – **enumerated** value.
* year – the **year** of offered car.
* **created** – a **date and time**.
* modified – a **date and time**.
* model – the **model of a car**.
* seller – a **user that sells the car.**

### User

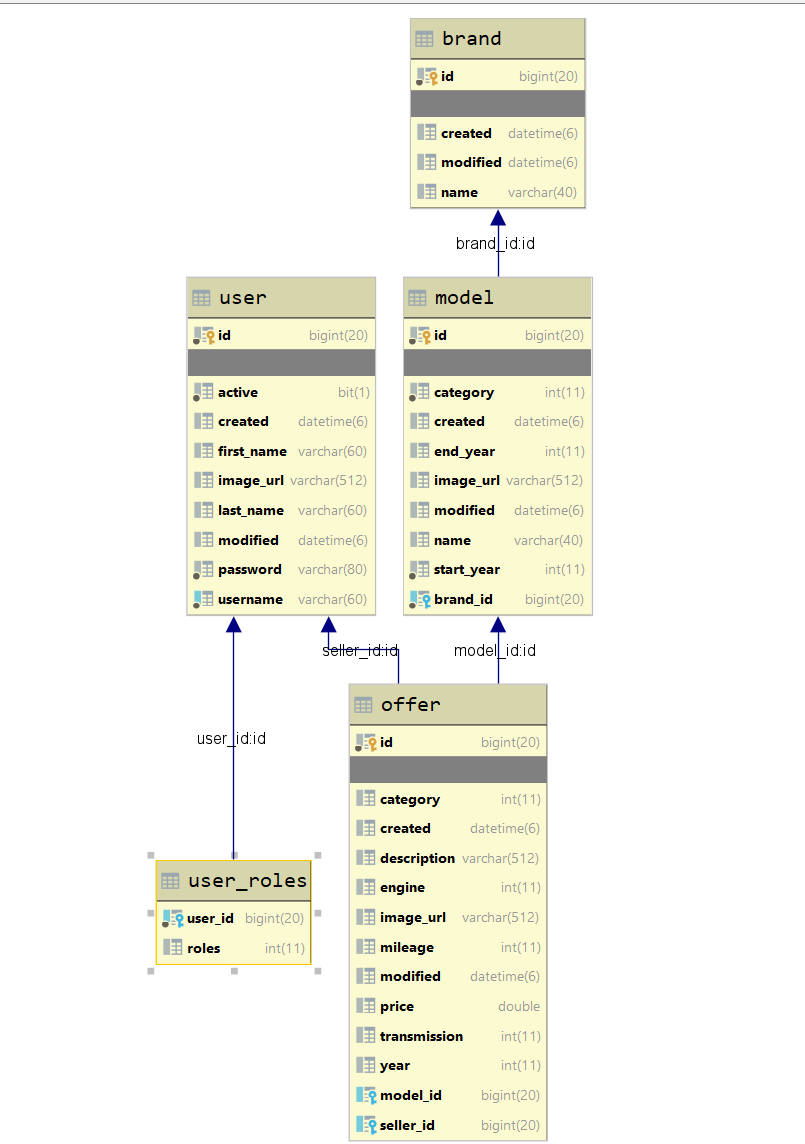
Create a User class, which holds the following properties:

* id – **uuid or number**.
* **username** – username of the **user**.
* **firstName** – first name of the **user**.
* **lastName** – last name of the **user**.
* active – **true OR false**.
* role – **user's role**.
* imageUrl – a url of user's picture.
* **created** – a **date and time**.
* modified – a **date and time**.

