**Assignment group 10.3 Summary**

Our team are responsible for testing codes. We separated our jobs into two different parts: front-end testing and back-end testing since the other two sub-teams are working on them.

**Front-end testing(By Yuyang Wang, utorid(wang1423)):**

For front-end testing, we use python unittest to verify the code.

The reason why we choose pytest is because the simplicity and tidiness.

It is easy to use python unittest and import useful packages, also, test functions are very clear to testers. For our front-end html files, we imported “BeautifulSoup” package as a useful tool to examine the elements in the file. The package provides many useful functions that help us locate elements and find their relationships instantly. For example, in our front-end test code, get\_text() can be used to find whether we contain certain text in the html file. We can use “.tagname” to get the info belongs to that tag, “.parent” can return the parent tag of the current tag, etc.

“front-end.py” contains two test cases for front-end codes, specifically, “index.html”.

“test\_html\_text()” tests whether “index.html” contains the text we want. For example, we want the text of our project’s name to be shown on the page. If this test run without error message, then the text is included in the web page.

“test\_html\_parent()” tests whether some tags are the parent of some other tags. For example, we may want all the tags are inside <div> tags, so that all elements are divided clearly by <div>. The test can do it.

**Instructions for “front-end.py”:**

1. **In order to run tests, you need to install python on your pc.** [**https://www.python.org/**](https://www.python.org/)
2. **An ide is recommended (e.g. Pycharm)**
3. **Need to install “BeautifulSoup” package(“**pip install beautifulsoup4” or install on your own ide)
4. **Run the file on ide, if no error message prompts in the console, the tests are passed.**