

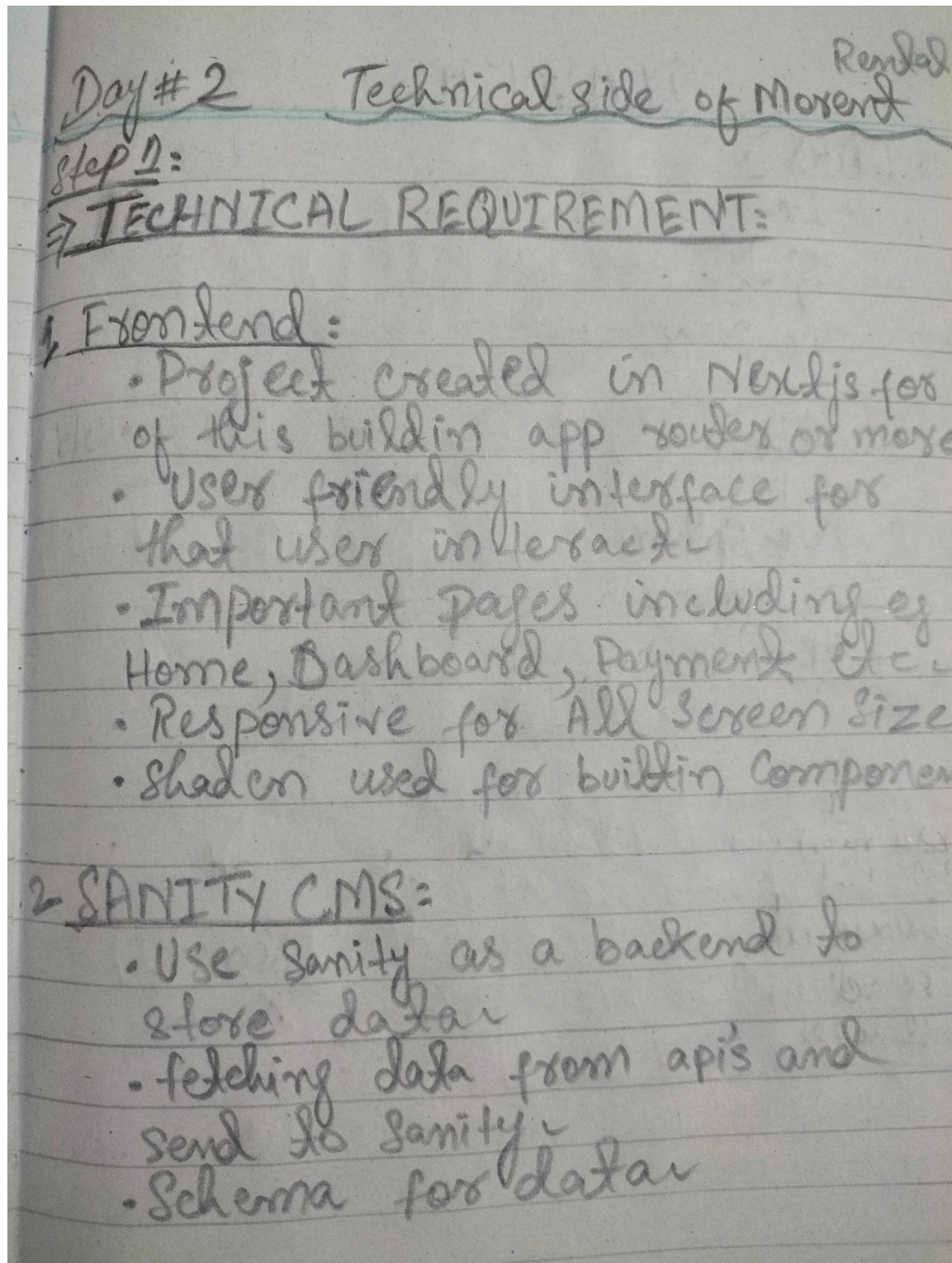


Morent Car Rental

Day2

On this day we created technical foundation docs in which include the technical requirement and how our web work through diagram we have explained and how data will