### UserRole

Create a UserRole class, which holds the following properties:

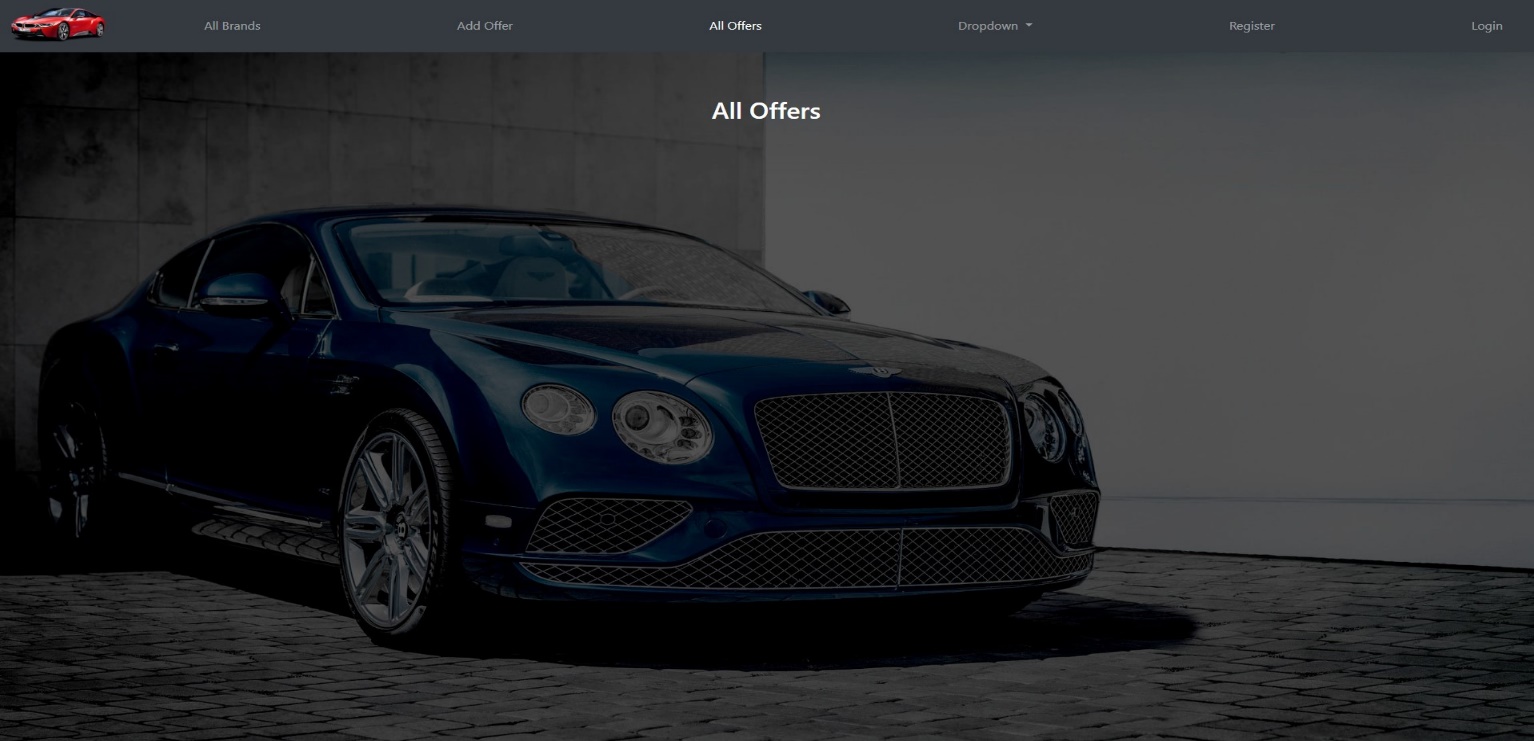
* id – **uuid or number**.
* **role** – **enumerated** value.

This is an example of ER Diagram

## Home/index - route ("/")

It should only support a **GET** request.

