# Implementation 1 Report

## Group 14 Members

* Kwame Odame
* Norbert Sackey
* Makani Mwembe
* Margaret Ayodele Morenike
* Benedicta Amo-Bempah
* Naa Korkoi Larmie.

## Requirement Specification:

1. A user (student/admin) should be able to log in with their user id and password.
2. A student should be able to reserve a lab equipment that is available as well as unreserve a previously reserved equipment.
3. An administrator should be able to add new equipment to the lab inventory database.
4. A user should be able to view all tools available in the lab inventory database.
5. A user should be able to search for a tool using any tool within the lab inventory database using a keyword/search pattern for the query.
6. An admin should be able to check-out and check-in an equipment for a student.

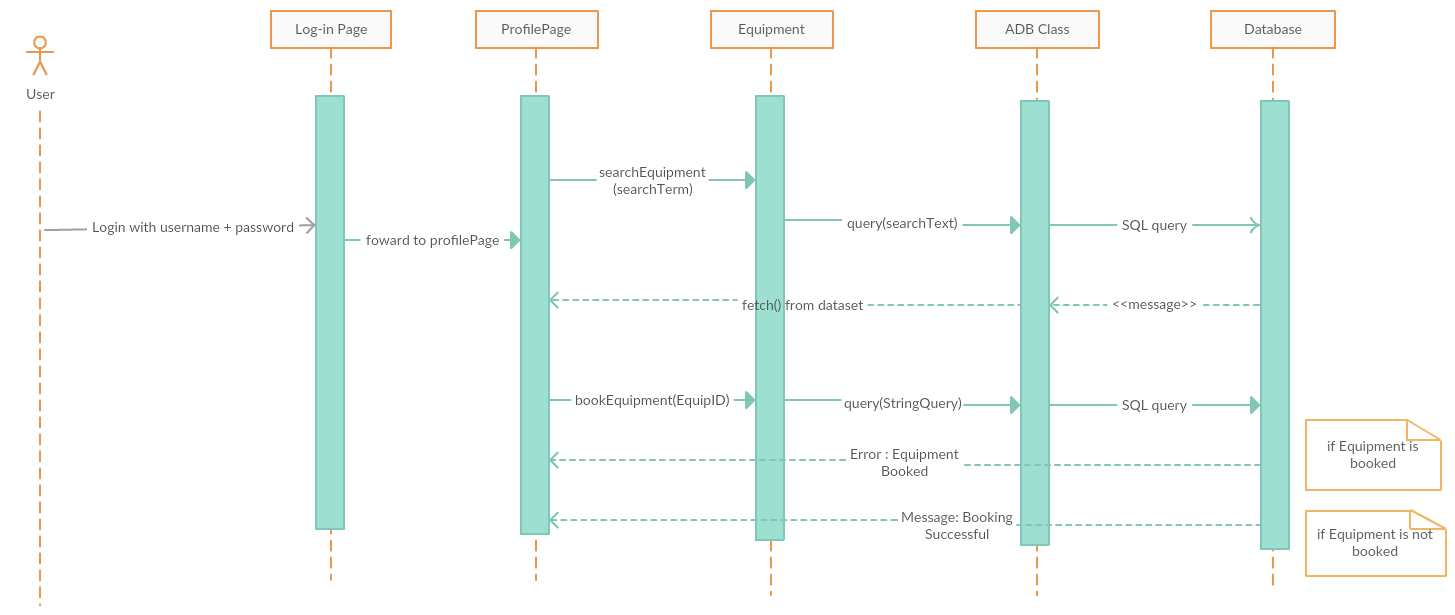
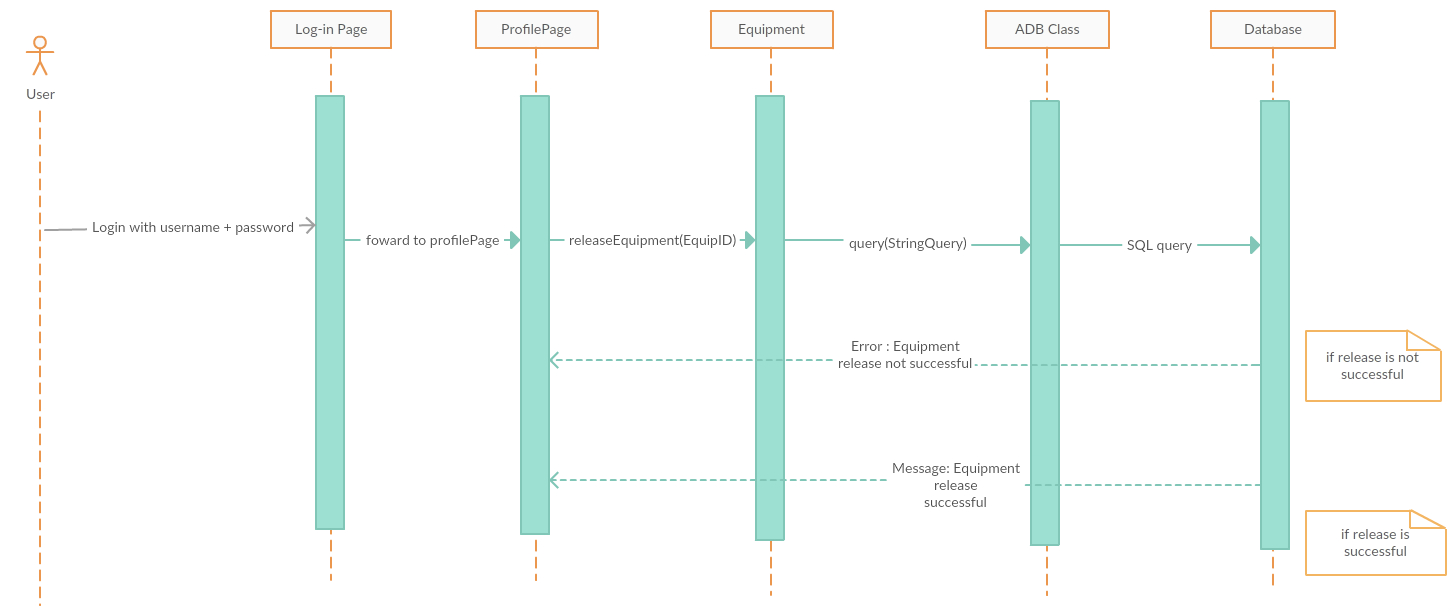


