**Project Acceptance Tests**

* **Visualising and graphing two degrees of separation on the network mapper**
  + Testing the visualisation of two degrees of separation from a researcher will involve front end and back end testing.
  + The input for the test will be a researcher’s name and the expected output will be a visualisation of the researchers that they have worked with in the past, out to two degrees of separation.
  + The front end testing will be done visually, without the need for any test suite.
  + The back end testing will use some sort of testing framework such as J-unit or the built in module from python ‘unittest’.
  + We will know that the test has passed if the visualiser works as intended and displays the correct researchers associated with a particular person and their work.
* **Adding and Removing members of a core research team**
  + Testing the ability to add or remove members of the core research team will involve front end and back end testing. Upon successful completion of the test, we should be able to successfully add a team member to the database as well as remove a team member from our database.
  + The input for the test will be the name of a researcher and the output should be an entry or a removal of a row into the chosen database.
  + The test can be completed using a testing frame work for the backend, such as unittest.
  + The frontend will be tested manually by visually checking that there is a place to enter or remove names as well as a button to submit the name to the backend.
* **Visualising and graphing three degrees of separation on the network mapper**
  + Testing the visualisation of three degrees of separation from a researcher will involve both front end and back end testing.
  + The input for the test will be a researcher’s name and the expected output will be a visualisation of the researchers that they have worked with in the past, out to three degrees of separation.
  + The front end testing will be done visually, without the need for any test suite.
  + The back end testing will use some sort of testing framework such as J-unit or the built in module from python ‘unittest’.
  + We will know that the test has passed if the visualiser works as intended and displays the correct researchers associated with a particular person and their work.
* **Filtering by specific researchers and publications**
  + Testing the filtering function will involve both front end and back end testing.
  + Testing the back end will involve using a test suit such as J-unit or the builtin module from python ‘unittest’.
  + Testing the front end can be done manually and visually by checking the correct layout is displayed and forms are submitted appropriately.
  + The tests will pass if the front end elements work correctly and are able to send data to the back end, on top of this, the back end tests will pass if the correct data is filtered out.
  + The inputs for the test will be a researcher name or publication and the output will be the visualisation of the network map.