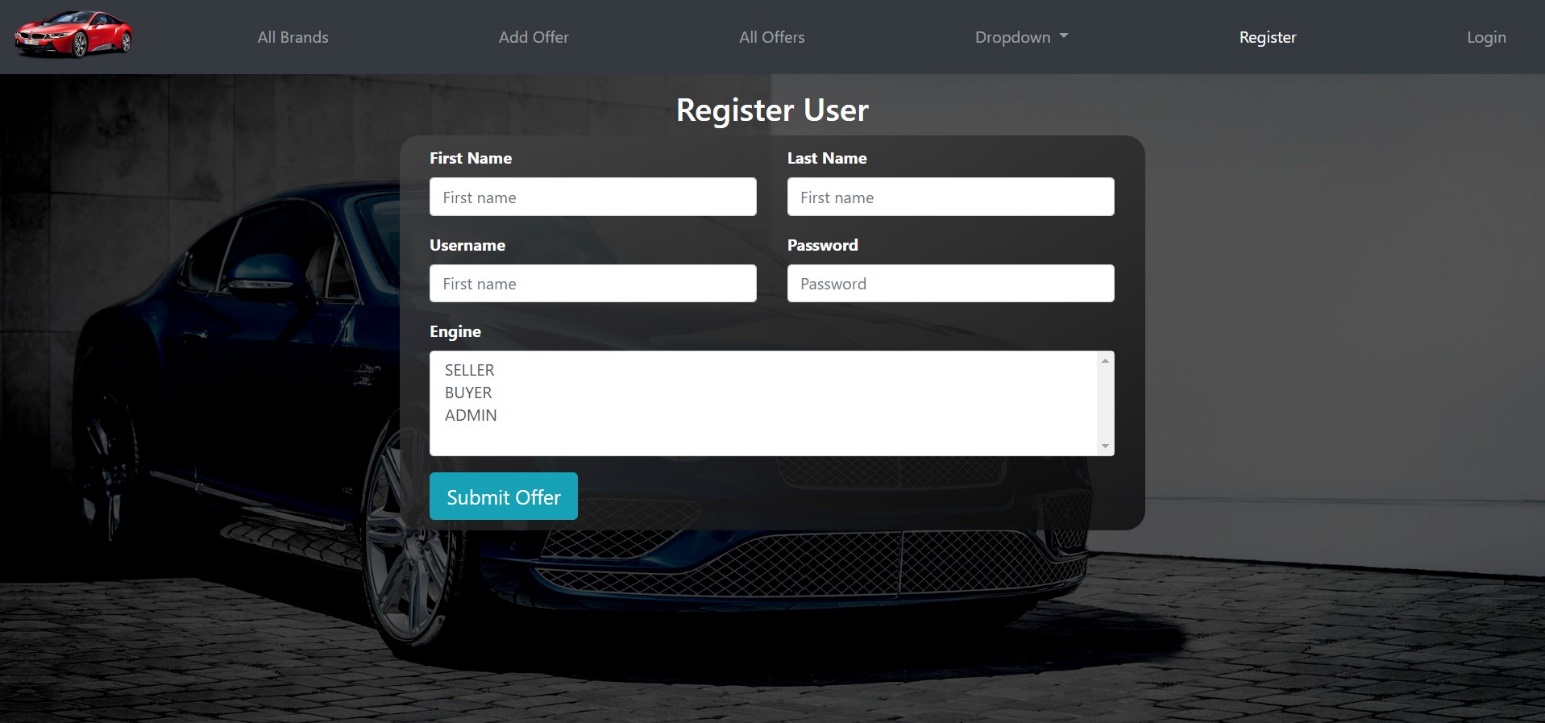
It should return the following HTML page, upon a **GET** request, when in DB **doesn't have offers**.



## Register User - route ("/users/register").

It should only support a **GET & POST** request.

