Testing and Beyond!

Created by Japneet Singh Sachdeva

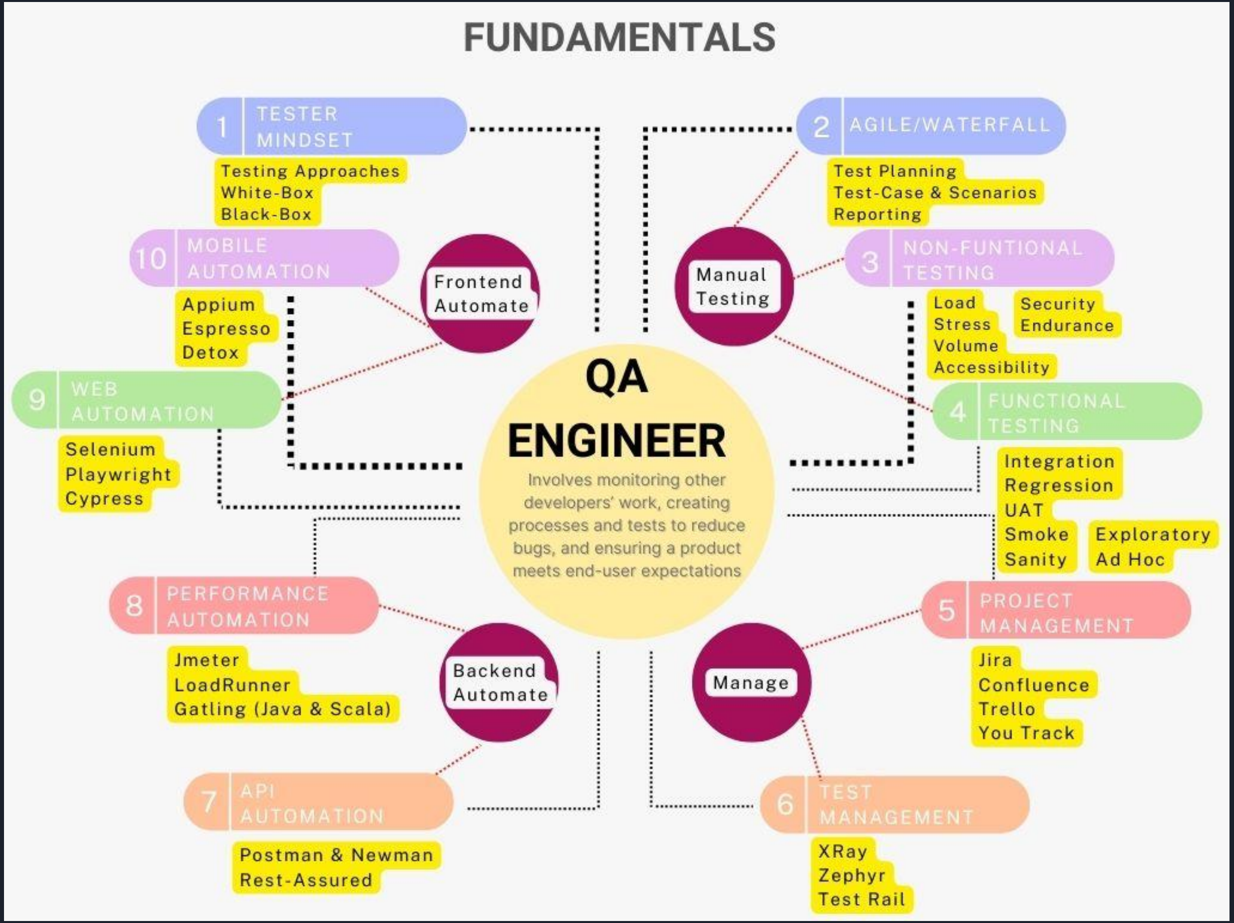
LinkedIn = Japneet Sachdeva

Learning Mindset is most important

Just remember

Every Day is a New Beginning and Every Day you can learn something new

RoadMap For QA Engineer



FUNDAMENTALS

QA ENGINEER

Involves monitoring other developers' work, creating processes and tests to reduce bugs, and ensuring a product meets end-user expectations