send and fetch from sanity and data schema design, Api Endpoints

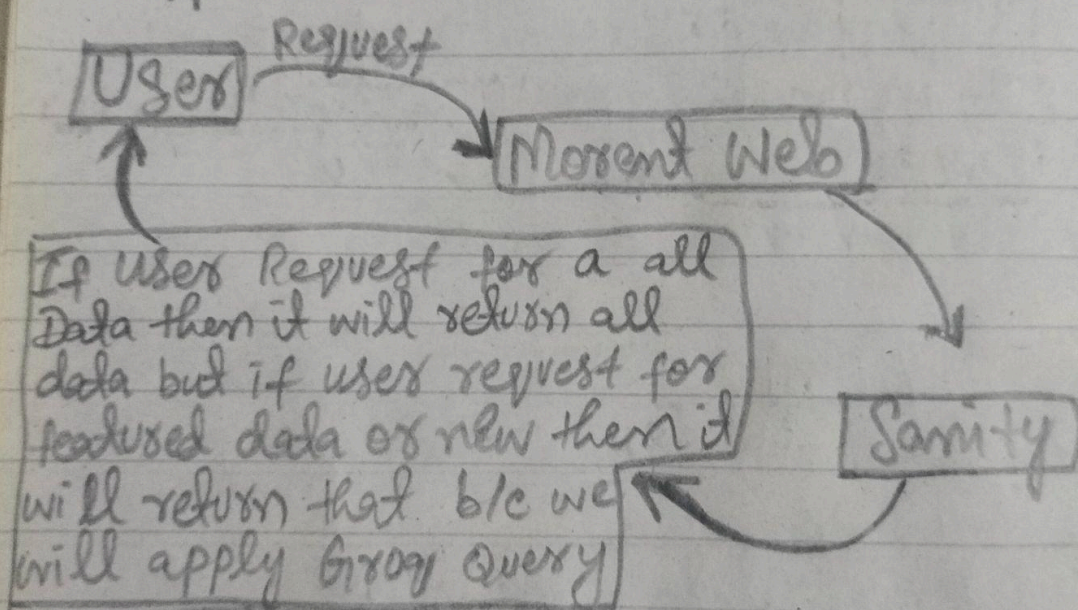


3 Third Party APIs:

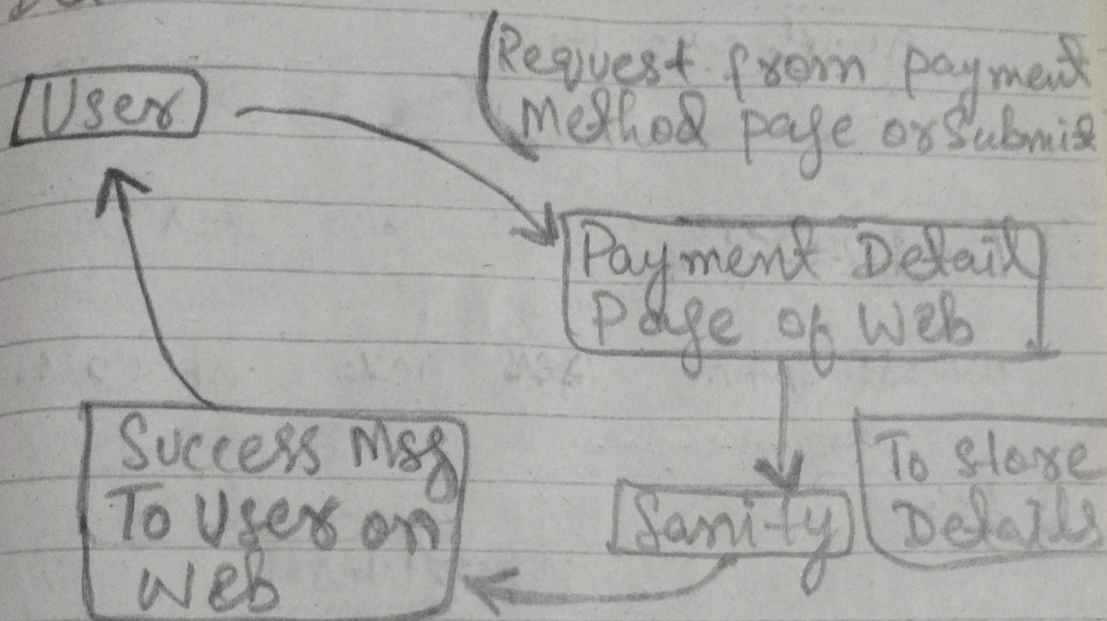
- Integrate APIs for payment gateways or data fetching