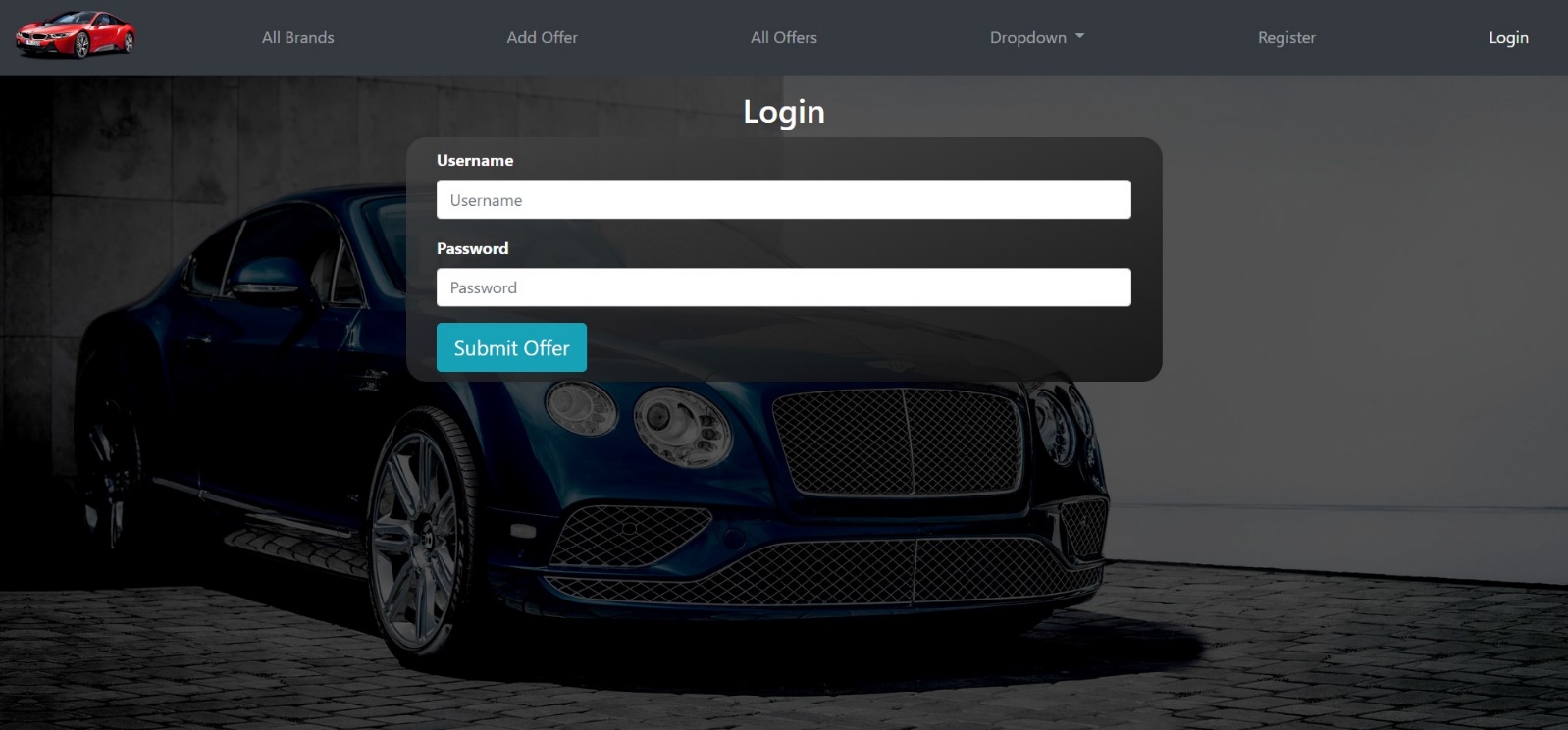
It should return the following HTML page, upon a **GET** & **POST** request.



## Login - route ("/users/login")

It should only support a **GET & POST** request.

It should return the following HTML page, upon a **GET** request.

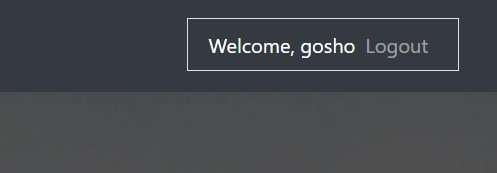


## Navigation bar when logged in

Let's see the navigation bar when successfully logged in user:

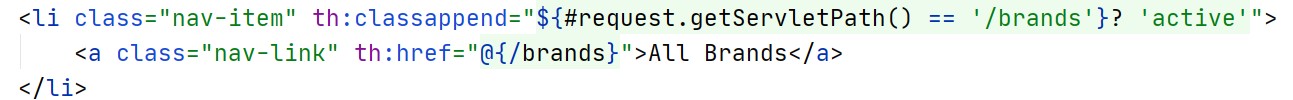
The text must be **"Welcome, {username}"**.

