**Part1**

<https://github.com/pure1017/iDrop>

**Part2**

*2.1 As a user, I want to search a book in the website so that I can find the information of the book, like the category of the book, the URL to buy it online and the rate from other users.*

My condition of satisfaction is that I give the right book name and click the search button. And there is book related to the name in the API or our database.

*2.2 As a user, I can rate for the book which I have already read for more than one day so that I can express my opinion for the book.*

My condition of satisfaction is that I have read book and signed in the calendar for more than one day. And then I click the ‘want to rate’ button and input all the rate information, like rate for plot.

*2.3 As a user, I want to set and unset the books I like, and get the list of my favorite books.*

My condition of satisfaction is that I login and click my favorite button, and then I will get the list of books I liked, and I can set and unset them in the list.

This one is a new user story we added to the project, which is the basis of recommendation algorithm.

*2.4 As a user, I want to get the recommendation about the books which I am maybe interested in so that I can read new interesting books.*

My condition of satisfaction is that I have logged in the website and have already chosen the category I am interested in. Or I have searched some books in the website with an account.

Acceptance testing plan: A user first logs from our website using google login. The user types in the name of the book he wants to search (user story 2.1). The valid inputs are words, numbers, letters and space. The invalid inputs include special characters, and a string over 80 characters. Then the user will read the book. After reading, the user will rate the book (user story 2.2). The valid input is a number in the range of the score, but the invalid input is a number out of the range. After reading and rating, if the user likes the book, he can set it as favorite (user story 2.3). The valid inputs include a request to set a book which is not in the list, and unset a book in the list. The invalid input is a request to set a book already in the list. Now the user wants to get some recommendations (user story 2.4). The invalid input is a request of getting recommended books. The invalid input is a request but the user has no favorite books in the favorite list.

**Part3**

Link to test cases:

<https://github.com/pure1017/iDrop/tree/master/iDrop/src/test/java/unit>

Link to configuration of build tool, which is maven:

<https://github.com/pure1017/iDrop/blob/master/iDrop/pom.xml>

Link to frontend test:

<https://github.com/pure1017/iDrop/tree/master/iDrop/src/main/resources/public/assets/js_test>

The method login() in GoogleApiLogin.java cannot be tested in unit test, because according to Google Login API, I need a one-time code from frontend.

**Part4**

Link to the first version of style checker report:

<https://github.com/pure1017/iDrop/blob/master/iDrop/target/site/checkstyle%20-%20first.html>

Link to the cleaner version of style checker report:

<https://github.com/pure1017/iDrop/blob/master/iDrop/target/site/checkstyle.html>

Link to the first version of spotbug report:

<https://github.com/pure1017/iDrop/blob/master/iDrop/target/site/spotbugs-first.html>

Link to the Final version of spotbug report:

<https://github.com/pure1017/iDrop/blob/master/iDrop/target/site/spotbugs.html>