**Author:** *Alessandro Ferro*

**ICTSAS519**

**Task 1**

Question 1: Choose one automated test tool for website and describe the key features and processes.

**Answer**

**LEAPWORK**

* It uses a flowchart model with a drag-and-drop interface to build test automation, without the need to write code. It has a quick learnig curve.
* Allows to manage multiple automation flows at once
* Captures processes in real time and transform them in automated flows.
* Manage large volumes of data from different data sources
* Has built-in Sauce Labs and BrowserStack to cover any technology.
* Can parallely run automation flows in multiple environments at once.
* Offer a visual scheduler to plan execute and automate the testing.
* Provide reporting functionalities with a video recording of the run, a debug version of the design canvas and a log with the debug information.
* Has a Public REST API to allow for integration in any project in any language.

Question 2: Describe the typical procedure for one of the testing and acceptance.

**Answer**

**Security Testing**

Considering how many websites need to handle sensitive data (eg e-commerce), it is vital to make sure that those data are safe from security issues.

There are two ways to test for a dynamic website security and both of them should be used for better resutls.

1. Code Review. Asking a developer who hasn’t worked on the code to go over the work done to check for bugs and security issues. A list of the top ten security issues in web applications is provided by the Open Web Applicatin Security Project (OWASP). Code reviews don’t need any external tool and can be performed in the normal development environment.
2. Pentesting. Setting up a replica of the production server and try to break into it. For this method of testing to be effective it is necessary for the penetration tester to have a deep knowledge of web security and “hacking” techniques. To maximise the chances of spotting problems, the pentester chould also have access to the source code.

Question 3: What are the Implications for organisational operations of system testing activities?

**Answer**

For small organisations, where budget limits may prevent the possibility to set up an isolated testing environment, testing could raise some issues. This is especially true for load testing and penetration testing, which could have an impact on other users.

On one hand, load testing could push the limits of the hardware, causing performance drops on the system or even preventing other users to access the server if the network gets flooded with traffic.

On the other hand, pentesting could trigger Network Intrusion Detection Systems, fill the log files with dodgy records or even damage the server or its configuration.

To avoid this kind of problems, the network administrator should always be made aware of which tests are to be performed and when, so that they can take the necessary measures to ensure the safety and the correct operativity of the system. This kind of tests are commonly planned outside of normal business hours to minimise the risk of affecting other people access to the server or network.

Each company will have different procedures to schedule and notify the administrators of the system, but the procedures always need to be properly documented to ensure that the policies are clear and can always be followed.

Question 4: Briefly describe the software life cycle in your own words.

**Answer**

There are plenty of different approaches to software development, but the two major ones, the ones that set the basis for any other in the modern day, are the waterfall model and the agile development.

The waterfall model is an extremely rigid approach, where development flows from one step to the next one. Each step must be completed in order to procede to the following one. Everything must be very well planned and delays or deviations could have dire consequences on the project. It is best suitable for small project, where the goal is perfectly known and understood by all the stakeholders, without any room for misinterpretation or ambiguity. The waterfall model doesn’t leave much flexibility for changes and on-the-run modifications.

The phases of the waterfall model are as follow:

* Requirement phase: The requirements are gathered from the stakeholders and are documented. In this phase is understood what the product should do.
* Design phase: The different options for the creation of the system are considered and a plan is defined to specify the system into details. In this phase is decided how the system will work.
* Implementation: The developer produce the code according to the specifications defined during the Design phase.
* Verification: The product is tested to ensure that the product does what is suspposed to di, that doesn’t what what is supposed not to do and that there are not bug or other issues.
* Maintenance: The product is monitored and maintained to ensure that everything keeps working. Some minor adjustment could also take place in this phase.

Agile development is an iterative model where the product is developed feature by feature in short sprints. The phases of each sprint are Requirement, Design, Implementation, Testing and Feedback. It’s a very flexible approach that relies on a close cooperation between developers and client to determine at the end of each sprint which direction the project should take. Such flexibility suits best larger projects that can be divided in features and where many changes from the initial concept can be expected.