Logout is a button.



**Hint:**

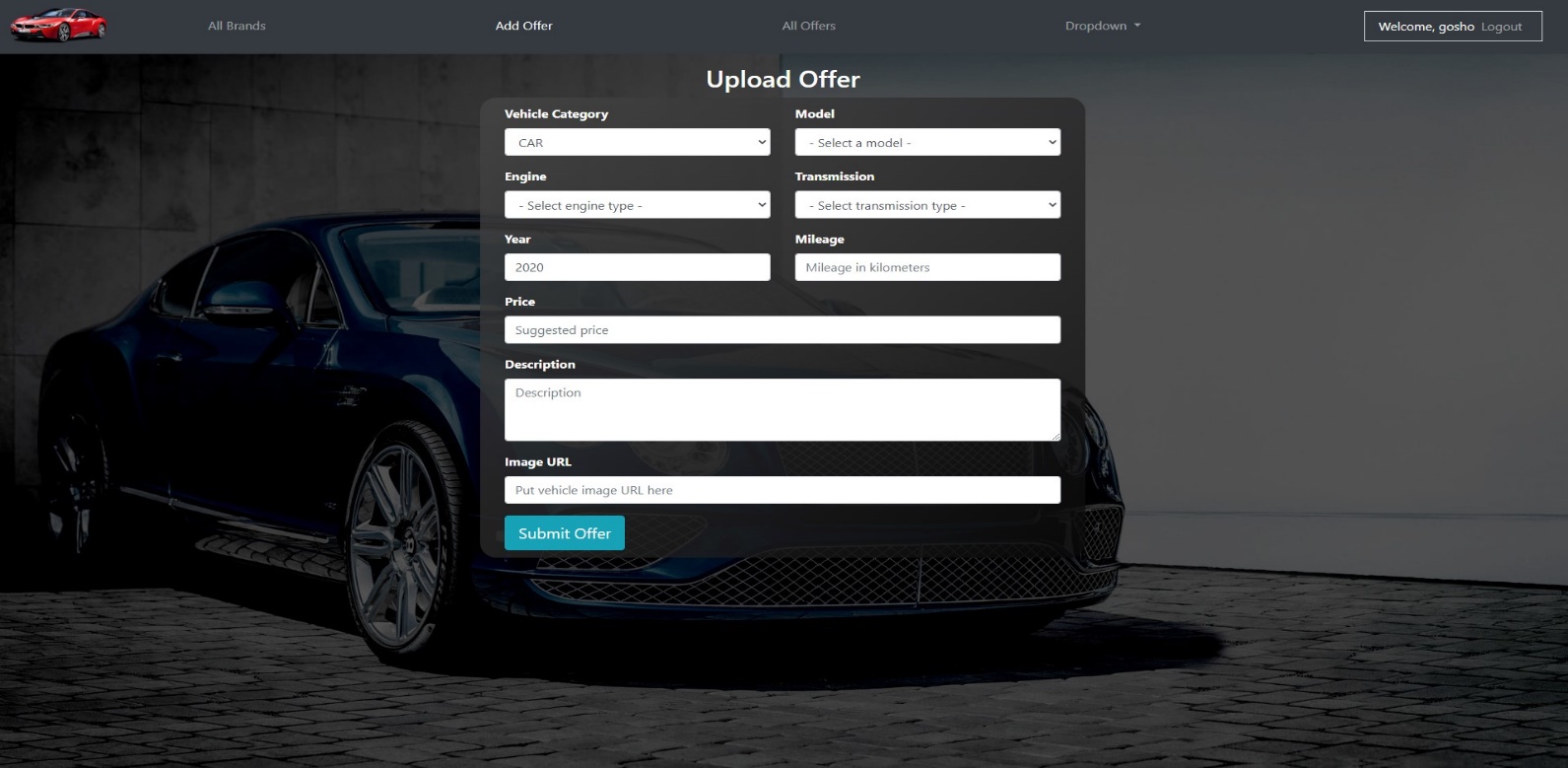
* To make menu (for example "All Brands") to be active may use this:



## Upload offers - route ("/offers/add").

It should only support a **GET & POST** request.

