Oliver Collins-Cope

2102775@rutc.ac.uk

Learning Aim A

Understand the software development and testing methodologies commonly used during the development life cycle to quality assure software.

UNIT 13 SOFTWARE TESTING

Assignment 1

Contents

[Introduction 2](#_Toc131502752)

[User requirements and typical software job roles 2](#_Toc131502753)

[User requirements 2](#_Toc131502754)

[Typical job roles 2](#_Toc131502755)

[Software developers 2](#_Toc131502756)

[Testers 2](#_Toc131502757)

[Business analysts 2](#_Toc131502758)

[Project managers 2](#_Toc131502759)

[Product owners 2](#_Toc131502760)

[Characteristics of common software testing methodologies 2](#_Toc131502761)

[Unit testing 2](#_Toc131502762)

[Acceptance testing 2](#_Toc131502763)

[Functional testing 2](#_Toc131502764)

[Performance testing 2](#_Toc131502765)

[Security testing 2](#_Toc131502766)

[Regression testing 2](#_Toc131502767)

[Stress testing 2](#_Toc131502768)

[Usability testing 2](#_Toc131502769)

[Features of testing for different software development methodologies 2](#_Toc131502770)

[Agile testing methodology 2](#_Toc131502771)

[Waterfall testing methodology 2](#_Toc131502772)

[Kanban testing methodology 2](#_Toc131502773)

# Introduction

Throughout this unit I will explore the importance of software testing and development process as well as different types of testing the tools and techniques used to perform them. In this paper I will be focusing on the fundamental principles of software testing. This paper will aim to document the various types of testing, their significance in the software development life cycle, and the importance of using different testing tools and techniques in order to ensure that the quality of the final product is maintained. Finally, this paper will also document use cases of different testing methodologies.

# User requirements and typical software job roles

## User requirements

The user requirements of the specific needs, wants, and expectations of the stakeholders who will be using the software system. Understanding these requirements, such as through gathering or defining them, is a vital step in the software development life cycle. This is due to the fact that it ensures the software system is designed undeveloped with its initial intended audience at the forefront of development.

From the perspective of a software tester, it is crucial to have an in depth understanding over the user requirements. This will ensure that when software testing, the software tester will be able to ensure that the software is meeting the user requirements. Furthermore an in depth understanding of the user requirements would allow the software tester to review and analyse them, giving them the opportunity to identify any potential issues or conflicts and raising these issues with the development team.

Finally an in depth understanding of user requirements will aid in identifying the scope of testing, such as how complex or critical the testing should be of the software, and therefore help in developing appropriate testing strategies for the software. Through the identification of different parts of the software such as features, functions, or other relevant aspects the end users might interact with, the software tester can develop different test cases that will aid in the applicable use cases and scenarios.

## Typical job roles

### Software developers

One typical job role in a software development team is a software developer. Among others, some of the job responsibilities of a software developer include:

* Developing software applications, this is where software developers design, develop, and maintain software through the use of different tools such as programming languages and other available development tools.
* Write and maintain code, this is why software developers writing and maintain code from the software applications that will later be used in testing.

Some typical skills of a software developer include:

* Programming skills
* Problem solving skills
* Communication skills
* Attention to detail
* Time management skills
* Adaptability

A combination of all of these things allow software developers to play a vital role in the software development life cycle, including job duties like designing, developing, and maintaining the software applications that they create.

### Testers

Software testers play a crucial role in the software development life cycle, as they are responsible for ensuring the software applications meet the end user requirements. Furthermore software testers are responsible for testing a variety of aspects of software including, functionality, usability, performance, and security. Some job responsibilities include:

* Developing and executing test plans, software testers create test plans to ensure that the applications they are testing successfully meet the end user requirements.
* Identifying and communicating bugs, software testers focus on finding issues and bugs within the software during their testing periods and work with the software developers to ensure that the bugs are resolved before the final release.
* Analysing test results, software testers must analyse the results of the software testing in order to ensure that any issues are found and consequently reported

Some typical skills of a tester include:

* Knowledge of software testing methodologies
* Attention to detail
* Communication skills
* Technical ability

### Business analysts

Contrary to software test is a software developers, business analysts play crucial role in ensuring in software applications align with the overall business strategy as outlined previously in the development of the project. Typically they have a large skill set ranging from business domain knowledge, analytical skills, and communication skills. This enables them to effectively complete their job as business analysts. Some job responsibilities include:

* Gather and analyse requirements, due to the nature as analysts they must analyse very acquirements from stakeholders, such as the users of the product and customers, in order to be able to translate their needs into functional requirements that the software developers and testers can use to create a successful product.
* Documentation, as business analysts are required To document all of the functional and non-functional requirements they have successfully analysed.

