

# Journal Jin Yang

The project is to write a website for a small business that offers vehicles to be personalized and bought online. Professor have finished the part of code for state management, routing, and injecting data. Followed is my writeup for the project.

## **Week 9, March 13 - Team splitting**

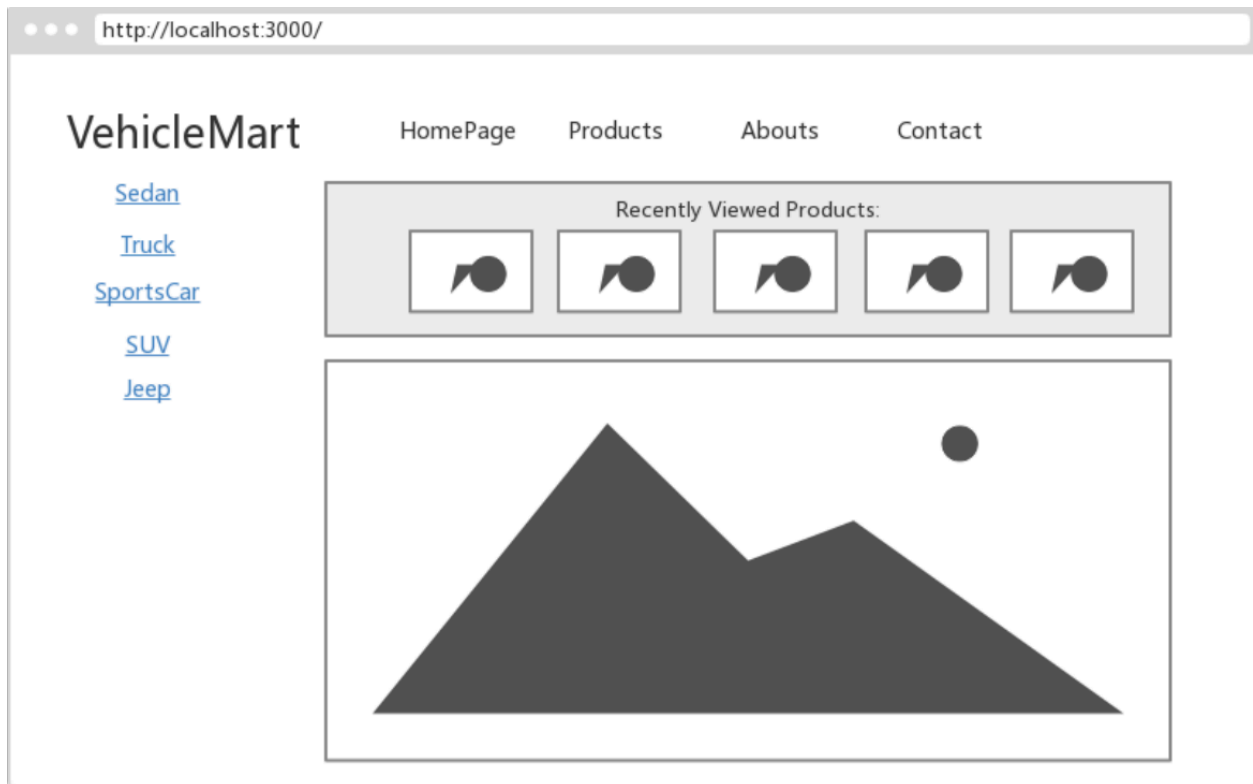
After the splitting of teams, we met each other and discuss about the project. We decided to separately go through all the code and requirements before diving into the process of final project. We also opened a slack channel for further use.