Step 2: Design System Architecture

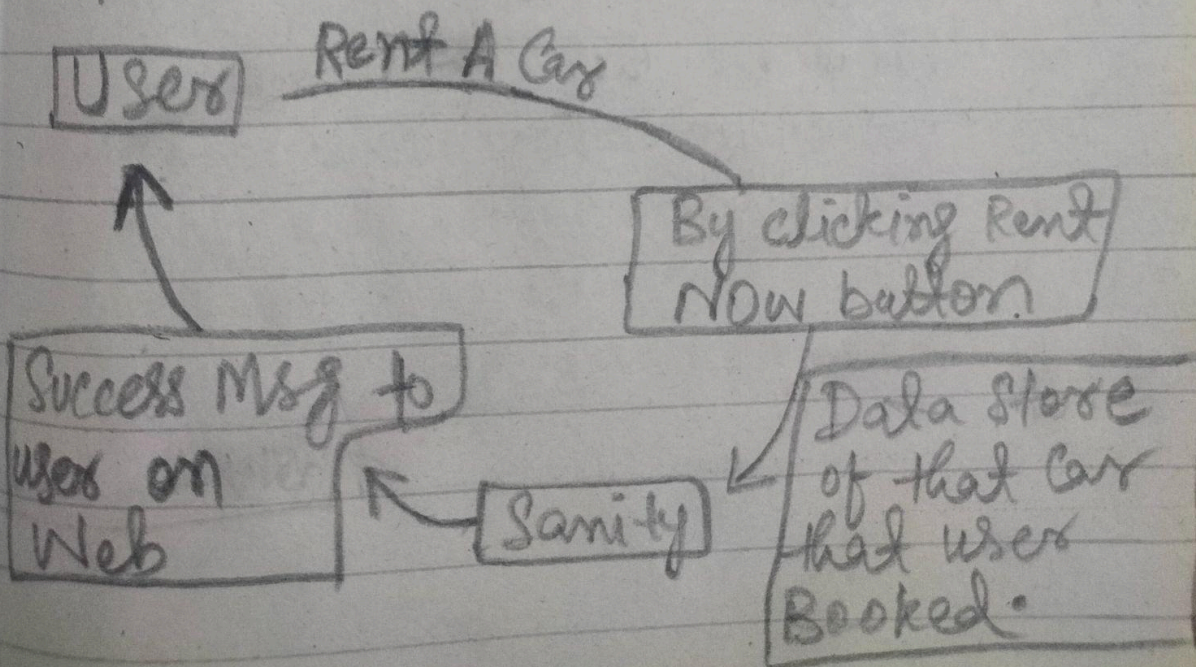
→ Suppose When User open Morent web



→ If User Type and Submit Payment Details.



→ If User Rent a Car.



Step 3: PLANNING API Requirements

① /Cars:

- Method: GET
- Response: All Cars Data.

② /Car/:CarId:

- Method: GET
- Response: All Detail of Car.

③ /Payment:

- Method: POST
- Response: paymentId, status

Step 4: Technical Document

Marketplace Technical Foundation Moxent Car Rental

* System Architecture Document:

- We have imp. pages likes Home, Dash board, Payment, Car Detail etc -
- Where user open web then home page will be open where user can see featured car, recommend car etc.
- Where user go to Payment page there user type their payment details and submit then their info will store in sanity.
- When user go to Car detail page there user can see more for that car like reviews, Description or more and can book the Car by clicking on the Rent Now button.

* API SPECIFIC DOCUMENT *

① /Cars:

- Method : GET
- Description : To show all cars.
- Response : All cars data.

