SOFTWARE TESTING

Trainer: Bharath

Email: Bharath.Nittech@Gmail.com

LinkedIn: https://www.linkedin.com/in/bharathnittech/

Module 1: Software Testing Fundamentals

What is Software & Software Testing?

What is Software?

Software is a general term used to describe any set of programs that controls the operation of a computer system.
(or)
Software is a collection of computer programs and related data that provides the instructions for telling a computer what to do and how to do it.
(or)
Software is a set of instructions that guides the hardware and tells it how to accomplish each task

Program:

The CPU of a computer understands a fairly limited set of very simple instructions, involving comparisons, arithmetic and data transfers (these are called machine-level instructions). Also, these instructions have to be represented in binary.

A program is an algorithm or set of algorithms, written using a programming language. Programs are often referred to as code, the process of writing programs is commonly called coding, and the people who write programs are called programmers or coders.

In order to convert a program into a set of binary-coded machine level instructions, one needs to compile the program. A compiler is a special program that accomplishes this task – it converts programming language instructions into machine-level instructions.

Types of Software's:

- System Software
- Programming Software
- Driver Software
- Application Software

System Software:
System software is computer software designed to operate the computer hardware to provide basic functionality and to provide a platform for running application software.
Example: Operating system and all utility programs that manage computer resources.

Programming Software:
Programming software is a toolset in the form of programs or applications along with programming language that software developers use to create, debug, maintain, or otherwise support other programs and applications.
programming language is a language that is used to code an algorithm in a way that a computer can eventually understand. Popular programming languages include Basic, Pascal, C++, Perl and Java.

Example: Eclipse, SQL Server etc...

Driver Software:

Driver software is a collection of computer programs that operates and controls devices that are plugged into a computer. These drivers make it possible for devices to perform their necessary functions.

Example: Printer Drivers, Audio Drivers etc…

Application Software:

A program or group of programs designed for end users .it Allows end users to accomplish one or more specific tasks. Application software is installed on a computer or mobile device based upon a user's need.

Example: Internet Banking Application, Online Photo Editor etc…

Projects:

A project is defined as an effort to create or modify a specific product or service. Project is something that is developed based on particular customer's requirements. Projects are temporary work efforts with a clear beginning and end. Projects can be completely contained within a specific unit or department, or include other organizations and vendors. A work effort may be considered a project if it meets the criteria established by the organization.

Product:

Product is something that is developed based on the company specifications and used by multiple customers.

Note: The product-based company will first have general survey in the market. Gather's clear requirements from different customers, and based on common requirements of so many customer`s. They will decide the specifications (Requirements).

Requirement:

Requirement is description of the services that a software system must provide and the constraints under which it must operate. Requirements can range from high-level abstract statements of services or system constraints to detailed mathematical functional specifications.

Requirement Document / System Requirements Specification (SRS):

Requirement Document is a document or set of documentation that describes the features and behaviour of a system or software application. It includes a variety of elements that attempts to define the intended functionality required by the customer to satisfy their customers/users.

In addition to specifying how the system should behave, the specification also defines at a high-level the main business processes that will be supported, what simplifying assumptions have been made and what key performance parameters will need to be met by the system.

Main Elements in Requirement Document:

Depending on the methodology employed (agile vs waterfall) the level of formality and detail in the SRS will vary, but in general an SRS should include a description of the functional requirements, system requirements, technical requirements, constraints, assumptions and acceptance criteria. Each of these is described in more detail below:

- Business Drivers

- Business Model
- Functional and System Requirements
- Business and System Use Cases
- Technical Requirements
- System Qualities
- Constraints and Assumptions
- Acceptance Criteria

Project Kick off meeting:

It is an initial meeting conducted in the software company, soon after the project is signed off, in order to discuss the overview of the project and also to select a project manager. Usually Project managers, technical managers, Quality managers, High level management, Test leads, Development leads and sometimes customer representatives will be involved in this meeting.