1 TESTER MINDSET
- Testing Approaches
- White-Box
- Black-Box

2 AGILE/WATERFALL
- Test Planning
- Test-Case & Scenarios
- Reporting

Manual Testing

3 NON-FUNTIONAL TESTING
- Load
- Stress
- Volume
- Accessibility
- Security
- Endurance

4 FUNCTIONAL TESTING
- Integration
- Regression
- UAT
- Smoke
- Sanity
- Exploratory
- Ad Hoc

Manage

5 PROJECT MANAGEMENT
- Jira
- Confluence
- Trello
- You Track

6 TEST MANAGEMENT
- XRay
- Zephyr
- Test Rail

Backend Automate

7 API AUTOMATION
- Postman & Newman
- Rest-Assured

8 PERFORMANCE AUTOMATION
- Jmeter
- LoadRunner
- Gatling (Java & Scala)

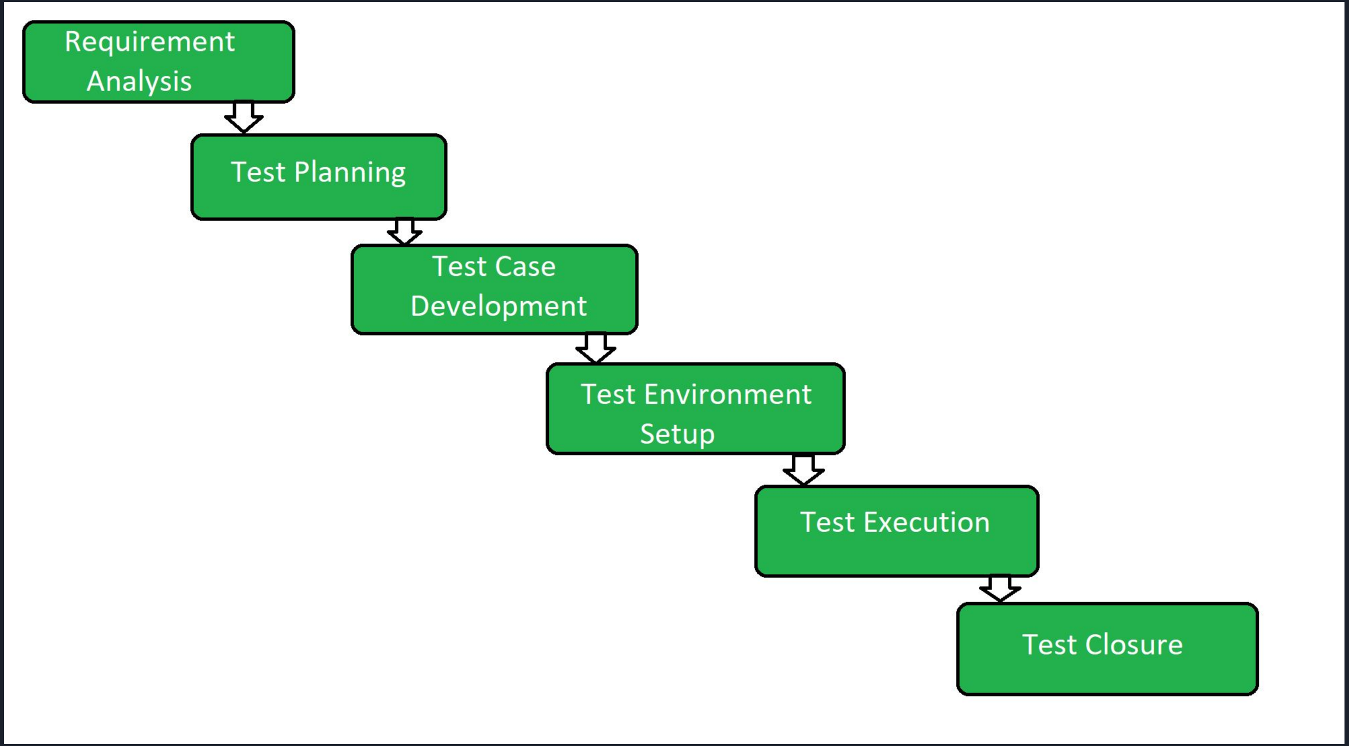
Frontend Automate

9 WEB AUTOMATION
- Selenium
- Playwright
- Cypress

10 MOBILE AUTOMATION
- Appium
- Espresso
- Detox

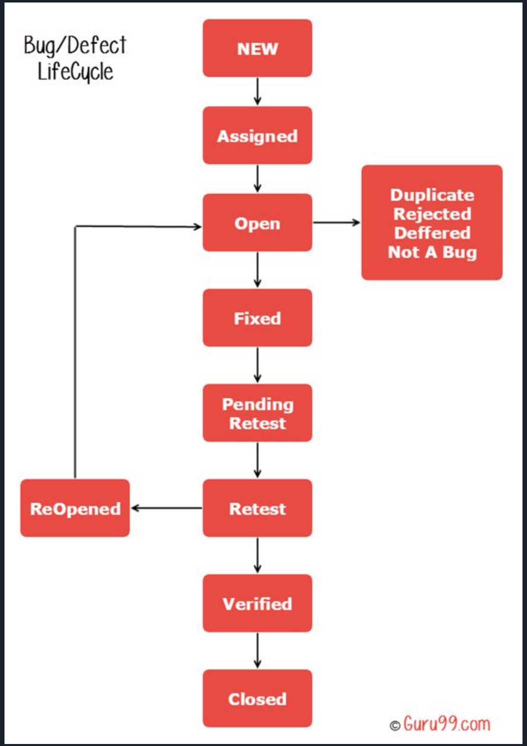
