From this assignment, I have learnt the basic use of html5, CSS, LESS, and JS. It’s quite easy to get start, but when I got deeper into this assignment, I have confronted something that takes time. For example:

1. CSS is cascading, so the elements are relational and hierarchical. The attribute of the parent influences its children. In Less, the relationship is a little clearer because we can use nesting, but still I need to pay attention to the attribute value. Like position and display when I tried implementing “dropdown” of the LOGIN button in index.html, and “footer”.
2. The use of CSS is very flexible. Before coding I have to design each section carefully, thinking about their subordinate relationship, which attribute belongs to which part. I have tried to reduce the repetition of .css code and make the code look graceful. But it’s still a large work.
3. The selectors of CSS have different types and combinations. They make the coding more clear and effective.
4. To avoid code repetition, I found a method to use a html as a reference of another html file. It only needs some .js function.
5. Still I am confused with the blank space at the right of the home page. Some one solve this by add a “overflow-x: hidden”, but it does not in fact clear the bug. I have to debug each line to find which element has caused this.
6. I also looked up the difference of section and div.
7. When I was designing the navigation bar, I found that not only did I need to set the position as “fixed”, but also I needed to make its “z-index” value the largest, or it sill will be covered by other section.
8. I wanted to get the data from a form, and directly render another .html with this data, without a database. Ajax may help implement this.
9. There are many ways to implement center horizontally and vertically. The vertical style is a little more tricky than horizontal. Engineers online post these methods. It’s interesting.

Because I am not familiar with front-end design, I think I have a large amount material to read and practice. The first page, index.html cost me 2 days, including reading tutorials, coding, and modifying. Reading and thinking took much more time. Totally, I spent maybe 24h within a week.

Intensive coding is definitely a good way to improve. The quantity of coding is subtle. We need to find problems by struggling with codes, but figuring out problems by thinking and reading technical material take time.