TEST CASE DOCUMENTATION

For this milestone, we have a single test case against which we test our peer to peer network.

TEST CASE 1 :

There are 4 components in total (You can refer to the design doc for more details about these servers).

1. Catalog Server
2. Order Server
3. Front End server
4. Client (This calls into the frontend server)

As a part of this test-case, we deploy 3 servers and 1 client on 4 different ports of the same machine.

There are 3 methods which the client can use to call into the front-end server.

1. Search
2. Lookup
3. Buy

<u>TestFlow</u>

1. As the diagram shows below, the client calls the frontend server API ‘search’ with a book topic passed as a parameter.
2. In the response, the client gets all the books having the topic passed above in the search API.
3. Following this, we select a book randomly and send the id of that book as an argument to the lookup API of frontend.
4. The lookup API retrieves the details of the book corresponding to that id.
5. The book details contains the number of books present in the book store currently. If the count is greater than zero, client goes on to buy the particular book.
6. Client calls the buy API of frontend with the same id, which results into the decrement of the count of the said book in the catalog database by 1.
7. We repeat this entire process n times where n is a parameter which we are passing to our script in client.py. (Default value of n is 5).

You can refer to the diagram in the design document to gain an idea of the overall architecture and it’s working.