Homework Lesson 6

Angelika Kryvoruchenko

Level 1

1. Make a comparative table

|  | **What is checked** | **When applicable** | **Restrictions** | **Peculiarities** |
| --- | --- | --- | --- | --- |
| **Functional** | Functions that the system should perform | is performed best when you have a limited scope. In other words, if you want to test a specific feature or an individual piece of your application, functionality testing is the way to go. | Generally it's testing a feature in isolation so there is no full end to end testing done. It can happen that testing will not expose user end to end defects | Sometimes defects will be very minor and not severe. There could be many defects found like this that do not affect the software in a critical way but should be fixed in the future |
| **Non-functional** | Characteristics of systems, such as usability, performance efficiency or security | used to evaluate a software application's performance, usability, dependability, and other non-functional characteristics | Usually a separate system or environment will be required in order that the non functional testing is not impacted by other users on the system and gives accurate result | In order to run a successful non functional test cycle you should understand intended user behavior on your application in order to create targeted tests |
| **Change- related** | system functionality | After changes in the system ( defects fix, adding a new functionality) | It must be run in order to verify existing functionality is still working along with the changed or fixed code | When a change or fix is completed testers should test in the area of defect. It is often the case that a fix in one area will have a connection to another area in code |

1. Explain the difference between regression and retesting

Retesting is done to check that the initial bug which was found and fixed is working as it should, while regression testing is used to sweep the application for defects that may have arisen from the change or other unknown residual bugs.

Regression testing is performed for passed test cases, while Retesting is done only for failed test cases.

Regression testing checks for unexpected side-effects, while Re-testing makes sure that the original fault has been corrected.

Regression testing doesn’t include defect verification, whereas Re-testing includes defect verification.

Regression testing is known as generic testing, whereas Re-testing is planned testing.

Regression testing is possible with the use of automation, whereas Re-testing is not possible with automation.

Level 2

1. Do you think it is possible to perform only functional testing for a product without checking non-functional requirements?
   * + If yes - in what cases?
     + If not, why not?

Support your answer.

Functional testing of a system involves tests that evaluate functions that the system should perform. The functions are “ what “ the system should do. So theoretically the system will work, because all critical functionalities were tested. The application will be usable and working. Then based on user interaction we can design non functional testing.

2.How do you see the need for smoke testing? Is it always appropriate?

Smoke Testing is performed to ascertain that the critical functionalities of the program are working fine. It doesn't delve into finer details. Often, smoke testing is known as build verification or confidence testing. It refers to a similarly fundamental form of testing where a build passes the test if it doesn’t “catch fire” on the first turn-on — hence the term smoke. It also can be used by developers to run when they check in any new piece of code. They can check if everything still works as expected after their code is checked in. Any issues found in smoke testing should be actioned immediately by developers and test teams. Smoke testing is only appropriate to check critical functionality.

Level 3

1. You are the founder of a startup planning to launch a mobile application for sharing cat photos on iOS and Android devices.  
     
   Users can upload photos of cats but they cannot upload photos of other animals/people/objects. Users can add friends and leave likes and comments.   
     
   Write 5 functional test cases that would test the application.

Test Case 1 : Upload cats photos

Pre Requirement

The user should be logged in

Steps: Expected Results:

1. Click on My Account button 1. My account page opens
2. Click on Plus button 2. Allow Catpics to access your photos window opens
3. Click Accept button 3. Phone Gallery opens
4. Click on the chosen picture
5. Click Next
6. Click Share 6. Picture was uploaded

Test Case 2 : Upload dogs photo

Pre Requirement

The user should be logged in

Steps: Expected Results:

1. Click on My Account button 1. My account page opens
2. Click on Plus button 2. Allow Catpics to access your photos window opens
3. Click Accept button 3. Phone Gallery opens
4. Click on the chosen picture
5. Click Next
6. Click Share 6. Error message “ Not acceptable content “ appears

Test Case 3 : Add friends

Pre Requirement

The user should be logged in

Steps: Expected Results:

1. Click on the Avatar icon 1. Other users Profile Page opens
2. Click “ Follow” 2. Button “ Follow” changes to button “ Following”
3. Go to your Profile Page 3. Users Profile Page opens
4. Click “ Following” 4. The New Following Account appeared on the list

Test Case 4 : Leave likes

Pre Requirement

The user should be logged in

Steps: Expected Results:

1. Click on “ Home” Button 1. Catpics homepage opens
2. Click “ Heart” under the picture 2 “ Heart” button turned red

Test Case 5 : Leave Comments

Pre Requirement

The user should be logged in

Steps: Expected Results:

1. Click on “ Home” Button 1. Catpics homepage opens
2. Click “ Add a comment” 2. Comment window opens
3. Write “ Wow! Love your cat”
4. Click “ Post” 4. Post was added

2. Write what non-functional requirements you would like to apply to your startup's product.   
Describe the tests that would check them (3-5 examples).

Non functional Requirements:

1. Page load should take no longer than 2 seconds
2. Each page has clearly laid out user interface
3. The site should be able to handle a maximum of 100 users logged in at the same time
4. A user should be able to navigate the site with keyboard only
5. The site should be able to stay functional if 50 users are using the site over 2 hours

Test Case: Page Load

Steps: Expected result:

1. A registered user logs into site 1. The homepage shows up in less than 2 seconds

Test Case: User Interface

Steps: Expected result:

1. A register user logs into site
2. A user evaluates the homepage layout. 2. The homepage which contains and textboxes are all correctly spaced.

Test Case: 100 Users Load Test

Steps: Expected result:

1. The test scenario is set up with

a hundred users logged into site

1. Each user will perform a set of steps 2. No errors are seen in the application

Such as uploading photos, clicking buttons,

inputting text

Test Case: Keyboard Accesability :

Steps: Expected result:

1. A registered user logs into site
2. The user can use a tab key 2. Tab keys should enable movement in site

to navigate through buttons and

text boxes

Test Case: Soak Testing

Steps: Expected result:

1. The test scenario is set up with

50 users logged into site

1. Each user performs various actions 2. There are no errors or crashes

over 2 hour period