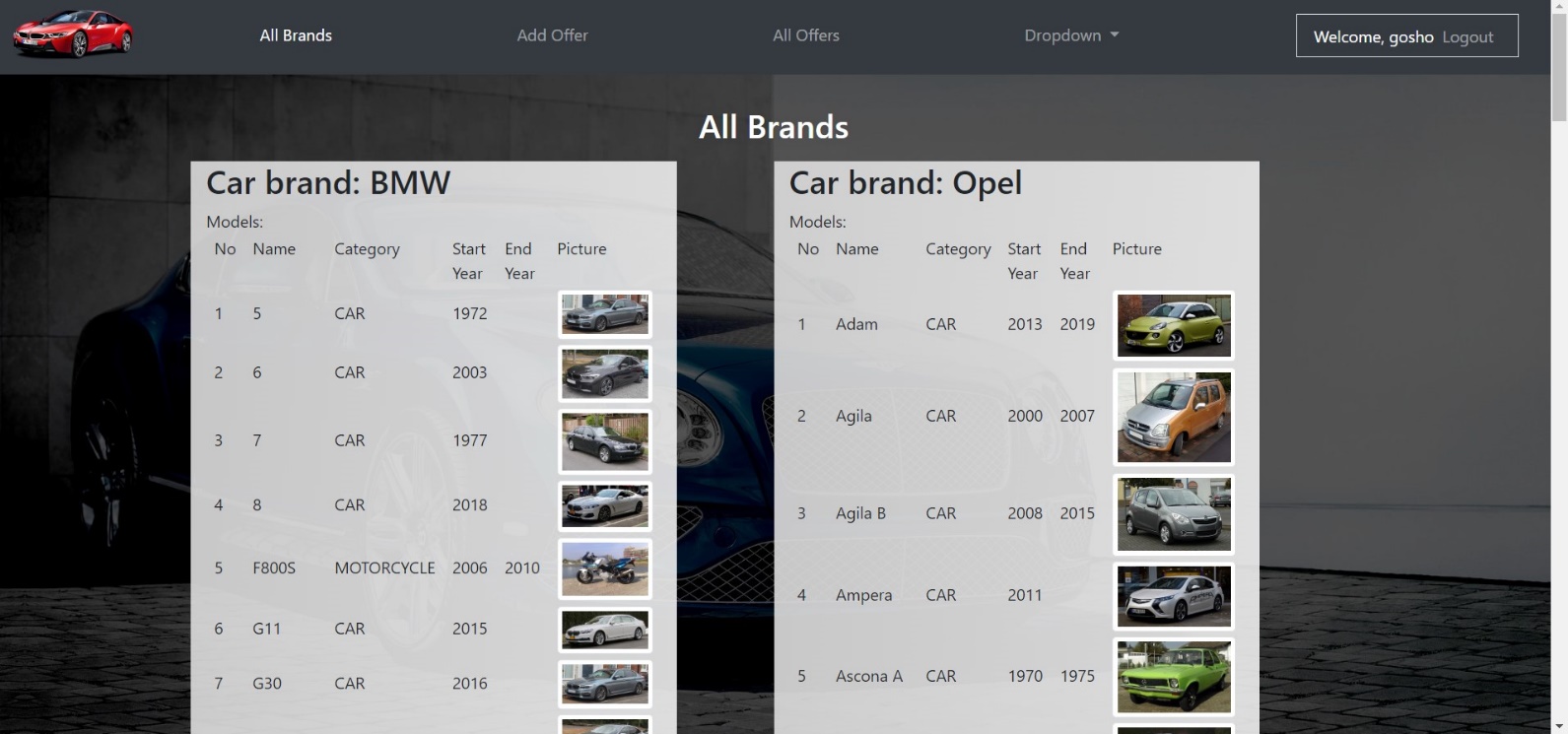
It should return the following HTML page, upon a **GET** request.



## All Brands - route ("/brands/all")

It should only support a **GET & POST** request.