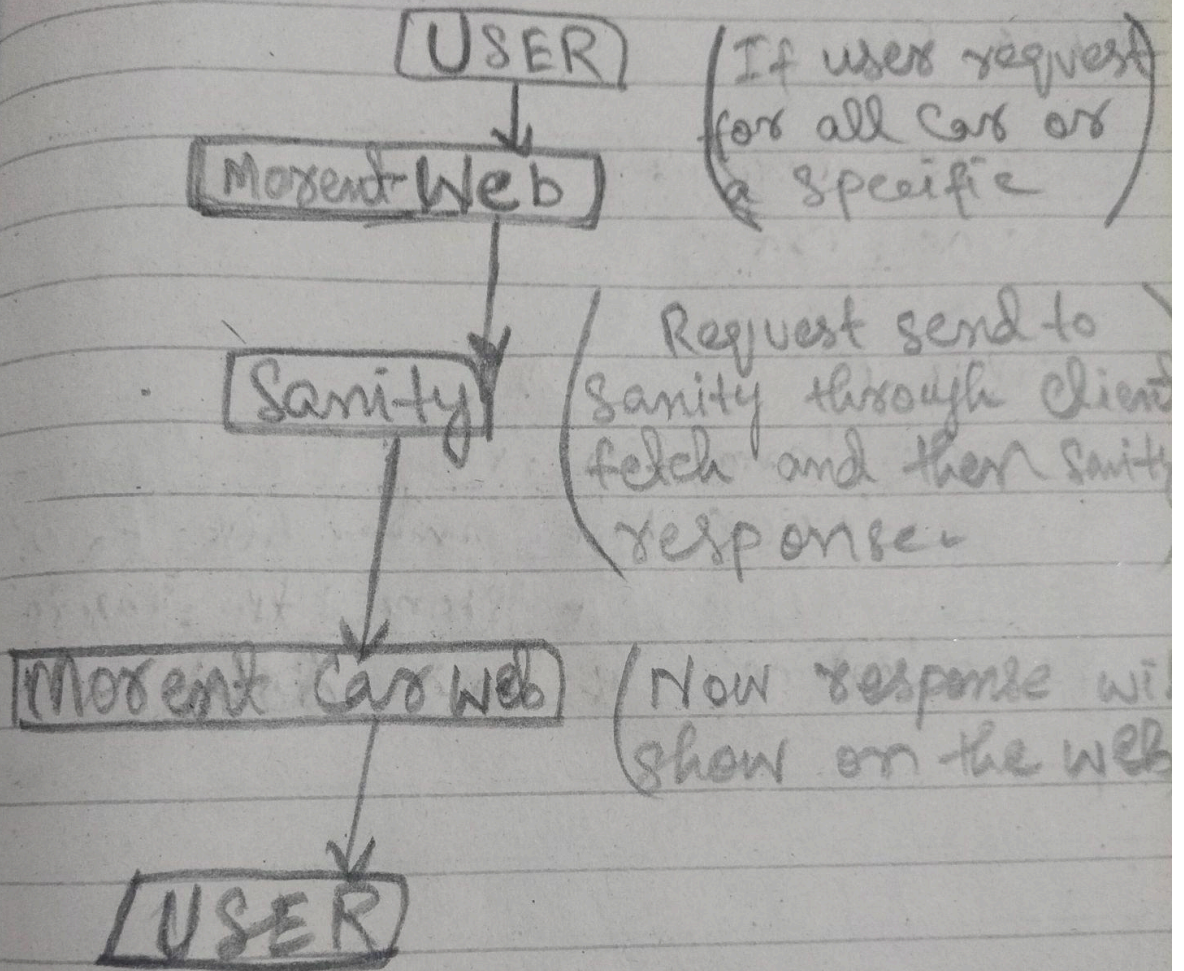
② /Payment:

- Method : POST
- Description : To submit detail of payment to verify
- Response : pay Id, status.

③ /Cars/:CarId

- Method : GET
- Description : To Car Detail page
- Response : All detail according to the Id format

* WORK FLOW DIAGRAM:



* If user Submit payment Data then also follow this above diagram

* DATA SCHEMA DESIGN:

* For Cars:

```

export default {
  name: 'cars',
  type: 'document',
  fields: [
    { name: 'name', type: 'string', title: 'Car Name' },
    { name: 'price', type: 'number', title: 'Price' },
    { name: 'description', type: 'string', title: 'Descrip' },
    { name: 'rating', type: 'number', title: 'Rating' },
    { name: 'PassengerCapacity', type: 'number',
      title: 'Passenger Capacity' },
    { name: 'GasolineCapacity', type: 'number',
      title: 'Gasoline' },
    { name: 'Steering', type: 'string',
      title: 'Steering' },
  ]
}

```