## **Week 10, March 20 – Look and feel, Wireframes, sitemap**

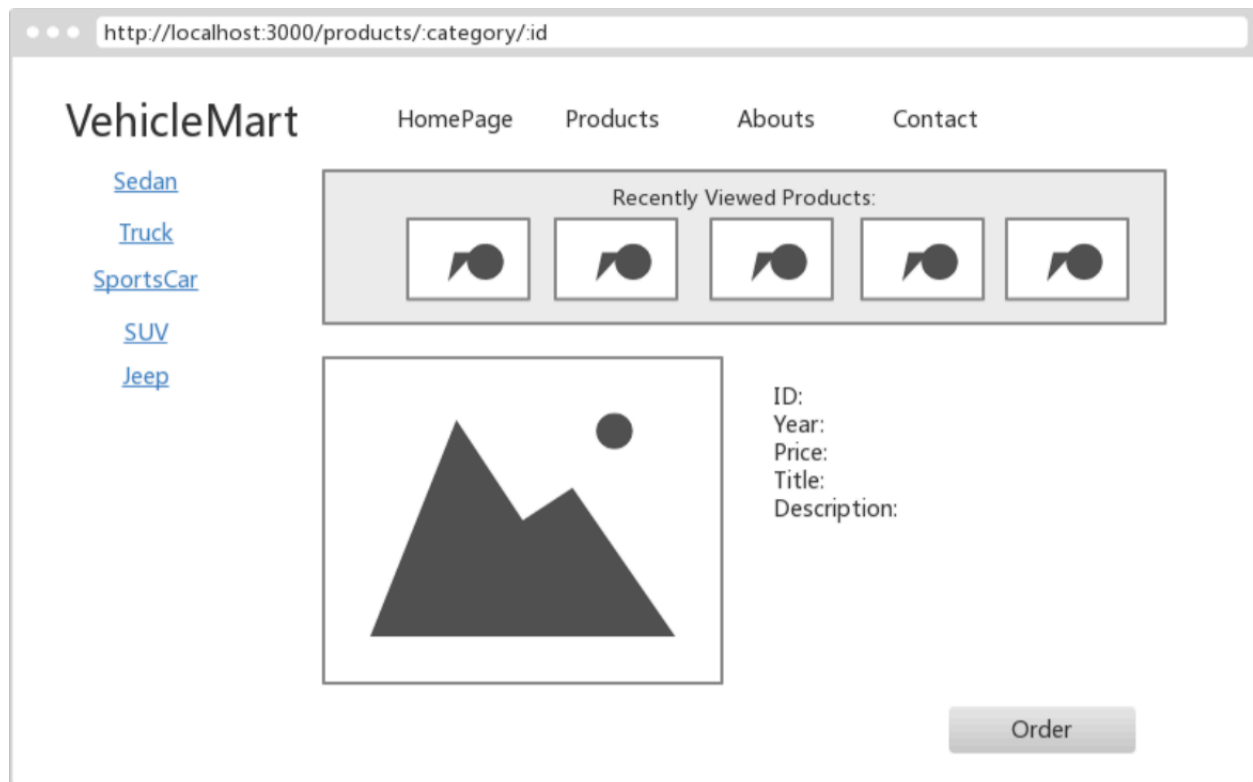
After thoroughly checking the project, we draw the sitemap and wireframes of our website.



We discussed about the homepage together, and this is the wireframe of homepage.



The original wireframe of product page looks like this, but we have made a few changes later.



Since the order flow is a rather heavy part in the project, we decided to do it together.

The screenshot shows a web browser window with the URL `http://localhost:3000/order/1`. The website is titled "VehicleMart" and has a navigation bar with links: "HomePage", "Products", "Abouts", and "Contact". On the left side, there are links for vehicle types: "Sedan", "Truck", "SportsCar", "SUV", and "Jeep". The main content area is divided into two sections. The top section, titled "Recently Viewed Products:", contains five placeholder images of a car. The bottom section, titled "Customize your choice", features a large image of a car and a series of four configuration panels on the right. These panels are labeled "Power", "Exterior", "Interior", and "Premium options". Each panel contains a dropdown menu for selection. The "Power" panel has an "Engine:" label. The "Exterior" panel has a "Color:" label. The "Interior" panel has a "Number of seats:" label. The "Premium options" panel has a "Hood ornament:" label. A "Next" button is located at the bottom right of the configuration area.