Some typical skills of a business analyst include:

* Business domain knowledge
* Analytical and problem-solving skills
* Communication skills
* Technical ability

### Project managers

Project managers play more of a central role in the development of software. Rather than having one specified function, they are often required to branch out and perform a wide range of functions in order to ensure a successful project. Some job responsibilities include:

* Defining the project scope, this is why project managers identify the goals and objectives of the project and through this develop an acceptable timeline.
* Develop project plans, based on the previously identified goals, objectives, and timeline, project managers develop project plans that outline the tasks that must be completed within the projects time.
* Managing project teams, this is where project managers manage the teams by assigning task and monitoring their progress.
* Monitoring and controlling projects risks, project managers help to identify and monitor any project risks that arise through the development of risk mitigation strategies, and taking corrective action when is necessary for the project.

Some typical skills of a project manager include:

* Leadership skills
* Communication skills
* Time management skills
* Risk management skills

### Product owners

Product owners require a combination of many skills including, customer focus, strategic thinking, communication, and analytical skills. These skills enable product owners to effectively play a crucial role in defining product vision and strategy. Some job responsibilities include:

* Defining product vision and strategy, this is where product owners have to define the product vision and strategy through the identification of customer needs which is typically provided by the business analyst.
* Developing product road maps, this is where product owners develop road maps that outline the future product features and functionality including timelines and delivery schedules in order to present a base timeline for the project.
* Prioritising product backlog, this is where product owners have to prioritise different backlog of features based on which features and functionality is most important valuable to the customer.
* Testing and validating product features, finally product owners have to test and validate product features to ensure that they meet the requirements of the customer that they previously outlined for the development team.

Some typical skills in a product owner include:

* Customer focus
* Strategic thinking
* Communication skills
* Analytical and problem-solving skills

# Characteristics of common software testing methodologies

## Unit testing

Unit testing is a software testing technique that involves testing individual parts of the software application while it is isolated. Further elaborate on this, this means that unit testing involves testing each individual part of the code separately in order to ensure that its functions as intended and producing the expected output.

Some of the key features of unit testing include;

* It can be automated based on specialised testing tools or framework
* It is typically performed by developers and testers who have knowledge of the programming language it used to develop the application
* It usually involves test cases that are for specific situations
* Unit tests are designed to be able to test even the smallest parts of an application and this enables developers and testers to catch errors and defects early in the development life cycle
* Unit testing is especially useful in identifying and isolating individual units of code the hold bugs

Furthermore, unit testing is a kind of white box testing which means that the tester has access to the application code and internal workings. It is important to make this differentiation as black box testing, such as system or acceptance testing, focuses purely on the external behaviour of an application and therefore does not access the code of the software. (TechTarget Contributor, 2023)

## Acceptance testing

Acceptance testing is a software testing technique that focuses more on validating a software based on specific requirements that are previously outline and detailed in order to be accurately used. This means that unlike unit testing, it does not look at the code.

Some of the key features of acceptance testing include;

* Acceptance testing is performed on the complete software application to ensure that the final product meets the specific requirements of the users
* Acceptance testing focuses on validating the applications based on requirements, rather than catching errors and bugs
* Contrary to unit testing, acceptance testing is completed by testers who are meant to emulate end users, or the application stakeholders as opposed to developers
* Typically, acceptance testing needs manual testing which is different compared to unit testing

Finally, acceptance testing is a kind of black box testing, meaning that the tester does not have access to the code of the application and therefore only works with the developed software. It has a completely different objective compared to something like unit testing and does not occur until near the end of the software development life cycle. (Gillis, 2023)

## Functional testing

## Performance testing

## Security testing

## Regression testing

## Stress testing

## Usability testing

# Features of testing for different software development methodologies

## Agile testing methodology

## Waterfall testing methodology

## Kanban testing methodology