Both approaches come with their own flaws and often a blend of the two is the ideal solution. Clients ideas aren’t always clear and well defined and this can lead to many problems that would eventually end in the project failure (or in the client not being satisfied). Starting a project with a well defined plan, as in the waterfall model, it’s helpful to bring everyone “on the same boat”, although at some point during development switching to agile would increase the chances of success.

Question 5: How to identify different users and modules?

It is important to identify how a website will be used by different people. Identify the different functionalities of a website, and how people will interact with it, allows to identify different user profiles that will us the website for different tasks.

This concept finds its application during usability testing, where different groups of users can be invited to test the website and its features giving a more thorough representation of how the website will be used.

Once the different types of users and how they will interact with the website are clear, it is necessary to identify every method to perform a task and verify if it works as expected without making assumption. This allows to test each functionality as a separate module, independently fom the others and accordingly to how it will be used and by who.

Question 6: Write three test cases for website.

**1.**

|  |  |
| --- | --- |
| **Test Suite ID** | TS001 |
| **Test Case ID** | TC001 |
| **Test Case Summary** | To verify that clicking the Login button without entering a valid username, output the appropriate error message |
| **Prerequisites** | 1. The user is registered |
| **Test Procedure** | Given a functioning username and password.   1. Enter a wrong username 2. Enter the correct password 3. Click Log In |
| **Test Data** | Username: alessandro.ferro  Password: Alessandro2020#  Wrong Username: allesandro.ferro |
| **Expected Result** | 1. A message “Invalid Login” should be displayed to the user on the same page. 2. The textboxes should clear to allow another attempt. |
| **Actual Result** | 1. The “Invalid Login” message is displayed to the user on the same page. 2. The textboxes are clear to receive a new input. |
| **Status** | Pass |
| **Remarks** | None |
| **Created By** | Alessandro Ferro |
| **Date of Creation** | 18/04/2020 |
| **Executed By** | Alessandro Ferro |
| **Date of Execution** | 18/04/2020 |
| **Test Environment** | * OS: Windows 10 * Browser: Chrome |

**2.**

|  |  |
| --- | --- |
| **Test Suite ID** | TS001 |
| **Test Case ID** | TC002 |
| **Test Case Summary** | To verify that the login procedure is case sensitive in relation to the password. |
| **Prerequisites** | The user is registered |
| **Test Procedure** | Given a functioning username and password.   1. Enter the username 2. Enter the password, but without capital letters 3. Click Log In |
| **Test Data** | 1. Username: alessandro.ferro 2. Password: Alessandro2020# 3. Test password: alessandro2020# |
| **Expected Result** | 1. A message “Invalid Login” should be displayed to the user on the same page. 2. The textboxes should clear to allow another attempt. |
| **Actual Result** | 1. The “Invalid Login” message is displayed to the user on the same page. 2. The textboxes are clear to receive a new input. |
| **Status** | Pass |
| **Remarks** | None |
| **Created By** | Alessandro Ferro |
| **Date of Creation** | 18/04/2020 |
| **Executed By** | Alessandro Ferro |
| **Date of Execution** | 18/04/2020 |
| **Test Environment** | * OS: Windows 10 * Browser: Chrome |

**3.**

|  |  |
| --- | --- |
| **Test Suite ID** | TS001 |
| **Test Case ID** | TC003 |
| **Test Case Summary** | To verify that clicking Log In without entering a password will output an error message and will allow another attempt. |
| **Prerequisites** | The user is registered |
| **Test Procedure** | Given a functioning username and password.   1. Enter the username 2. Leave the password field empty.   Click Log In |
| **Test Data** | 1. Username: alessandro.ferro 2. Password: Alessandro2020# 3. Test password: |
| **Expected Result** | 1. A message “Invalid Login” should be displayed to the user on the same page. 2. The textboxes should clear to allow another attempt. |
| **Actual Result** | 1. The “Invalid Login” message is displayed to the user on the same page. 2. The textboxes are clear to receive a new input. |
| **Status** | Pass |
| **Remarks** | None |
| **Created By** | Alessandro Ferro |
| **Date of Creation** | 18/04/2020 |
| **Executed By** | Alessandro Ferro |
| **Date of Execution** | 18/04/2020 |
| **Test Environment** | * OS: Windows 10 * Browser: Chrome |

Question 7: What is proofing content?

