Software Testing Assignment 1

Mathis Neumann, student number 605304 (UTU) and 72043 (Åbo)

Part 1

Requirement Errors
Requirement 2 and 12 are contradictory, therefore change 12 to:
12. If the user tries to set a negative base price, the base price should be set to one.

In addition to that it is is set that the sell price should not be possible to be negative, but it is not said what should happen. I decided that the lowest price can be 0, so a free book. Also it is not very realistic that VAT can be more 100%, but that I did not fix.

The coverage with all tests is 100% for both lines and branches.

The following requirements where not properly implemented:

1: Base price constraint (>= 1, see above) is failing in 50% because the constructor of the book class does not use the constraints
2: The base price can be set to a negative value
4: the VAT can be negative
6: discount can be negative
8: the final price requirement is not met in jFeature, because I set it to be depended to other tests (like base price) which fail. The overall calculation seems to be correct, though.
9: like 8, but the price can also get negative because parts of the calculation (like the base price) can get negative.
12: Base price does not have a minimal value of 1
13: VAT min value is not 0
14: discount min value is not 0

It is notable that only 8 and 9 use the test dependency feature of jFeature and also that there is are separate tests for the constructors which are not mapped to the requirements. The same constraints that are not properly implemented in the setters are also not implemented in the constructors.

After fixing the book class, the coverage was at first only 97% and 90% for branches, because it was previously checking for the base price > 0 inside the setSellPrice() method, which I then deleted, because it is already handled inside the setters.

Part 2

There are now 17 requirements that are covered by the tests, each new one has one additional test. One to make sure it is possible to set onSale to true and another for checking the bestseller-sale constraint for both directions.
Code coverage is still/again 100%, since I wrote a test for each new method, while old tests could stay the same, so the bestseller tests were not affected (see explanation above). The new tests were testSale() and testSaleNoBestseller().

It is important to note that the new requirements are not sure what happens to the other discounts. Since in the real world you often can't get an additional discount, I implemented it that way, that regular discounts are not possible if the book is on sale, this should be an additional requirement.