Intro to Software Testing

Interview Questions:

- What is software testing?
- Who performs testing ?
- How is software testing done? What are the testing procedures?
- Why is software testing required?

Q) What is software testing?

Software testing is a **process** to try to **ensure** software **quality by finding bugs**. Software testing is performed to check if the software meets the client's requirements. Testing **measures** a **software's** overall **quality** in its correctness, completeness, performance, and other functional and non-functional attributes.

In short, software testing is done to verify if the software meets the customer's requirements, if it has high quality, if it is bug-free and fit for use.

Q) What to test in software testing?

In software testing, not just the app itself being tested. All the outputs of each SDLC phase are tested.

- **Documents** are tested in the early steps of SDLC to prevent bugs.
- **Codes** that developers created to build the app
- The **software**/system **itself** from a functional and non-functional perspective.

What is Sotware Testing?

- A **process** to **assure** the **quality** of a software
- The **goal** is to **prevent** & **identify** any bugs in the app.
- Verify if the **actual** software **meets** the client's **expected requirements**

What is to be tested in software testing?

- Documents
- Codes
- Software's functional and non-functional parts

CYDEO

Q) Who performs testing?

Each step of SDLC requires performing testings, and different people are responsible. Testing is/can be done by all technical and non-technical people associated with the project. People who perform testings are:

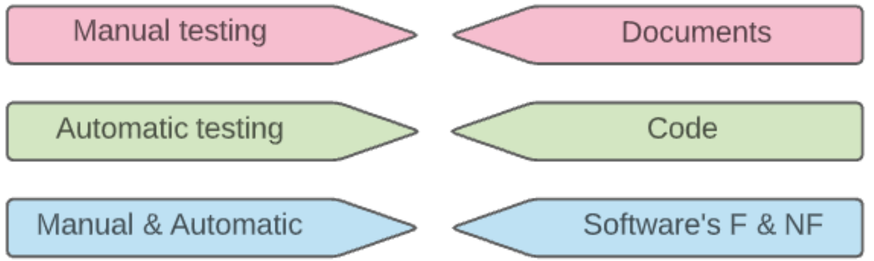
1. **Business team members & Designers**: **Test** SRS, SDS and many other business **docs** called static testing.
2. **Developers**: Developers **test** their **codes** by performing **Unit** & **Integration testings** to ensure that the individual unit of the software work correctly.
3. **Testers:** Testers verify if developers developed the app as business required - it is called **System testing**. System testing is divided into two parts: **functional** & **non-functional testing**. Functional testers test the application's functionality; performance testers check the application's non-functional aspects.Testers also prepare various testing documents like test plan doc, test case doc, bug report doc.
4. **The client/stakeholders:** To ensure that the final product meets the requests of the client, its called **UAT**(User Acceptance Testing)- **Beta testing**.
5. **End users:** A group of end-users do the application's **UAT-Alpha testing** to make sure the software can work in the real world.

Who performs Testing?

- Business team members
- Designers
- Developers
- Testers
- The client / stakeholders
- End users

How is software testing done?

- Manual testing
- Automatic testing



CYDEO

Q) How is Software Testing Done?

Testing are done **manually** and **automatically**.

Manual testing refers to a manual test process to identify bugs by humans. Application must be tested manually before it is automated.

Manual testing

PROS :

1. Test all kinds of applications manually
2. Handle difficult functionalities
3. No environment limitations
4. Programming Knowledge is not required
5. Accurate UI feedback

CONS:

1. Time-consuming
2. Requires more human resources
3. Not all testing can be done manually - especially performance testing
4. Not repreduciable
5. Testing process is slow

Automated testing is done through an automation tools and is generally more accurate than manual testing.

100% automation is not possible.

Automation testing

PROS :

1. Cost-effective in the long run
2. reduces the overall test execution time
3. Can be re-used
4. Automated test scripts remove the chance of human error
5. Helps in **working with a large set of data**

CONS:

1. Automation tools has limitation
2. Skilled automation testing experts to write test scripts
3. Additional effort to write scripts is required upfront

	Manual Testing	*Automatic Testing*
Programming	Not required	Required
Reliability	Human error occur	Comparatively more reliable
Testing time	Slower	Faster
Reusability	Not reproducible	Reusable
Limitation	No tool or Env limitarion	Automation tool limitation
Budget	Cheaper for short-term goal	Cost-effective in the long run

What is the benifit of software testing?

1. Fast
2. Reusable
3. Cost Reduction
4. Improve Security
5. Customer Satisfaction
6. Reliable
7. Enhance performance
8. Repeatable
9. Better Quality Software
10. Required in SDLC