We utilized the unittests framework in Python to write unit tests for the following files: server.py, client.py, backend.py, frontend.py, and dashboard.py. We didn’t directly interact/utilize sockets in frontend.py, so we didn’t need to employ @patch decorators. In all of our files we tested our methods to make sure that they were returning expected values.

Clienttest.py

- Test if we can successfully connect to the server
- Test if we can successfully parse log entries

Servertest.py

- Test if log entry is written correctly
- Test if we can successfully create a user

Backend.py

- See if we can remove csv file successfully
- Test if we can successfully read the csv file
- Test if we can successfully get the auth level of the user
- Test if we can successfully authenticate
- Test failed authentication

Frontend.py

- Tests if the appropriate menus are being displayed based on the role of the user

Dashboard.py

- Test if we can receive a photo successfully
- Test if we can close connection properly

For our last sprint, we will be sure to do a code coverage analysis for a more thorough check of our code.