

Test

Intends to create a website related to displaying ads related to vehicle rents. Vehicle owners can post their ads on this platform

A vehicle owner registers on the website and during registration the details related to the vehicle must be entered separately.

API should be created to post data and update data accordingly, consider the following arguments.

- After entering all the information correctly API must return,

```
{
  "success": true
}
```
- If not

```
{
  "success": false
}
```
- User need to register atleast 1 vehicle with the registration process. If not API must return false response.

A sample ER for the diagram attach with this. You can follow it and every symbols and annotations have relevant meanings. If you need to modify it, you can do it also.

Once an ad is posted, a website user can see it.

1. Full name of the owner of the vehicle.
2. Date of publication of advertisement.
3. Title of advertisement
4. Content of Advertisement
5. Vehicle brand and model
6. Charges per kilometer
7. AD image (not mention in ER)

Create an API for get and request related to ADs.

According to above scenario, backend developer need to create a proper API.

Developer need to use

- Python3 - Django for the Backend process with REST API. Also need to maintain default Django Super Admin dashboard.
- You should provide Postman Collection also. (Swagger also accept)
- For the database, you need to use MySQL (SQLite also accept) Design and create a MySQL database with tables to fit all the fields and data types
- Ensure your code is tidy and well documented
- All endpoints must return a JSON response
- Add validation to all endpoints where applicable

Summery of the API Endpoints.

(you can decide the number of the API endpoints with your coding styles and assumptions)

Create User and vehicle details (one or two endpoints)

Update User and vehicle details (one or two endpoints)

Create Ads

Update Ads

Delete Ads

View Ads

Developer need to push their codes to GitHub and share the repository link with us with postman collection

If you follow different methods or procedure, please add a updated readme.md file with Git Repo