STLC

STLC - Software Testing Life Cycle



Defect Cycle

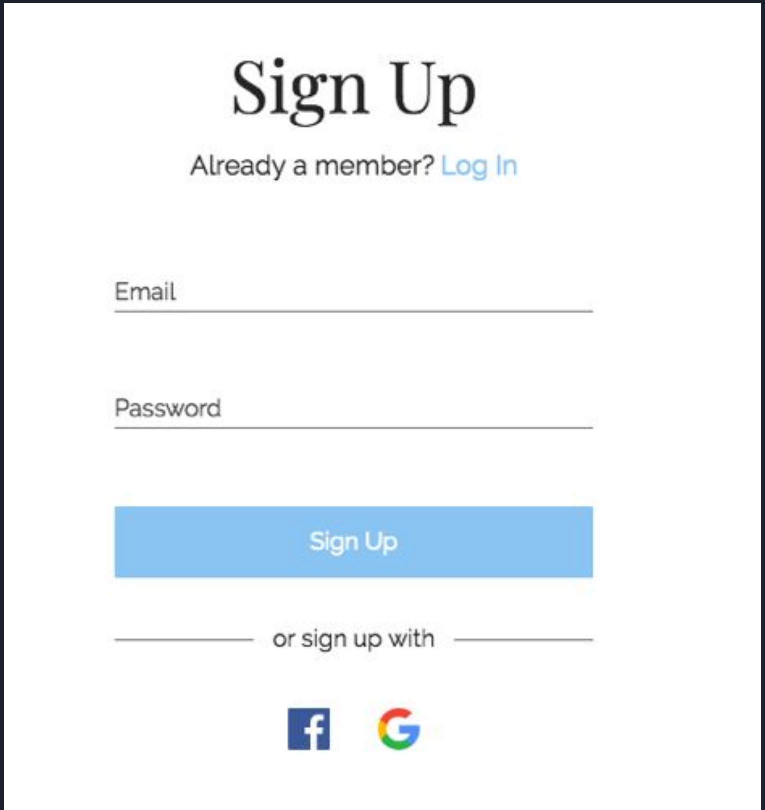
Defect management



Why Testing?

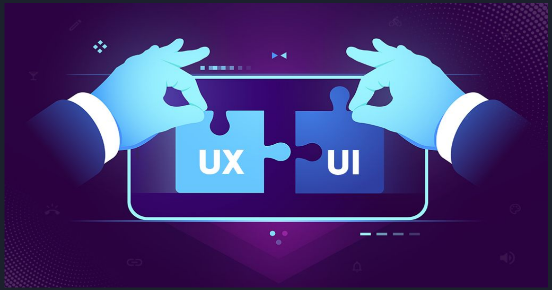
Testing proves a software is functionally and non-functionally working as expected

Sign Up Page Test Scenarios

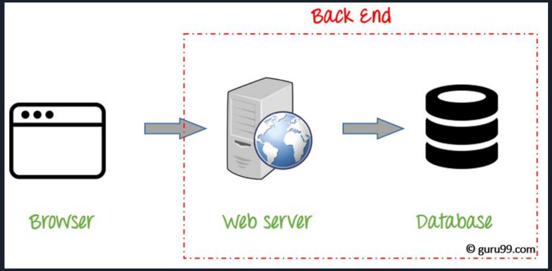


Types of Testing?

- Functional



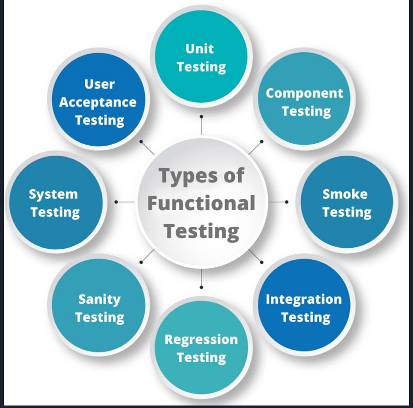
- Non-Functional

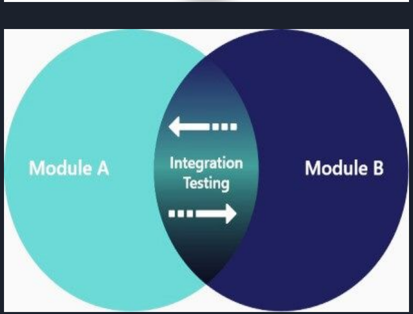


Functional Testing

- Integration
- System
- Regression
- Smoke
- Sanity
- Monkey (exploratory)
- Re-Testing
- Ad Hoc
- Accessibility

etc.

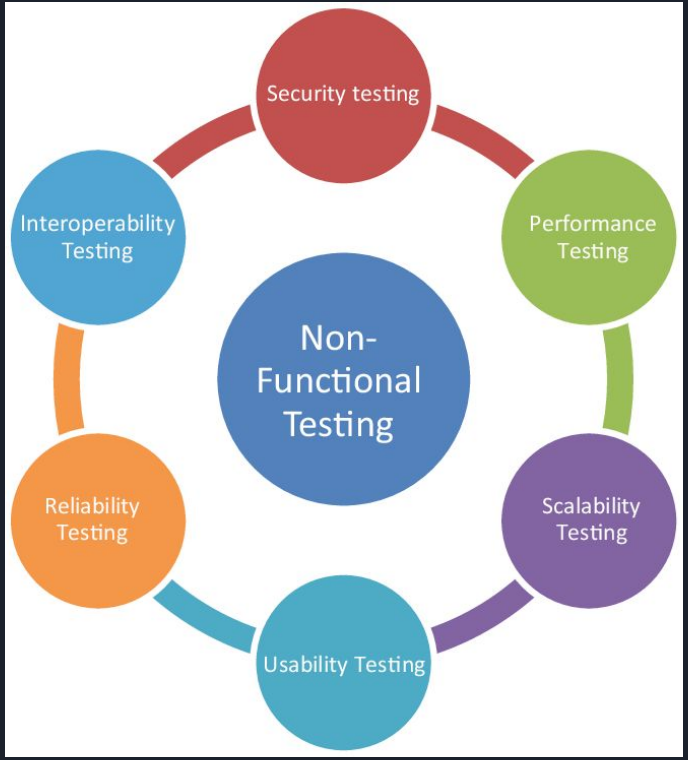




Non-Functional Testing

- Interoperability
- Load
- Volume
- Stress
- Reliability
- Security
- Usability

etc.

