BASICS OF SOFTWARE TESTING

Mansi Thakur

Contents

1. Introduction 1
2. What is Engineering? 1
3. What is Software Engineering? 1
4. What is the Software Crisis? 1
5. Why is Software Engineering Required? 1
6. Attributes of Software Engineering 2
7. What is a Program? 2
8. What is Software? 2
9. Difference Between Program and Software 2
10. Types of software 3

1. Introduction

To start learning about software testing, it's important to first understand what software is, how it's created, and why it's important. This document will explain the basics of software and software engineering in a way that's easy to understand.

2. What is Engineering?

Engineering is the use of science and math to solve problems and build things. Engineers use their knowledge to design and create machines, buildings, and systems that solve real-world problems. Engineering can be applied in many areas, such as building bridges, making cars, and even creating software.

3. What is Software Engineering?

Software Engineering is the process of designing, building, testing, and maintaining software. It's like engineering for computers and applications. Software engineers follow a set of rules and principles to create software that works well, is easy to use, and can be updated or fixed when needed. Software engineering helps make sure that software meets the needs of users and works without problems.

4. What is the Software Crisis?

The **Software Crisis** is a term used to describe the growing problems that come with creating software. As technology advances, software is becoming more complicated and harder to manage. Programs are bigger, have more features, and need to work on many different devices. This can lead to problems like bugs (errors), security issues, and software that doesn't work as expected. To fix these problems, software engineers need to use better methods to design and develop software.

5. Why is Software Engineering Required?

Software engineering is important because it helps solve many of the problems in software development. Here's why it's needed:

1. **Manage Complexity**: As software grows and becomes more complex, engineering practices help keep it organized and manageable.
2. **Quality Assurance**: Engineers follow specific methods to make sure the software works well and does what it is supposed to do.
3. **Scalability and Maintainability**: Software needs to grow and change as time goes on. Software engineering helps make sure it's easy to update and can handle more users or data in the future.
4. **Minimize Errors and Bugs**: By following good engineering practices, engineers can catch mistakes early in development, reducing problems later.
5. **Security**: Well-designed software is more secure and protected from hackers and other security threats.

6. Attributes of Software Engineering

When software is built using engineering principles, it should have the following qualities:

1. **Correctness**: The software should do exactly what it was designed to do.
2. **Efficiency**: The software should use resources, like memory and processing power, in the best way possible.
3. **Maintainability**: The software should be easy to fix and improve when needed.
4. **Reliability**: The software should work properly without crashing or breaking, even when it's used for a long time.
5. **Usability**: The software should be easy for people to use, with a simple and clear design.
6. **Portability**: The software should work on different devices and systems without too many changes.

7. What is a Program?

A **Program** is a set of instructions written in a programming language that tells a computer how to perform a specific task. These instructions are executed by the computer's processor to achieve a desired outcome. Programs can range from simple scripts that perform basic functions to complex applications that handle numerous tasks simultaneously.

8. What is Software?

Software is a collection of programs, data, and instructions that tell a computer how to perform specific tasks. Unlike hardware (the physical parts of a computer), software is intangible. Software can range from simple applications, like a calculator, to complex systems that manage the operations of entire businesses.

9. Difference Between Program and Software

Many people think "software" and "program" are the same, but they are different:

- A **program** is a specific set of instructions that a computer follows to do something. It is a single set of instructions to perform a specific task, focused on a single function or task, and relatively simple, performing one function. Programs typically lack proper documentation and are smaller in size, usually encompassing one application.
- On the other hand, **software** is a collection of programs and other tools that work together to perform a task. Software covers a wider range of tasks and often includes multiple programs, making it more complex and composed of multiple programs and components. Software is well documented and larger in size, often including many programs and tools. For example, a complete word processing program like Microsoft Word is considered software. Software can include many programs that do different things, such as formatting text or printing documents.

So, a **program** is a smaller part of **software**, but all programs are software.

10. Types of software

1. Product: Software's developed based on market requirements. Developed product-based companies like Microsoft.

2. Project: Developed for a specific customer. Developed by service-based companies.

These are further classified as:

- **System Software:** System software is designed to manage and control the computer hardware so that other software can run.
- **Application Software:** Application software is designed to perform specific tasks for the user, such as writing, browsing, or playing games.
- **Development Software:** Development software helps developers create, test, and maintain other software.
- **Middleware:** Middleware acts as a bridge between different software applications or between software and hardware.
- **Firmware:** Firmware is software that is permanently programmed into hardware devices and is used to control hardware.
- **Enterprise Software:** Enterprise software is designed to help businesses and organizations manage and automate their processes.