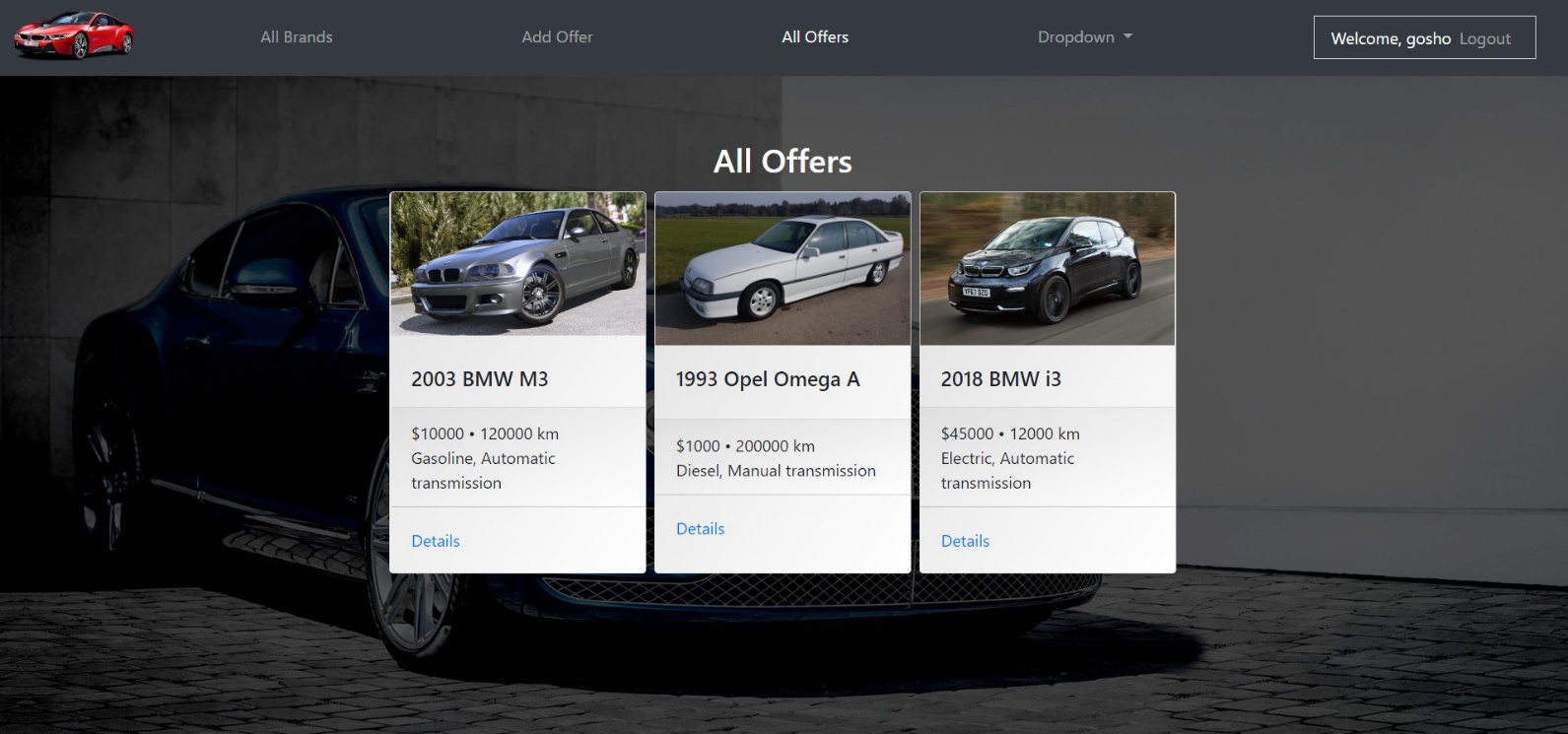
It should return the following HTML page, upon a **GET** request.



## All Offers

It should only support a **GET** request.

It should return the following HTML page, upon a **GET** request.



## Footer & Dropdown

You must implement your custom footer and add new functionalities in a drop down menu.

Feel free to add anything you want. 