Project Initiation Document (PID)

A Project Initiation Document defines the project scope, management and overall success criteria that the team can go back to during the project. It contains the basic information of the project such as context, scope, team, and collaboration. It is equally important as an internal guide and for external stakeholders

Product vs Services Based Companies

There are two types of organizations the IT industry is divided into – "Product based IT company and Services based IT company." Both these verticals have their own importance and they are similar in a few aspects but are different in many aspects. Let's delve into the core concept that defines these two organization types.

Product-based IT Company – This type of organization is driven by an idea to help different customers. Here the main focus would be on "what should I do to make my product better". They are continuously working on building a software product using various technologies. Some of the known examples are – Microsoft, Paytm, Cisco, Adobe, Novel Vox, etc.

These companies have their own products and when used by other customers they are customized as per the client's needs. The products could be for the B2B or B2C market which purely depends on the usage of the product.

Services based IT Company – On the other hand, a service-based IT company is driven by customer needs. They offer services and solutions as per customer requirements.

Such organizations do not have any products of their own but help other organizations/individuals to build products or provide services to run their IT needs. Some of the known examples are Infosys, Wipro, TCS, cognizant, etc.

You must be wondering how does it matter for a developer whether it's a Product or Services Company as he/she has to ultimately code. Here is the catch:-)

Though the developers of both types of organizations are ultimately coding, there is a huge and visible difference in the working style of these organizations, their culture, their approach, etc.

Here are some of the major differences between both types of organizations

Product-based IT Company	Services-based IT Company
Innovation: Here you will get challenges that will take you towards innovation. Every time something better is expected so if you are keen to learn, you will get good problems to solve. There is always a scope of doing something better to enhance the product line as per competition.	**Monotonous:** Here, you would be given a fixed set of work with a fixed deadline. No one is going to ask for anything extra if you are done with your task. There will be no challenges as the work is planned and allocated as per client requirements. Many times, you keep doing the same work as that's the need of the business as a service to be given to the client.
Joy of accomplishment: If you are coming up with a new idea and working towards it, you get satisfaction and a sense of accomplishment.	**Just completing tasks:** Since the requirement is coming from the client, once a task is allocated, you are just completing the task.
R&D at its highest peak: Over here, everyone is expected to keep doing R&D as any new idea is always welcomed. It may change the whole business line. So, people are encouraged to speak up and share their ideas.	**Less scope of R&D:** Since the requirements are frozen at the start of the work, there is less scope of R&D for all. A specific team takes care of that area and not everyone gets the chance to research.
Code Quality Check: Since you are building a product, its quality is at the highest priority. A lot of importance is given to your code quality and efficiency.	**Code Quality Check not a priority:** As your company is not creating anything for themselves apart from good relations with the client, they will not bother you until the client bothers them. So, you may not worry about code quality.
Growing Tech Stack: Here, the chance of working on new technology is very high as the world is changing every single day and you need to keep upgrading your product accordingly.	**Limited Tech Stack:** Here, people may end up using the same technology for years as there are different teams for different technologies, so the scope of growing the tech stack is very low.
Healthy Competition: As such a place is filled with great talent, you always get a strive to be better. There is always a healthy competition to learn more and provide better solutions. Multitasking is appreciated the most.	**Political Environment:** As they have many people to do the same task, people tend to get in politics. Technical competency doesn't play a huge role, thus usually office politics, backstabbing, flattery take the front seat during appraisals and reviews. As a consequence, you see a lot of arrogance from people and a lot of chaos in project executions.
Business understanding: In a product company, you need to understand the whole business idea to come up with more ideas so the freedom to expand your business acumen is high.	**Domain understanding:** Since most of the services companies specialize in a particular domain only, so there is less scope of understanding the whole business. You may gain expertise in a particular domain like health care, telecom, etc.
Higher salary benchmark: The focus is more on talent and thus good paymasters. There is no client billing here so they cautiously choose better talent and give them better salaries so that they stay for long with them.	**Lower salary benchmark:** The focus is on billing, so more billing, more people are allocated to a project, so they may compromise on talent and pay a particular range only.