Fig 1. Sequence diagram for Reserve an Equipment

Fig2. Sequence diagram for Unreserve/Release a booked equipment

**Unit Testing for addBooking();**

|  |  |  |
| --- | --- | --- |
| Unit Test Point | Testing | Result |
| mySQL statement to add a booking | Test input:  SQL values Equipment ID(int), User ID(int)  Expected Result: 1 new row should be added to bookings table. | New booking was added to database with the given Equipment ID, User ID and a timestamp of when the booking occurred. |
| addBooking method in  Booking class | Test input: Equipment ID(int), User ID(int)  Expected Result:  New booking record should be added to database | New booking was added to database with the given variables by the addBooking($Equip\_ID, $User\_ID) method along with a timestamp of when the booking occurred. |
| addBooking method in  Booking class | Input: Equipment ID(int), User ID(int)  Output: Test passed.  1 Test 1 Assertion.  Assert Condition :  $this->assertEquals(true, $obj->addBooking($EquipID, $studID)) |  |

**Unit Test for bookEquipment method**

|  |  |  |
| --- | --- | --- |
| Unit Test Point | Testing | Result |
| mySQL statement to book an equipment | Test input:  SQL values Equipment ID(int), User ID(int)  Expected Result: Equipment status should be updated to ‘reserved’ | Equipment Status with given Equipment ID  was updated to ‘reserved’. A new record in bookings table was also added with the given Equipment ID and User ID |
| bookEquipment method of  Equipment class | Test input: Equipment ID(int), User ID(int)  Expected Result:  Equipment status should be changed to ‘reserved’  New booking row should be added | Equipment Status with given Equipment ID  was updated to ‘reserved’ and a new booking was added using  bookEquipment($Equip\_ID, $User\_ID) method |
| bookEquipment method in  Equipment class | Input: Equipment ID(int), User ID(int)  Output: Test passed. 1 Test 1 Assertion.  Assert Condition :  $this->assertEquals(true, $obj->bookEquipment($EquipID, $studID)) |  |

**Unit Test for unbookEquipment method**

|  |  |  |
| --- | --- | --- |
| Unit Test Point | Testing | Result |
| mySQL statement to unbook an equipment | Test input:  SQL values Equipment ID(int), User ID(int)  Expected Result: Equipment status should be updated to ‘available’ | Equipment Status with given Equipment ID  was updated to ‘available’. Record in bookings table corresponding with given Equipment ID and User ID was deleted. |
| unbookEquipment method of  Equipment class | Test input: Equipment ID(int), User ID(int)  Expected Result:  Equipment status should be changed to ‘available’  Booking record with given Equipment ID and User ID should be deleted | Equipment Status with given Equipment ID  was updated to ‘available’ and specified booking record was deleted using the  unbookEquipment($Equip\_ID, $User\_ID) method |
| unbookEquipment method in  Equipment class | Input: Equipment ID(int), User ID(int)  Output: Test passed. 1 Test 1 Assertion.  Assert Condition :  $this->assertEquals(true, $obj->bookEquipment($EquipID, $studID)) |  |