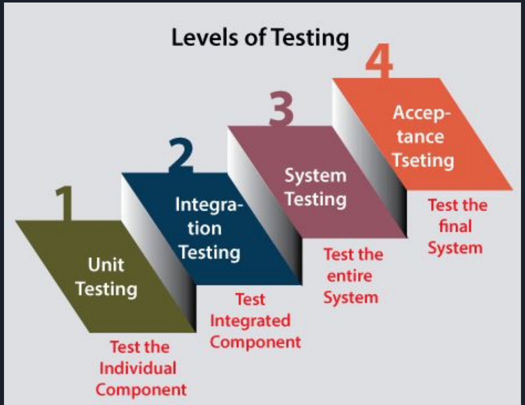


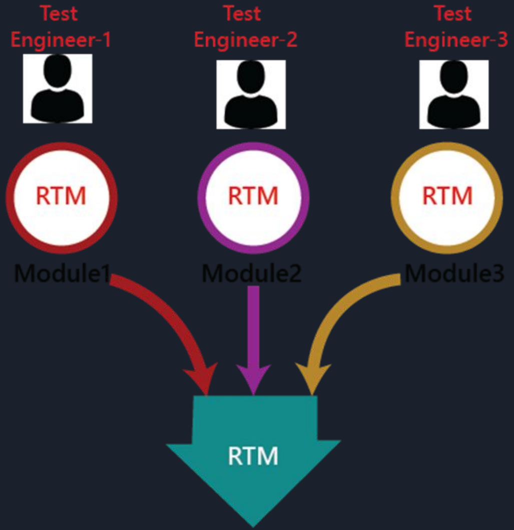
Preparation for Testing

How can we start to test an application?

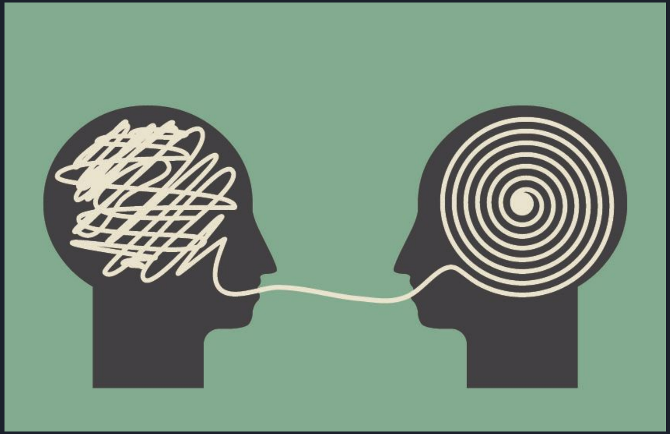
- Understand application under test
- Understand application use for end-user
- Understand business flow and its importance
- Understand Priority and Severity for defects based on application development timeline
- Interaction with Client and understand their perspective about features yet developed
- Understand Pesticide Paradox
- Understand Defect Clustering
- Understand Absence of error fallacy

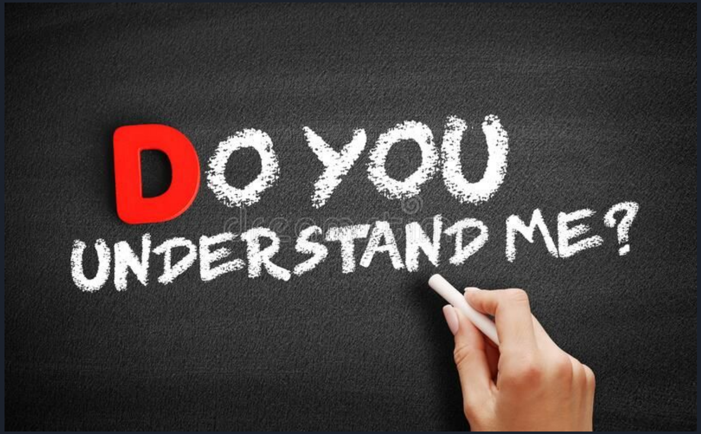
- Regression
- Smoke
- Sanity
- UAT
- A/B
- Re-testing
- Stress
- Load
- Volume
- Levels of testing and its importance (also when can you skip it?)
- Entry criteria
- Exit criteria
- Bug and defect ?
- Risk Analysis (Project risk, Product risk etc.)
- Verification and Validation
- Compatibility
- Cross browser
- Suspension criteria
- RTM and its requirement
- Show-stopper defect
- Error Seeding
- Defect cascading
- Latent defect





Are you able to understand all the terms defined in previous two slides (atleast)? If yes then you can start with test automation!





But before moving to automation,
Its very important understand the use and importance of Manual Testing (Although there should not be any term like Manual Testing because - No matter what you do related to Automation or Manually - We call it testing only!)

Manual testing acts as basis of knowledge so that you can become a better Test Automator, also there is no ending of learning. Just remember - You need to use Pesticide paradox!

