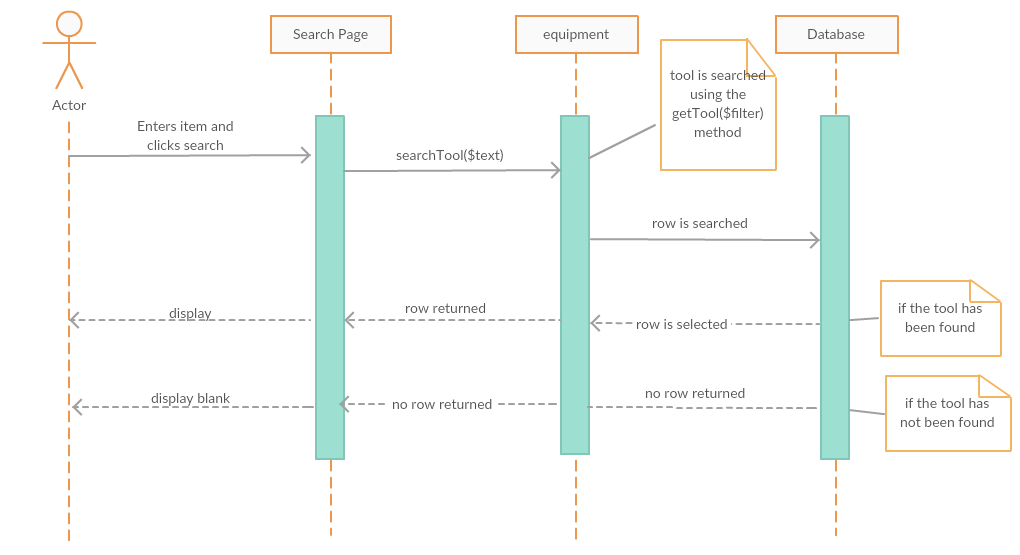
**Requirement Specification: searchTools implementation**

My requirement was to implement a search functionality, which could be used by both the administrator and the students, when searching for a tool. The students can use the search functionality to search for equipment when trying to reserve, loan or view the tools in the lab inventory.

Sequence Diagram



Unit Test: **getTools**

|  |  |  |
| --- | --- | --- |
| **Unit Test Point** | **Testing** | **Result** |
| SQL statement to get Tools | **Write an SQL statement with test inputs and run it using phpMyAdmin.**  ***Test input:*** “SELECT Equip\_ID, Equip\_Name, Equip\_Description, Equip\_Status, Equip\_Category, Equip\_Price, Equip\_Manufacturer, Lab\_Id, Supplier\_Id FROM equipment”  ***Expected result:***  Query is successfully run and result is successful.  Query returns rows of results. | Rows of the information related to the equipment is returned in rows. |
| getTools method in equipment class | **Write code to create an object of the class and call the method with test inputs and compare the output in database.**  $obj = new equipment();  $filter = false;  getTools($filter=false);  this->query($strQuery);  ***Expected Result:***  Information relating to the Tools/Equipment existing in the database should be returned. | The existing equipment in the database are successfully returned. |
| getTools method in equipment class | **Write a PHP unit test to test the code**  ***Input:*** none  ***Output:*** Test passed. 1 Test, 1 Assertion  ***Assert condition:*** $this->assertEquals(true, $obj->getTools(); | Unit testing is successful. |

Unit Test: **searchTool**

|  |  |  |
| --- | --- | --- |
| **Unit Test Point** | **Testing** | **Result** |
| SQL statement to get Tools | **Write an SQL statement with test inputs and run it using phpMyAdmin.**  ***Test Input:*** SELECT \* WHERE Equip\_ID like ‘%Volt%’ or Equip\_Name like ‘%Volt%’ or Equip\_Description like ‘Volt%’ or Equip\_Status like ‘%Volt%’ or Equip\_Category like ‘%Volt%’ or Equip\_Price like ‘%Volt%’ or Equip\_Manufacturer like ‘%Volt%’ or Lab\_Id like ‘%Volt%’ or Supplier\_Id like ‘%Volt%’  ***Expected result:***  Information relating to the Tool/Equipment being searched for should be successfully returned/found. | A row(s)containing the text searched for in the database, is returned. |
| searchTool method in equipment class | **Write code to create an object of the class and call the method with test inputs and compare the output in database.**  $obj = new equipment();  searchTool(Volt);  this->getTools($filter);  ***Expected Result:***  The tool in the database is returned/found if it exists in the database | A row(s)containing the text searched for in the database, is returned.  searchTool(‘Volt’) |
| searchTool method in equipment class | **Write a PHP unit test to test the code**  ***Input:*** ‘Volt’  ***Output:*** Test passed. 1 Test, 1 Assertion  ***Assert condition:*** $this->assertEquals(true, $obj->searchTool($text); | Unit testing is successful. |

Professional Test Result (Norbet Sackey)

The search functionality has been well implemented, taking into consideration all the possible searches a user may make, with regards to searching for tools in the lab inventory. The search implementation takes into consideration the equipment ID, equipment name, equipment description, equipment status, equipment category, equipment price, and equipment manufacturer. These are all properties of a tool, which a user may want to use in order to search for a tool in the lab inventory. This takes care of the case whereby a user may not know specific information about a tool (for example, the tool’s name), but the user would be able to search the same tool using information that may be associated with the tool (equipment description, equipment status, equipment category, equipment price, and equipment manufacturer).

In addition, the search implementation supports both administrator searches, as well as student searches. This implementation could be used on the administrator’s login system, or the student’s login system, as the search implementation contains generic information for any tool. Despite its successes, the search implementation may need to cover more columns in the database as there is more information that a user may expect to be associated with a tool in the lab inventory.