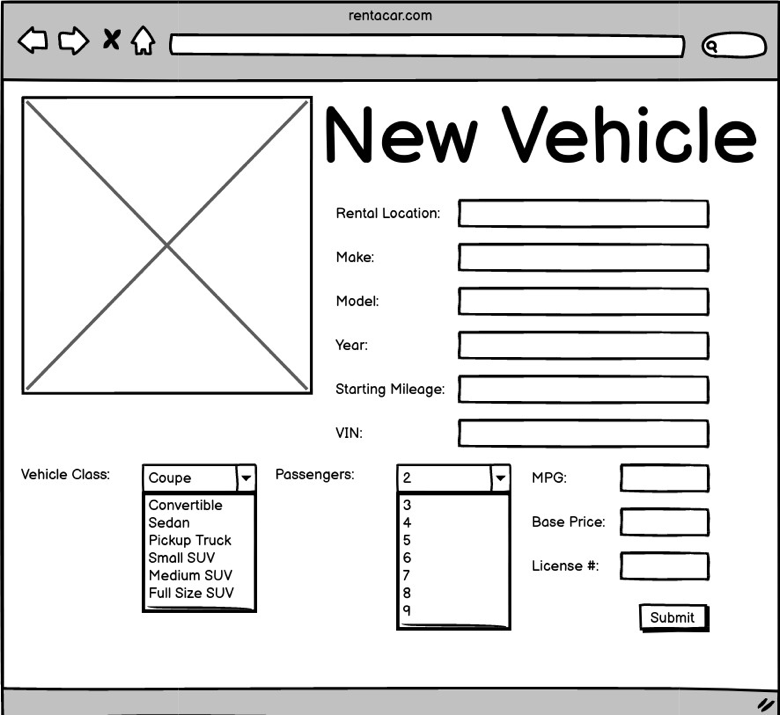
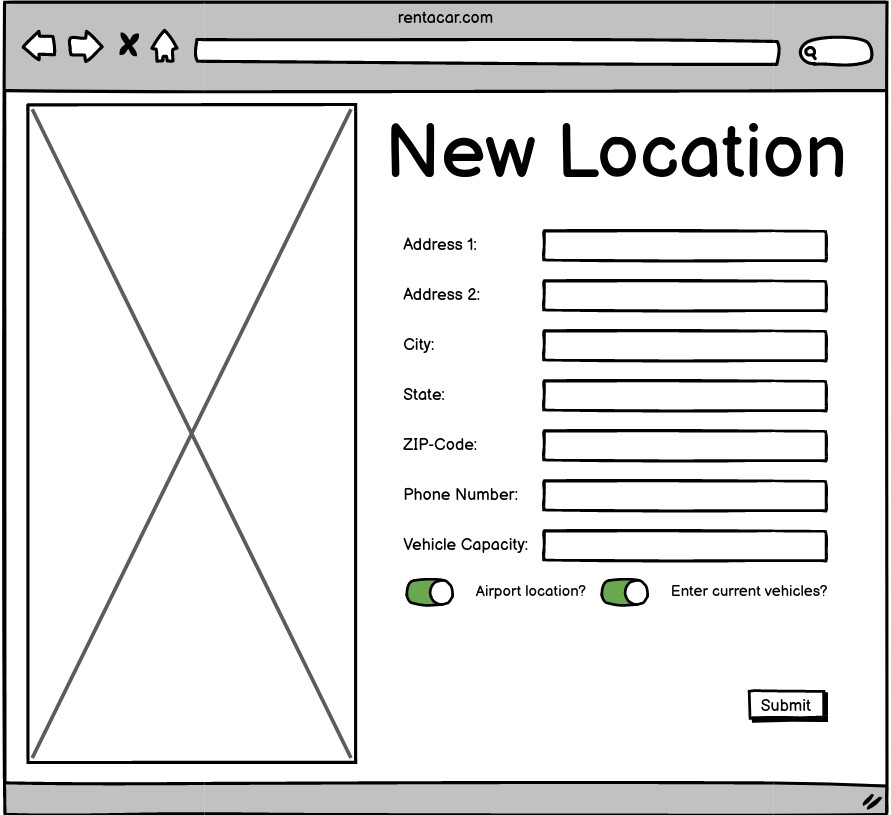
UI UPDATES