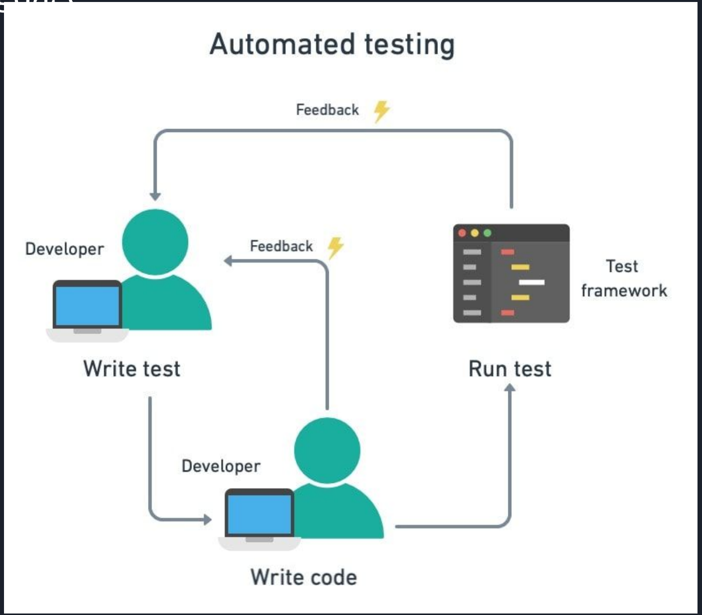
What is test Automation?

Automation means you are using a library which helps you to convert certain manual test-scenarios into Automated one's so that you can focus on other crucial parts of the application or its design

For Automation you must need to know a programming language -

1) Java
2) Python
3) C# (C sharp)
4) Javascript

etc.



Why test automation?

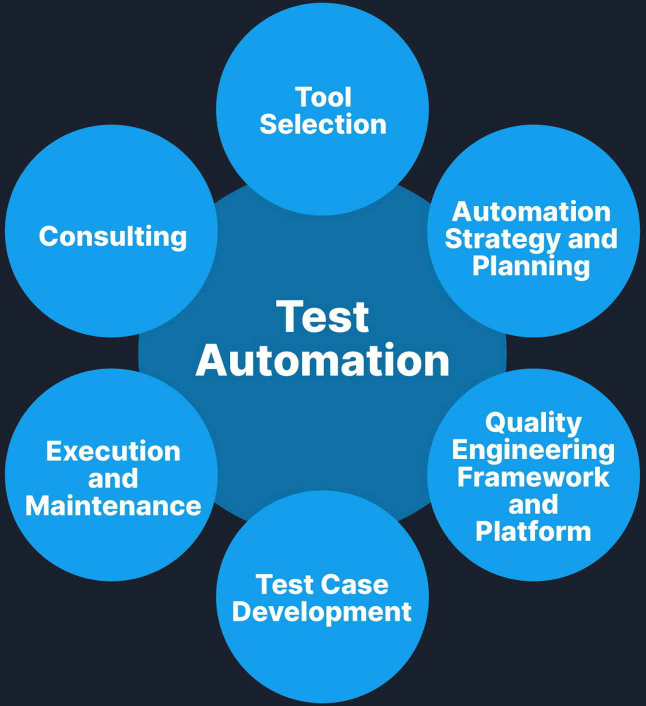
You can eliminate your efforts towards repeated tasks/test-scenarios

Some examples:-

1) Data driven testing
2) Smoke testing
3) Regression testing
4) API automation (using mock servers)

etc.

Process of development for Test Automation



Test Automation

Tool Selection

Automation Strategy and Planning

Quality Engineering Framework and Platform

Test Case Development

Execution and Maintenance

Consulting

Who can design test automation?

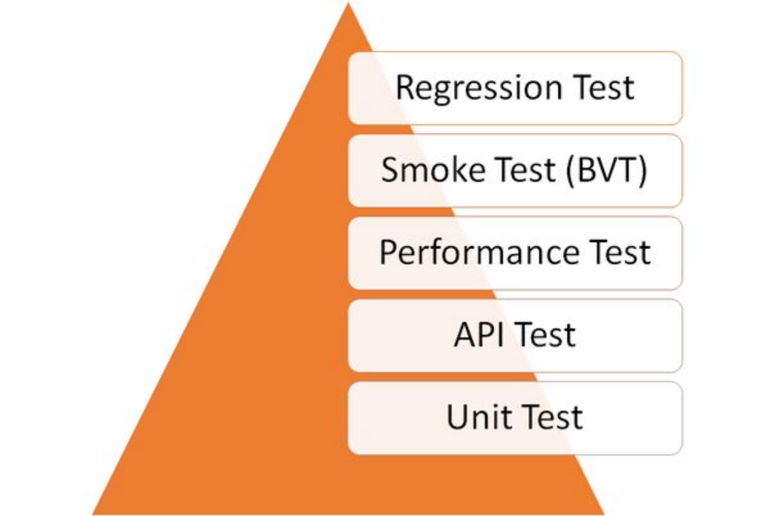
A tester who has experience in designing test-scenarios from base , understands which one of them are repeated and can be part of automation-suite

By designing I mean - A person who fully understands the application and then can assign what needs to be tested and how depending on priorities.

What type of testing is Automation?

You can apply Automation in different fields like :-

- Functional
- API
- Non-Functional (to some extent)



Let’s Discuss Functional Test Automation

Functional test automation has various fields we will discuss two most commonly used:-

1) Web-Application Testing - Libraries which can be used -

   i) Selenium (most widely used in market and best user support)

   ii) Playwright (*puppeteer is a base for Playwright development***)**

   iii) Cypress etc.