VehicleMart

HomePage Products Abouts Contact

[Sedan](#)  
[Truck](#)  
[SportsCar](#)  
[SUV](#)  
[Jeep](#)

Recently Viewed Products:

Customize your choice

Power

Engine:

Exterior

Color:

Interior

Number of seats:

Premium options

Hood ornament:

Next

## Week 11, March 27 - Starting of project

We decided to start with the order flow part of the project, since other pages like homepage, thank-you or 404 seems to be just some simple mortify of CSS file. YuXin and I am responsible for order step 2 and summary while other two members working on the step one.

To start with, I checked the wireframe we draw at the beginning of the project.

# VehicleMart

[HomePage](#) [Products](#) [Abouts](#) [Contact](#)

[Sedan](#)  
[Truck](#)  
[SportsCar](#)  
[SUV](#)  
[Jeep](#)

## Recently Viewed Products:



Personal

address

Shipping  
Address

Last Name:

First Name:

Date of birth:



phone number:

cell number:

Address

City:

State

Zip code

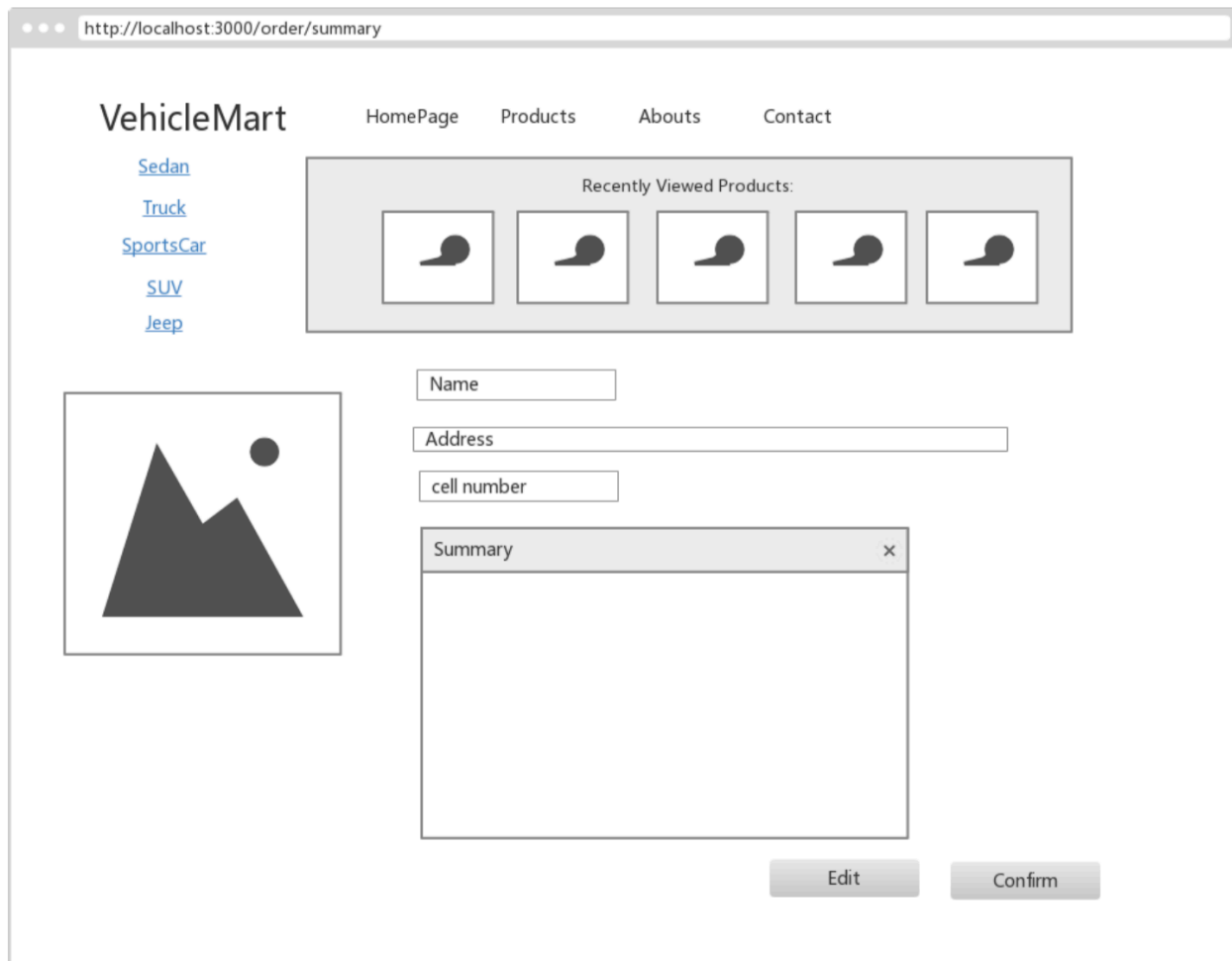
Shipment  
address:

☐ Same as Shiment address

☐ Enter new address

Address

Next



Based on the wireframe, I wrote the code for the two pages and added some CSS file. I knew it is better to work on the function first and then on the CSS file. But it felt much better to check a website with neat look than randomly distribute components, right?

## Week 12, April 3 – the changing point of project process

At first, we separated the project based on the function, but then we found out that when two people are making changes at the same time or someone edit the CSS file, much time will be wasted on merging the code. Also, there is not much time left for us since the usability test is next week and we have not been working on the other pages. We then decided to each claim several pages so that we will not be modifying other people's code. My responsibility is to do the order step 2 page as well as the summary page.

One of the few problem I need to work with is when user selects to use the billing address as shipment address, it will automatically fill in the form and the fields should be uneditable. At the beginning I thought I can manipulate DOM to change the value of the shipment address fields. However, after some searching I learned it is inappropriate to use `querySelector` in React. Also, I have thought about using `refs` attribute. Eventually, I put every change in state after reading the React Tutorial. This is the final effect of this part.

Address Information

\*Billing Address  
1260 Republican Street, 202

\*City  
Seattle

\*State  
WA

\*Zip Code  
98109

\*Shipment address ☒ Same as Billing address ☐ Enter new address

\*Shipment Address  
1260 Republican Street, 202

\*City  
Seattle

\*State  
WA

\*Zip Code  
98109

### Week 13, April 10 – usability test

I have not realized the importance of usability test until I did one. One thing I learned is that users will not do exactly what you think they will do, and they will be affected by some stupid CSS and decide not to use your website!

This is the note for what I need to work on after usability test and group discussion.

#### Order Step 2:

- Images viewed should be added Enter new address hidden
- CSS changes
- Validations for input fields

#### Summary:

- Car image
- Car Information
- Car customization data should also be displayed
- CSS

Validation is the most difficult part of the project from my point of view. First I think of using built-in form validation that HTML5 provided, but then I found that the use is limited and are unable to be customized. So I searched it online and self-learned regular expression and use it to validate the user input. Also, the error message will only appear when the user input is invalid.

I implemented this by adding a variable in state and use conditional operator to decide whether to show it.

Personal Information

\*First Name  
Jin666

\*Last Name

First Name should be 2-20 letters

Please fill out this field.

\*Date of birth:  
mm / dd / yyyy

\*Phone Number  
2068888888

Cell Number

```
const regName = /^[a-zA-Z]{2,20}$/g;
if (regName.test(value)) {
  this.setState({
    isLastNameValid: true
  });
} else {
  this.setState({
    isLastNameValid: false
  });
}
```

## Week 14, April 17 – final change

Another problem I had was to display the information in the summary page. It was not until then that I realize that if CSS style is added to a label, it will be implemented to the entire project even though CSS module is used. After making changes in my module.css file, the display of other page was affected, so I rewrote all changes I made on labels.

## Summary

Below is screenshots of order step 2 and summary:



Although we do not need to write the connect logic for the project, I have read through the code professor gave us and I really have learned a lot. The use of bind function, react-redux, as well as react-router. This experience has taught me so much.

Mar 3, 2019 – Apr 17, 2019

Contributions: Commits ▾

Contributions to master, excluding merge commits

