



Report

Assignment 3 - *Testing*



12 Feb, 2017

Semester: Spring 2017
Course: Software Technology
Author: Sarpreet Singh Buttar

Contents

Task 1 Test Plan	1
Reflection	1
Subtask 2 Test Cases	1
Task 2 Test Cases	1
Reflection	1
Task 3 Unit Testing	1
Reflection	1
Task 4 API Tests	2
Reflection	2

Task 1 Test Plan

Reflection

In the beginning I was bit confused regarding the design of test plan such how the structure should look like and how will I figure out the requirements because I have not implemented all of them. Somehow, I decided to follow the instruction given in the Task 1 and made a simple plan with all the required sections. In addition, I would like to add test cases in the test plan but I could not because under the submission heading, test cases is written separately not along the test plan and I assumed that I should submit them in a separate document. Other than this I must say its time consuming task.

Subtask 2 Test Cases

Reflection

It was not hard for me to create test cases from the use cases. I have used all the alternative paths as a separate test case. In addition, I also covered server response code which I have not mentioned in the use case cases flow chart. Returning responses was bit challenging and I made one assumption in this. If everything is correct and no internal error occurs then I return 200 OK otherwise 404 Not found. Returning 404 was not correct technically if internal error occur such as problem in reading or writing XML file but because API does not mention 500 so I return 404 on every error.

Task 3 Unit Testing

Reflection

This was the most fun part and as well as time consuming. In this part I made unit tests for all the implementation I have done so far which means use cases comes automatically in it. However, I have not clearly mentioned that these tests belongs to use case 1 or 2 but the name of the tests explain what they are doing. In addition, I mocked all the dependencies. For this I have used mockito and I suggest you to add this dependency in the 'build.gradle' file for running my tests. Below picture shows how it should look.

```
!
! dependencies {
!     compile 'io.dropwizard:dropwizard-core:0.8.0'
!     compile 'io.dropwizard:dropwizard-assets:0.8.0'
!     compile 'org.codehaus.jackson:jackson-mapper-asl:1.9.13'
!     testCompile 'junit:junit:4.11'
!     testCompile 'io.dropwizard:dropwizard-testing:0.8.0'
!     testCompile 'org.easytesting:fest-assert:1.4'
!     testCompile 'org.mockito:mockito-all:1.9.5'
! }
! |
```

While creating test for 'booksDAO' class I faced one problem. In this class 'JAXBContext' is a static interface and when I tried to mock it I got exception all the times. I am not sure why but in the exception it was mentioned that 'cannot mock static interfaces'. So then I decided to test this class with some dependencies.

Task 4 API Tests

Reflection

This is the first time I tested API by creating the automative tests. It was very nice experience for me to learn this. While implementing I was confused regarding how to show 404 because it is mentioned in the API. So, I assumed if anything goes wrong such as requested book is not available, cannot read or write the XML file etc happens then I will present 404. However, it is not technically correct because these types of error produce 500 internal server error but because this is not required so I decided to use 404 instead. Also in the 'APITests' package all the resources which are not implemented yet throws 'NotImplementedException' and I will implement them in next iteration. Other than this all the testing went very well.