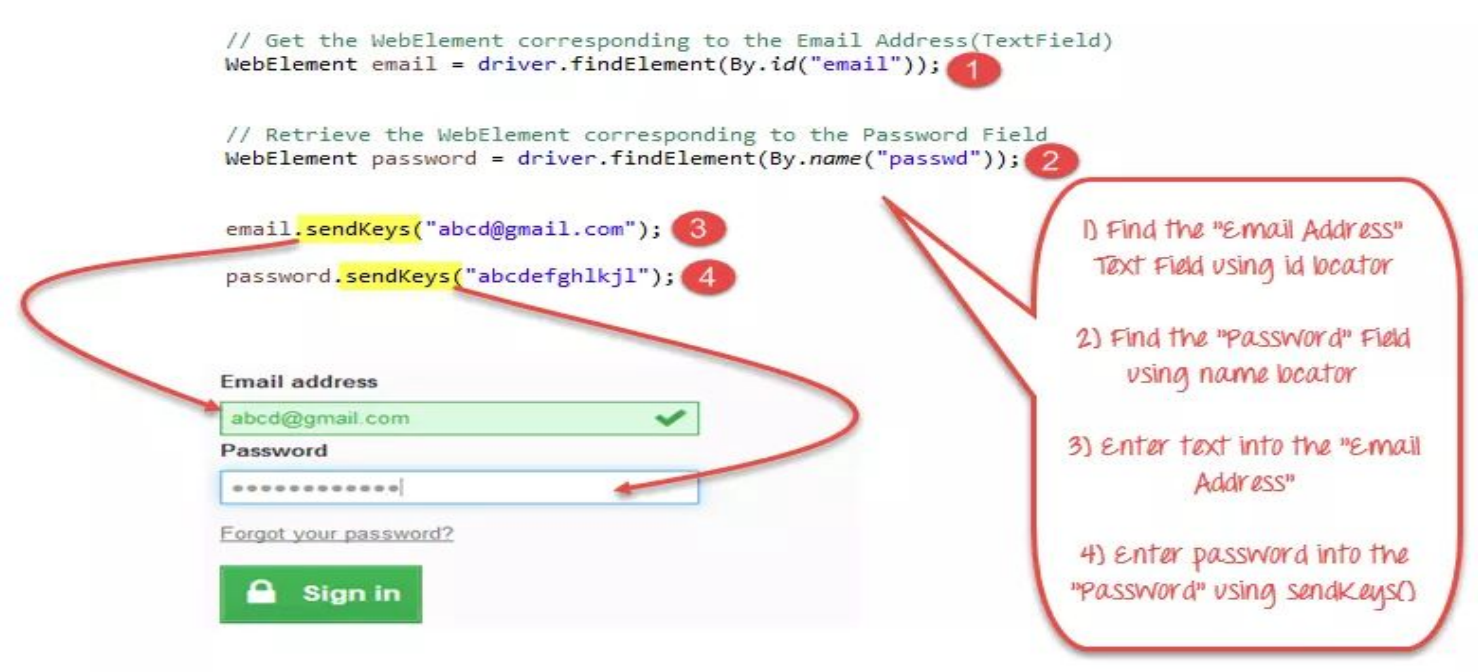
2) Mobile-Application Testing - Libraries which can be used -

   i) Appium (Supports both Android and iOS)

   ii) Espresso (developed by google for Android)

   iii) TestRigor (*Codeless automation but requires good understanding of test-scenarios*)

Example for using Selenium



```java
// Get the WebElement corresponding to the Email Address(TextField)
WebElement email = driver.findElement(By.id("email"));

// Retrieve the WebElement corresponding to the Password Field
WebElement password = driver.findElement(By.name("passwd"));

email.sendKeys("abcd@gmail.com");

password.sendKeys("abcdefghlkjl");
```

Email address

abcd@gmail.com ✓

Password

●●●●●●●●●●●●

Forgot your password?

Sign in

1) Find the "Email Address" Text Field using id locator

2) Find the "Password" Field using name locator

3) Enter text into the "Email Address"

4) Enter password into the "Password" using sendKeys()

Framework Building and Design Strategy

To start with this you can use TestNG and Page Object Model for Java as your framework goal

If you are using python then you can use PyTest and Page Object Model.

Playwright can be used with Javascript, Java etc with similar Design Patterns

Similarly it goes for Cypress as well. But it only supports Javascript as of now.

What is Security Testing?

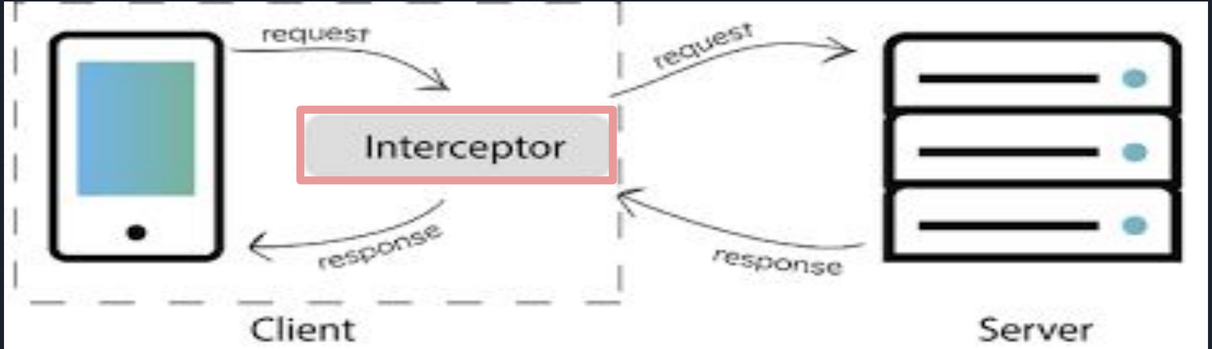
This helps in uncovering the vulnerabilities of an application so as to prevent malicious attacks.