Graphical user interface, application

Description automatically generated

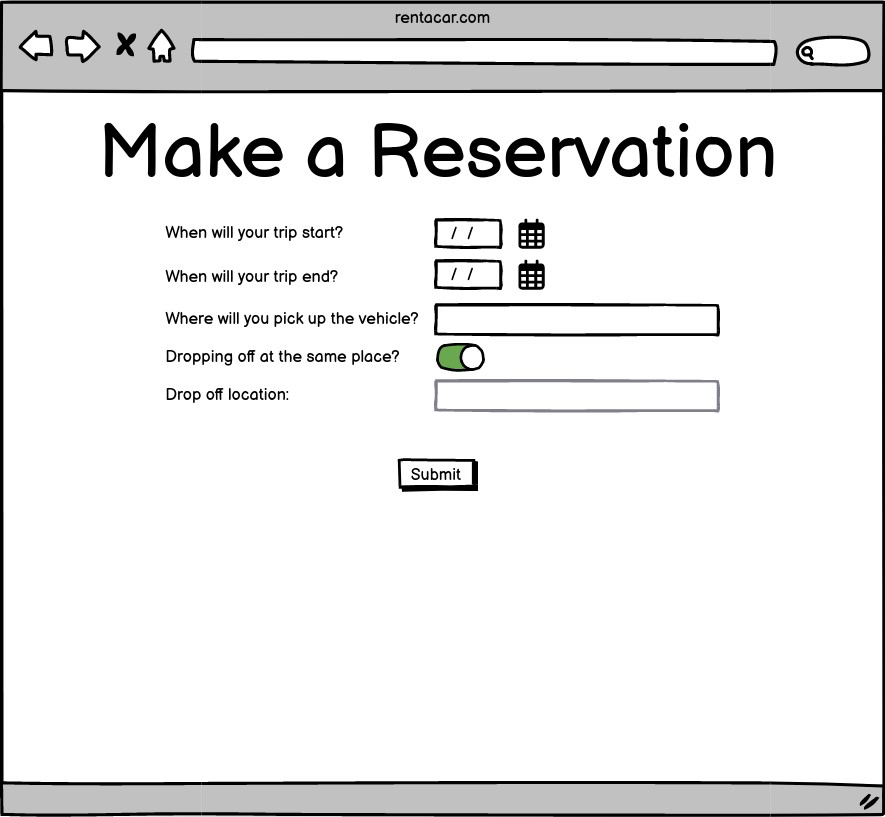
The New Vehicle screen is very close to the original idea. All fields are still present. I just changed the image field to a textbox, because storing images on the DB is not a great idea.



Graphical user interface, application, Teams

Description automatically generated

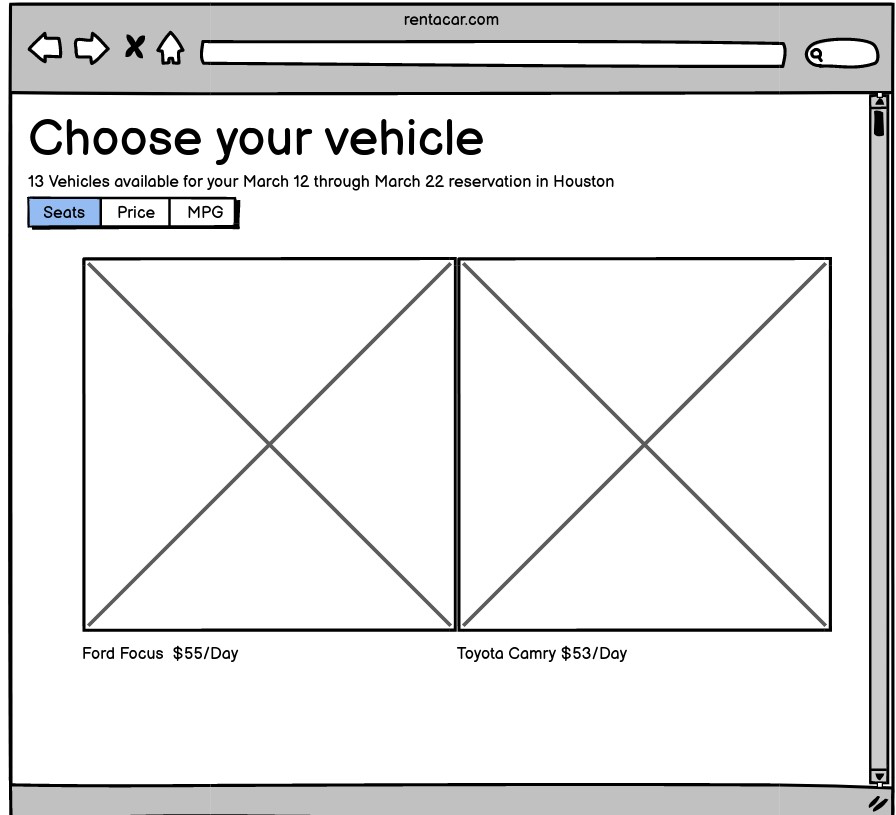
The New location page is very similar to the new vehicle page. The same things had to be changed regarding images.



