# Iteration 1

**Milestones**

|  |  |
| --- | --- |
| Milestone | Date |
| Starting the iteration | 2018-03-16 |
| Ending iteration | 2018-03-17 |

**Objectives**

1. Enable the users to add a book.

**Tasks**

|  |  |
| --- | --- |
| Description | Estimated time |
| Plan the iteration | 1h |
| Use case documentation for add book | 30 mins |
| Create an activity diagram for the add book use case | 1h |
| Writing manual test cases | 1h |
| Implement add book | 2h |
| Implement unit test | 1h |
| Implement API tests for add book | 2h |
| Update test plan | 1h |

**Evaluation criteria**

1. The user is able to add a book.
2. All tests pass.

# Reflections

I choose to just create a use case documentation and an activity diagram to be able to get a good understanding of the expected functionality. I did not, however, see the need for a sequence diagram for this implementation since I had a quite good understanding of how to implement it and wanted to do it using a test first approach. TFD worked quite well in this case. I wrote unit tests to verify that the helper functions worked as they are supposed to before combining everything. The main reason to why the implementation took as long as it did was because my previous implementation of the API tests had some bugs which could sometimes affect the XML depending on the order the tests were exercised. This meant some debugging to understand and fixing the problem temporarily to be able to continue with the implementation. During the API test phase, I decided to rewrite my implementation of how the API tests were executed. This meant that it took more time to implement the API test for Add book but hopefully it will reduce the time spent on implementing upcoming features and tests.