Importance of Security Testing

We will discuss most basic form of it now - through interception

Now what is Interception? When you submit a form (for registration) there is a request made to a server, Using a tool (called Burp Suite, Fiddler, etc.) this request can be intercepted - Using this

You can map different data to that Body of request - Basically you can validate if there are server-side validations available or not. If not then you can save in-correct data in there application's database.



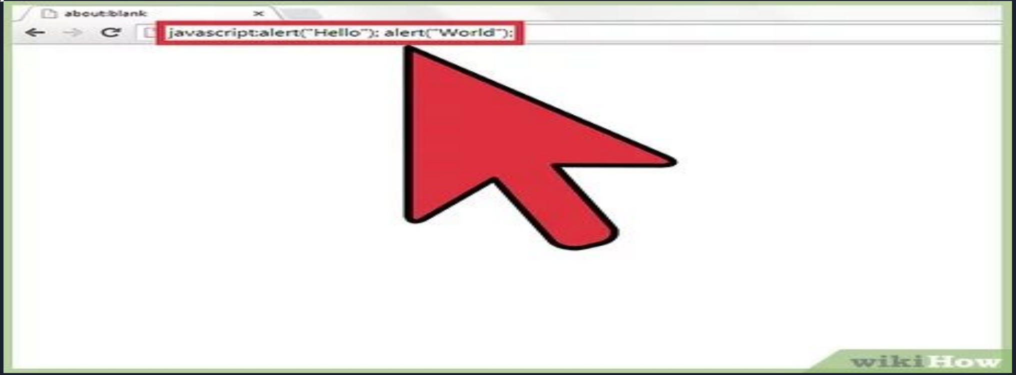
Penetration Tester/Testing (is not a hacker/hacking)

People often confuse Security testing and Penetration testing are same - but its not!

Penetration testing is a subset of Security - So in this we basically be testing the system

By penetrating irrelevant data or code into an application - so that application might lead to show

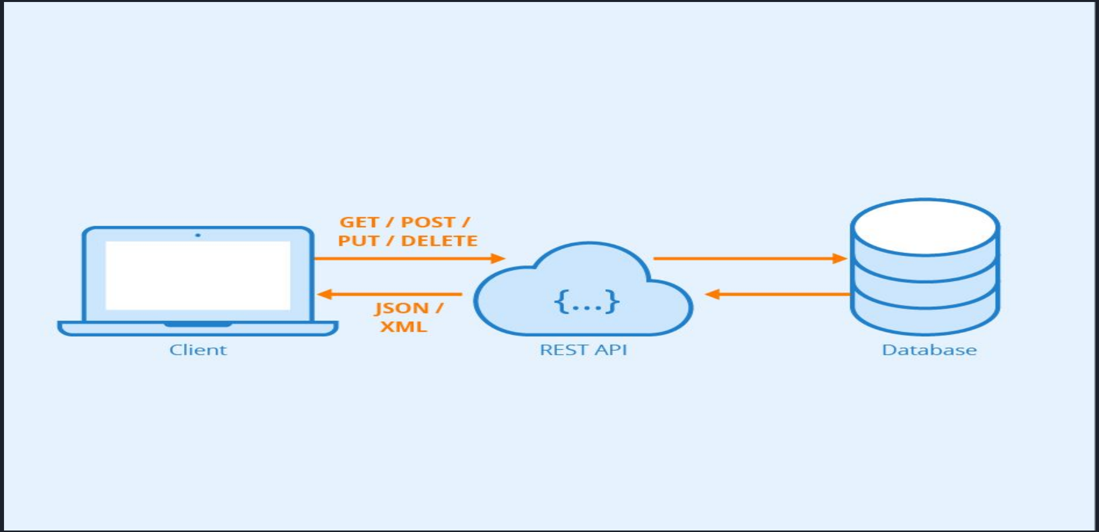
Some privacy concerns



```
javascript:alert("Hello"); alert("World");
```

What do you mean by API?

When you use an application on your mobile phone, the application connects to the Internet and sends data to a server. The server then retrieves that data, interprets it, performs the necessary actions and sends it back to your phone. The application then interprets that data and presents you with the information you wanted in a readable way. This is what an API is - all of this happens via API.



Importance of API testing

Most important factor for API testing is you can test an API even before your Functional Testing.

Because even when your UI is being developed you can discuss with your team for API contracts and then start mocking your own server and create and test the API

Once your backend is setup then you have to just update your base URLs and then start your testing

Major advantage in this is - once your test-scenarios are completed you can move your focus towards other types of testing for your application

API Test Automation

It's a part of Backend Automation

- You can start testing even before actual UI development is started
- API Contract is required
- Mock Server
- Tools - Postman/Newman & Rest-Assured

Design Patterns

Test Automation for functional testing can be developed in different Designs - Which we call different design patterns.