Graphical user interface, text, application

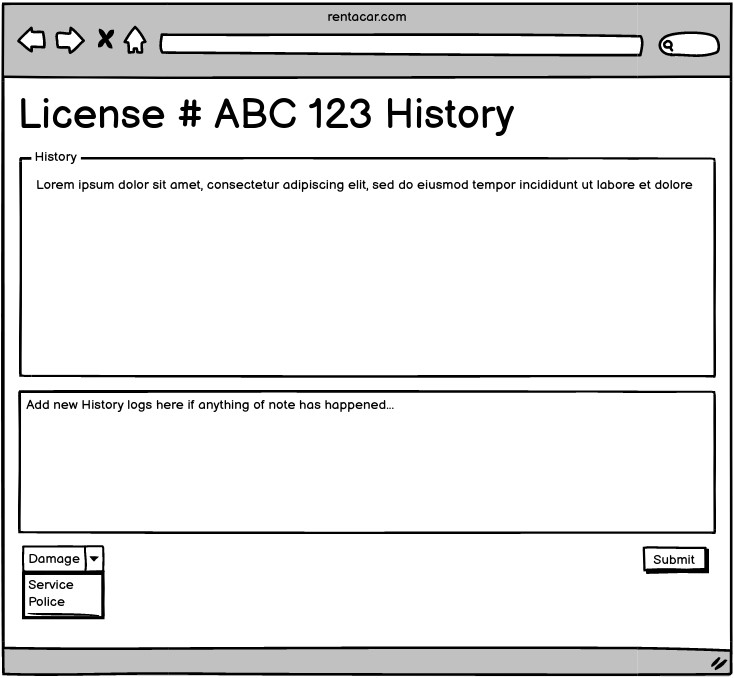
Description automatically generated

The new Reserveation page is very much the same as the initial concept. All the same fields are present. That being said, there are fields for copying both the date and location because maybe there is a one day rental within a single city.

Graphical user interface, application

Description automatically generated

The vehicle selection screen retains much of the same functionality as the wireframe. It has not gotten the filters yet, prices are also hidden within the click through on the vehicle in order to hopefully retain user attention longer.



Graphical user interface

Description automatically generated

The History update page now shows the vehicle’s specs as well as past history items. History is stored as an array of strings on the DB, so types are not necessary

TESTS

Most testing for my application will be done via Puppeteer or some similar front-end manipulation tool. This is because this application does not implement any significantly complex algorithms.

* Test login
  + Should be able to log in to front end as
    - Admin
    - Employee
    - Customer
  + Each type will be verified by checking the contents of the navbar
* Test Create and delete items
  + Each type of collection
    - Vehicle
    - User
    - Location
    - Reservation
  + Should get a success message or redirect upon completion
  + This will test the integration with backend via front end
* Test Lifecycle of rental
  + Steps involved
    - New rental (same day)
    - View reservation
    - Add specific vehicle
    - Checkout (no payment)
    - Check back in
    - Update history (possibly)
* Test Authentication
  + If not logged in should redirect to home page
  + If logged in with wrong permissions, should redirect to valid page

Coverage of tests is nearly complete. Being able to run all of these processes should test the front end design, the back end connecting to the front end and modifying data as necessary, and the integration between the two.

In terms of testing nonfunctional requirements and UI requirements, a good strategy would be user testing with friends and family to critique usability as well as functionality.