**Testing:**

Tests have been made on both servers and tests. in the server part, our models are all tested. Since data in Role model and Good model are all added beforehand, Role model is tested with User model and Good model is only test with the operation add. User model has been tested with generating a correct password hash, correct reset and change tokens, duplicate email change, administrator and user roles connected with Role model and gravatar. Order model has been tested with operations add and query the good in orders. Cart and Add models are tested together and some functions in cart is called to test the add a product, remove a product and update a product. All tests are passed proving that our database is established correctly.

Selenium is used to test an end-to-end test. An administrator account will be created for testing. When we use selenium to test, instuctions are made to web browser and never interacted with application. Instructions are similar to the mouse and keyboard operations. Find\_element\_by\_lin\_text() can find a linkage and click it to jump to another page. Find\_element\_by can search the elemnt in HTML pages. In this approach, the process of discovering a product and then adding it to a cart and checking out the order can be simulated to test.

****

**Logging:**

We use those web application logs for at least 3 things: Detect attacks. Detect application misuse. Detect errors (loss of availability). By the way, Logging is really an important function since the performance of application will degrade with time because the data in the database are largely increased. In this case, logging needs to be applied so that customers can know that the application is still working and not breaks down.

In the implementation, the logging includes different levels of severity and useful messages which is saved is designed. When user logs in and logs out, the logging will give a message and when user enter wrong password, the logging will detect errors and give out warning.