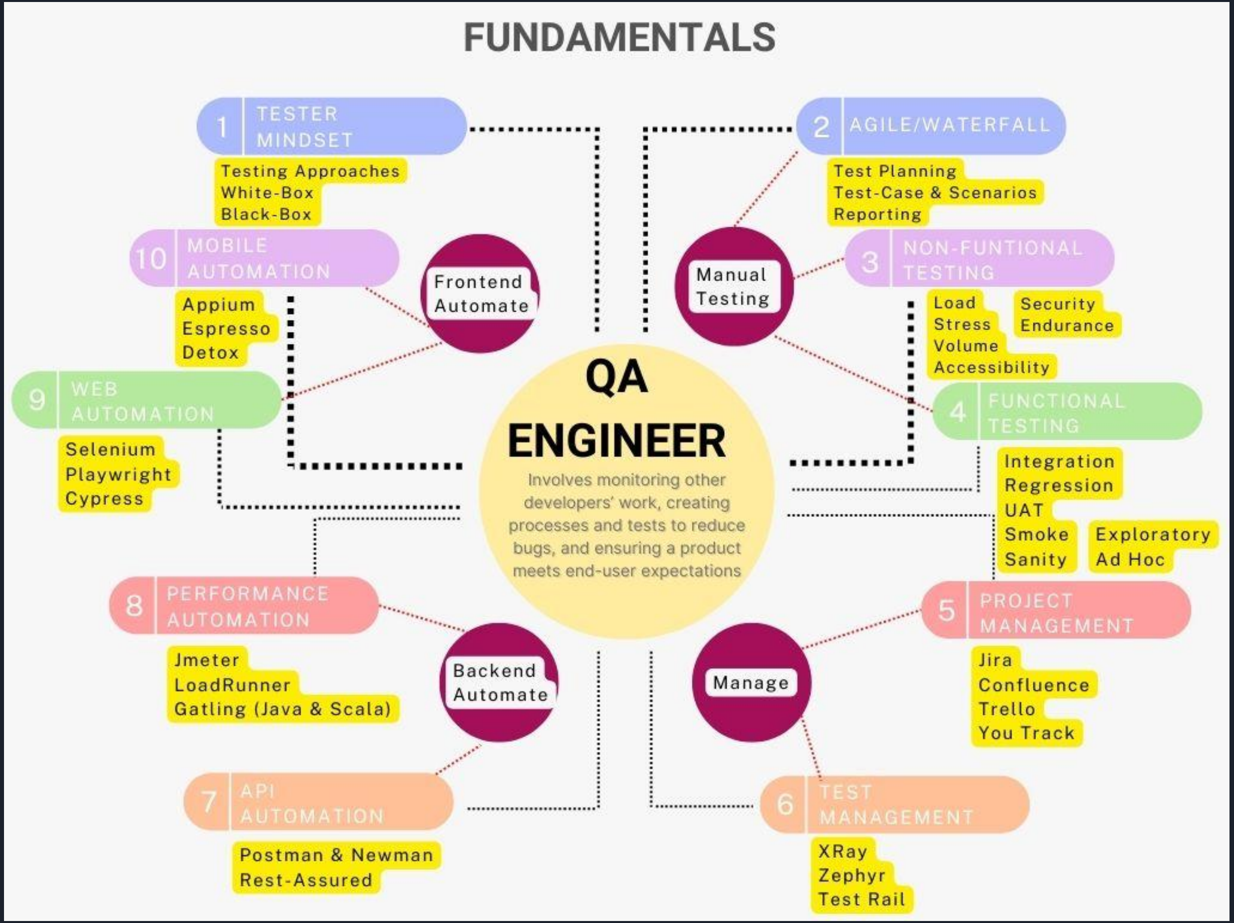
Most famous and widely used is Page Object Model - In this we develop a framework depending upon different pages available in an application

Similarly there are there design patterns:-

- Single Responsibility Principle
- Strategy Design Pattern

etc.

RoadMap For QA Engineer



FUNDAMENTALS

QA ENGINEER

Involves monitoring other developers' work, creating processes and tests to reduce bugs, and ensuring a product meets end-user expectations

Manual Testing

1 TESTER MINDSET
- Testing Approaches
- White-Box
- Black-Box

2 AGILE/WATERFALL
- Test Planning
- Test-Case & Scenarios
- Reporting

3 NON-FUNTIONAL TESTING
- Load
- Stress
- Volume
- Accessibility
- Security
- Endurance

4 FUNCTIONAL TESTING
- Integration
- Regression
- UAT
- Smoke
- Sanity
- Exploratory
- Ad Hoc

Manage

5 PROJECT MANAGEMENT
- Jira
- Confluence
- Trello
- You Track

6 TEST MANAGEMENT
- XRay
- Zephyr
- Test Rail

Backend Automate

7 API AUTOMATION
- Postman & Newman
- Rest-Assured

8 PERFORMANCE AUTOMATION
- Jmeter
- LoadRunner
- Gatling (Java & Scala)

Frontend Automate

9 WEB AUTOMATION
- Selenium
- Playwright
- Cypress

10 MOBILE AUTOMATION
- Appium
- Espresso
- Detox

Q&A

Important Takeaways

- Never stop learning!
- Knowledge is important but without implementation its pointless
- Everyday spent some time retrospecting
- Keep small goals and start achieving those to reward yourself with happiness