Proofing content is checking a website for grammatical errors and broken links. Grammatical errors include misspelt words, wrong punctuation and in general any improper use of the contextual language of the website.

Checking for broken link means assuring that every link in the website point to the expected location. It can happen that an hyperlink point to the wrong page or that a spelling error point to an entirely non-existing location.

If the hyperlink point to a third party website, the risk is even bigger, because is out of the control of the website owner or of the developer. The third party website can go offline or can be modified in such a way that the hyperlink breaks, pointing to something that isn’t there.

Other important part of content proofing is to make sure that the website is accessible by people with disabilities.

Question 8: Why is it not a good idea to manually check for broken links?

Although checking for broken links can be done manually, humans can easily make mistakes. It is a boring and repetitive task and it’s easy to get confuse and lose track of the process. Nowdays there are tools that allow to automatically check for broken links. These tools can already be part of the development environment or can be as-hoc programs. Online tools are also available for the task.

The existence of these automated tools makes checking the links in a website faster and more accurate than a human could possibly ever do.

Question 9: What sort of environment should be set up to conduct the various tests on a website?

The environment set up varies according to the test to be performed. Usually to avoid disruption in the production or in the developers work, a copy of the website is tested on a separate system.

In the case of load testing is a good idea that the environment should be an isolated replica of the real production one, to avoid problems for the other people on the network and at the same time guarantee that the results are meaningful. This involve using the same hardwares, using real data from the database and the simulation of both realistic and exagerated levels of traffic. This also includes the software used for the testing and implies that everything is properly licensed to ensure the legality of the operation.

The same applies for security testing, where apart from a simple code review, a replica of the server configuration can be set up and tried to be hacked by the pentester.

It is important that the testing environment replicates as closely as possible the production environment, to ensure that the tests are meaningful and accurate.

# **Unit Assessment Result Sheet (UARS)**

## **Assessment Task 1 – Unit Knowledge Test (UKT)**

## **Student and Trainer/Assessor Details**

|  |  |
| --- | --- |
| **Unit code** | ICTSAS519 |
| **Unit name** | Perform systems tests |
| **Outcome of Unit Assessment Task (UAT)** | |  | | --- | | **First attempt:** |   Outcome (please make sure to tick the correct checkbox):  Satisfactory (S)  or Not Satisfactory (NS)  Date: \_\_\_\_\_\_\_(day)/ \_\_\_\_\_\_\_(month)/ \_\_\_\_\_\_\_\_\_\_\_\_(year)   |  | | --- | | **Second attempt:** |   Outcome (please make sure to tick the correct checkbox):  Satisfactory (S)  or Not Satisfactory (NS)  Date: \_\_\_\_\_\_\_(day)/ \_\_\_\_\_\_\_(month)/ \_\_\_\_\_\_\_\_\_\_\_\_(year) |
| **Feedback to Student** | |  | | --- | | * **First attempt:** |  |  | | --- | | * **Second attempt:** | |
| **Student Declaration** | * I declare that the answers I have provided are my own work. Where I have accessed information from other sources, I have provided references and or links to my sources. * I have kept a copy of all relevant notes and reference material that I used as part of my submission. * I have provided references for all sources where the information is not my own. I understand the consequences of falsifying documentation and plagiarism. I understand how the assessment is structured. I accept that all work I submit must be verifiable as my own. * I understand that if I disagree with the assessment outcome, I can appeal the assessment process, and either re-submit additional evidence undertake gap training and or have my submission re-assessed. * All appeal options have been explained to me. |
| **Student Signature** |  |
| **Date** |  |
| **Trainer/Assessor Name** |  |
| **Trainer/Assessor Declaration** | I hold:  🗹 Vocational competencies at least to the level being delivered  🗹 Current relevant industry skills  🗹 Current knowledge and skills in VET, *and undertake*  🗹 Ongoing professional development in VET  *I declare that I have conducted an assessment of this candidate’s submission. The assessment tasks were deemed current, sufficient, valid and reliable. I declare that I have conducted a fair, valid, reliable, and flexible assessment. I have provided feedback to the above-named candidate.* |
| **Trainer/Assessor Signature** |  |
| **Date** |  |
| **Office Use Only** | Outcome of Assessment has been entered onto the Student Management System on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (insert date)  by (insert Name) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |