Test report

Table of contents

Version: 1.1
Date: 2012-04-16

Authors: Oscar Brodefors, Filip Askviken, Rikard Andersson, Emil Nyström

This version overrides all previous versions.

1 Introduction

1.1 Purpose of application

A simple application for smartphones where students can gather to sell and buy used books needed at courses given at Chalmers (and possibly other Universities). Users can register books for sale and other users can search and find these books. The unique selling point compared to existing sites like blocket or tradera is the nische market. When buying from our application you know that the seller is close by and that you can set up a meeting to get the book the same day.

1.2 General characteristics of application

No transactions are handled by the application. The application simply matches buyer with seller and provides the opportunity for market mechanisms to function in the interaction between these two parties.

Functionality to add a plethora of attributes about the book and for users to search for books on these criterias. Also the ability to describe a book in general and to upload a photo of the book taken with the mobile camera. A book could be uploaded by filling in a ISBN-number and then obtaining book title, author and edition.

The ability for other users to comment on the book.

Possible functions to be added in future versions:
- Auctions
- Connection to Cremona to compare price and to buy new books
- A website connected to the application

2 Test environment

This android-project is preferably tested and runned from Eclipse. First step for a new tester or developer should be to install EGit from Eclipse Marketplace and the ADT plugin for Eclipse (http://developer.android.com/sdk/eclipse-adt.html.). Please visit and read the installation guide (http://developer.android.com/sdk/installing.html).

After installation of Android ADT and connection with Github in your Eclipse, import the project in Eclipse: file → import → Git → Projects from Git → next → URL → insert https://emil-nystrom@github.com/oscarbr/Brainiacs-startup.git as URL → next → next → next → finish. Now you are ready to build or run the application.

The test report document to new developers and testers is avaliable on: https://docs.google.com/document/d/1oXC8qSHJzy3TRIEg0nVonFR0eNtPC01yBdP7oxgWOlc/edit.
The test report table is also avaliable at: https://docs.google.com/spreadsheet/ccc?key=0Ao75Xu-pWZuDdGxoc1ZROF9BZXlEUThuUnJ5UUl1OHc.

Vilka förberedelser behöver göras innan man kan testa
*** Var finns koden eller .apk?*
*** Hur installeras denna?*
*** Behöver man tömma eller skapa en databas?*
*** Behövs internetuppkoppling (kanske ett speciellt nätverk)?*
*** …*

2.1 Hardware environment

For the tests, three MacBook's and one Dell XPS M1530 have been used. Three computers with the operating software Mac OSX-Lion and one with Windows 8 have been used. A android emulator in Eclipse (Target Android 2.1, skin WVGA800) and a real mobile phone (Samsung Galaxy S plus) have been used for the tests.

(what computers/devices was/wered used?)
Använder ni emulatorn eller riktig telefon vid dessa?

2.2 Software environment

For every test, the following software have been used:

- Eclipse SDK version 3.7.2
- Android Development Toolkit version 18.0.0
- Eclipse Mylyn project version 3.6.5
- Eclipse EGit 1.3.0

(what software (incl version) was used?)

2.2.3 Softwares

2.2.3 Software settings

3 System information

3.1 System version

(what version of your software is tested in this document)

4 Known bugs and limitations

(list all bugs you knew of before the test)

5 Test specification

(list the name and version of the test documentation (doc with test cases))

Test ID 1.1 **The seller can register a book for sale**

Positive test: The tester is uploading a new book with the following information: (Robinson Crusoe, Daniel Defoe, 2nd edition, literature, excellent quality, 300 SEK, telephone: 1234567, password: muffins. The book should then be uploaded to the market. The tester then tries to update/change the price to 100 SEK by pressing the edit-button and use his password.
Negative test: The tester is uploading a new book with the following information: (, Daniel Defoe, 2nd edition, literature, excellent quality, , telephone: 1234567, password: muffins. The tester should get a error message saying "missing title and price".

Test ID 1.1.1 **The seller can set a price for the book**

Positive test: The tester uploads books with following testnumbers: 99 SEK, 300 SEK, 1000 SEK, 0 SEK
Negative test: The tester uploads books with following testnumbers: hundra kronor, 133700 SEK and should get a error message stating: "Invalid price, please write a number between 0-10000 SEK.

Test ID 1.1.2 **The seller can upload a picture of the book for sale**

Positive test: The tester takes a photo of a book and upload it together with the book.
Negative test: N/A

Test ID 1.1.3 **The seller has to set a title for the book**

Positive test: The tester should write "Robinson Crusoe" as title and upload a book.
Negative test: The tester should write the title "xxyy" and get a exception message "Title must be greater than 10 characters".

