**MERN Stack Developer Project**

**Case 1: To Do List Application**

* Login and Logout Functionality
* Store the user in the DB
* Handle Login and Logout via API
* Implement the Session and token handling
* Shouldn’t allow user to navigate to other page manually or any other means without login Restricted access needed.
* **Create a schema** for storing the user details
* **Implement API** to store the details of the registered user.
* Implement required validation to be done to handle the user details before storing into DB.
* Creation of Token and Deletion of the token.
* Config file to handle the configuration related like (DB link, token key, any other required) for the different environment (Prod,Staging,QA,Development)

**UI and API Work: -**

* Login and Registration screen.
* Logout Functionality
* Nav Bar to Navigate the different screen.
* Once user register successfully, user will get the email.
* All Mails trigger from backend.
* **Separate Screen:** Homepage is the Main page**;** we are showing some message and image based on the car booking related.
* **Separate Screen:** Booking slot, when user clicks the booking slot, we are showing one popup, in this popup we are showing state and district
* State and district come from backend.
* Based on the state district will appear.
* Details needs to be captured like state and district.
* Based on the state and district car will be available.
* Each and state and district have 10 cars.
* when we submit the details, popup will be closed.
* **Separate Screen:** Based on the state and district we will show the car details and price information using card view.
* **Separate Screen:** User can click any one the car, it will navigate to another page
* In this page, we are showing the car details and price information etc.
* Same page, they have option called book slot.
* If you click the book slot, we are showing Confirmation popup Message.
* If user clicked the yes, User get the mail
* Details need to be captured like car details, price, slot timing, Shift, car type, policies, payment type.
* Details are stored in the backend.
* Once user gets successfully booked, User gets email.
* **Separate Screen:** Thanking page for user.
* Same page option called Homepage Link
* **Provision** to mark the To Do Item as Completed update the dB via API and store when it was completed.
* **DB Schema -** Needs to develop for the storing the data related to the TO DO Item.

**Note:**

1.Web Site will be responsive.

2.Navbar will be common for all the pages.

3.If you familiar with redux, you can use it.

3.After code submission code review will happened post successful working scenario of the application.

4.Followed by Code Review Question will be asked based on your coding standard and coding aspect of what you have done.

5.You can make website look and feel better as per your creative skills. You can use the Material UI, Bootstrap any 3rd party libraries.