Test ID 1.1.4 **The seller can set an ISBN for the book for sale**

Positive test: The tester should fill in "9789144060743" as ISBN (Java direkt med Swing.
Negative test: The tester should fill in 123456 as a ISBN-number and get a exception message: "The ISBN-number must be 10 or 13 digits long."

Test ID 1.1.5 **The seller can set a publishing year for the book for sale.**

Positive test: The tester should fill in 1999 as publishing year.
Negative test: The tester should fill in 2015 as a publishing year and get a exception message: "this is not a valid year".

Test ID 1.1.6 **The seller can set a course for which he/she used the book.**

Positive test: The tester should fill in "Databases" as Course and "TDA357" as a course.
Negative test: N/A

Test ID 1.1.7 **The seller can add a comment about the book.**

Positive test: The tester should fill in "excellent quality" as a comment.
Negative test: N/A

Test ID 1.2 **The seller can update information about a book or take a book off**

the market

Positive test: The tester is uploading a new book with the following information: (Robinson Crusoe, Daniel Defoe, 2nd edition, literature, excellent quality, 300 SEK, telephone: 1234567, password: muffins. The book should then be uploaded to the market. The tester then tries to delete the ad by pressing the edit-button and use the password.The book should be deleted from the market.
Negative test: The tester should fill in a wrong password and the not be able to delete the book from the market. The book will automatically disappear from the market after two months.

Test ID 2.1.1 **In a search, a buyer can specify a book's title**

Positive test: The tester is searching for the title "Robinson Crusoe" and then get a match.
Negative test: The tester should fill in "Petter och hans fyra getter" as a title and not get a match in the market with the following message: "your search didn't match any books at the market".

Test ID 2.1.2 **In a search, a buyer can specify a book's author**

Positive test: The tester is searching for the author "Daniel Defoe" and get a match.
Negative test: The tester should fill in "Emil Nyström" as a author and not get a match in the market with the following message: "your search didn't match any books at the market".

Test ID 2.1.3 **In a search, a buyer can specify which course the book should be used in**

Positive test: The tester is searching for the course "objektorienterad programmering" and then get a match on the book "Java direkt med Swing". The tester searches for the course "TMV215" (linear algebra) and get a match on the book "Linear algebra and its applications".
Negative test: The tester should fill in "Harry Potter Knowledge" as a course and not get a match in the market with the following message: "your search didn't match any books at the market".

Test ID 2.1.4 **A search can be sorted on any of the available attributes that a book can have. Originally it is sorted by price.**

Positive test: The tester is searching for the title "Robinson Crusoe" and then get three books with the cheapest first.
Negative test: N/A

Test ID 3.1 **A book can be uploaded by ISBN-number**

Positive test: The tester should fill in 9789144060743 as ISBN (Java direkt med Swing and the get the title: "Java direkt med Swing", the author "Jan Skansholm", and the edition "6".
Negative test: The tester should fill in "1234567" as ISBN and get a error message saying: "Not a valid ISBN-number".

Test ID NF1 **Usability**

Positive test: The tester should ask someone who hasn't used the application before and ask them to add a book. and take time of the process and ask questions afterwards, in order to understand if the application is considered easy to understand. Read more under headline 7 regarding black box testing.

Test ID NF2 **Reliability**

Positive test: The tester will add books at the same time as three other testers. This small test will indicate if the server can handle four uploading books at the same time.

Test ID NF3 **Performance**
Positive test: The tester should view information about a book and then turn of the internet connection the the phone. The tester should then open the application and still see the information on the book viewed last time.

Test ID NF5 **Implementation**
Positive test: After every release, the tester should control that the client-side application cannot exceed 10 MB.

6 Automatic test

6.1 Code coverage

Använder ni kodtäckningsverktyg, vilken grad av kodtäckning siktar ni på? – motivera

6.2 Nightly builds

Om ni gör detta, beskriv hur, var, när detta görs

6.3 Unit test

Vilka unit test-programvaror använder ni
Finns det saker som ni inte kan testa via unit tests – dokumentera och motivera detta

7. Black box testing

Black box testing is a method of software testing that tests functionality of an application without considering the internal structures or workings. The testers in the black box test will only consider the functionality and ease of use of the application, and nothing regarding the actual code. A black box test will be done together with three potential users of the application who never have seen the application before. The testers will be adding a book and do a search of a book and give comments on the application. Information from this black box test will be used in the non functional tests listed under headline five above, named NFX.

8 Test report (kan vara bilaga/bilagor)

Vad testas? Är det senaste versionen i GIT (inte bra!), eller en upptagged release (bra!).
(present a table with Test id, Result, Comment)
(use comment to say what bug the test resulted in or that the test could not be performed since the requirement is not yet implemented)