# LTI TESTING

# Software Testing

# Importance of Testing

# What happens if we don’t test?

# Software Testing Principles

1. Testing shows the presence of defects
2. Exhaustive Testing is not possible
3. Early Testing
4. Defect Clustering (Pareto Principle or 80:20 rule)
5. Pesticide Paradox
6. Testing is context dependent
7. Absence of errors fallacy

# Types:

1. Manual Testing
   1. White Box Testing: clear box, open box, transparent box, glass, structural
   2. Black Box Testing:
      1. Functional Testing (FT)
         1. Unit Testing
         2. Integration Testing
         3. System Testing (E2E)
         4. Acceptance Testing
            1. Alpha
            2. Beta
      2. Non Functional Testing (NFT): performance, security, usability, compatibility
2. Automation Testing

# SDLC: Software Development Life Cycle

1. Requirement Phase (SRS)
2. Design Phase
3. Development Phase
4. Testing Phase
5. Deployment Phase
6. Maintenance Phase

# STLC: Software Testing Life Cycle

1. Requirement Analysis
2. Test Plan creation
3. Environment Setup
4. Test Case Creation (Test Specification)
5. Test Case Execution
6. Defect Logging
7. Test Closure

# Waterfall Model

1. Requirement Collection: SRS
2. Feasibility Study
3. Design
4. Coding
5. Testing
6. Installation
7. Maintenance

# V Model (Verification and Validation) (V&V)

Verification <<>> Static Testing <<>> we don’t execute the code

Validation <<>> Dynamic Testing <<>> we execute the code

# Test Plan

# White box Techniques

1. Data Flow Testing
2. Control Flow Testing
3. Branch Coverage Testing
4. Statement Coverage Testing

# Black Box Techniques

1. Decision Table
2. Boundary Value Analysis
3. Equivalence Portioning
4. Error Guessing
5. State Transition Testing

# Types of Testing

Test Cases

RTM

Defects

Defect Life Cycle

TC activity

Pending

12th >> after 6 months >> graduation exam

Priority (high, Medium, low)

Severity (critical, major, medium, trivial)

Defect Activity

Confirm all the topics covered

Salary:

1. Planned knee surgery for mom

2. Bike for son

3. Daughter schoolfees in 1 week

4. Electricity bill

5. Accident case

HP and HS

HP and LS

LP and HS

LP and LS

## Selenium

1. Desktop

2. Web Application

## Selenium Suite

1. Selenium IDE
2. Selenium RC (deprecated)
3. Selenium WebDriver
4. Selenium Grid

## AGILE

## Core Values of Agile

1. Individual and team interactions over processes and tools
2. Working Software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan

## SCRUM

## Ceremonies in Scrum

1. Sprint (2 weeks, 3 weeks or 4 weeks)
2. Sprint Planning
3. Daily Scrum
4. Sprint Review
5. Sprint Retrospective

## Roles in Scrum

1. Product Owner
2. Scrum Master
3. Scrum Team

## API Testing

Application Programming Interface

## ReST API: Representational State Transfer

## SOAP API: Simple Object Access Protocol

## HTTP Methods

1. Post
2. Get
3. Put
4. Delete

API URL: <https://reqres.in/api/users>

API URL = hostname + endpoint

Host\_name = <https://reqres.in/>

Endpoint = api/users

## Request

## Response

## Response Code

Authentication vs Authorization

## DevOps/CICD

Continuous Integration and Continuous Delivery

## Components

1. Dev
   1. Build
   2. Code
   3. Test
   4. Plan
2. Ops
   1. Deploy
   2. Release
   3. Operate
   4. Monitor

## Devops Life Cycle

1. Continuous Development
2. Continuous Integration
3. Continuous Testing
4. Continuous Monitoring
5. Continuous Feedback
6. Continuous Deployment
7. Continuous Operations

## Executing from Maven

1. Make sure that testing.xml file is present
2. In pom.xml file add the plugin details
3. In pom.xml file, add the properties details
4. Pom.xml >> right click >> run as >> Maven Clean
5. Pom.xml >> right click >> run as >> Maven Install

## Executing from Jenkins

* Configuring Maven in Jenkins
  + Click on Dashboard >> Manage Jenkins
  + Click on Global tool configuration
  + Maven >> Maven Installation
  + Give name as Maven
  + Save
* Create Item in Jenkins
  + Click on Dashboard >> New Item
  + Give a name
  + Select Freestyle project
  + Click on Ok/Save
  + Give some description
  + Go to Build Step section
  + Click on Add Build Steps
  + Select Invoke top level maven targets
  + Select Maven version as Maven
  + Mention clean install in goals
  + Click on Advanced button
  + Paste the location of your pom.xml file from eclipse on POM field
  + Save the project
  + Click on Build Now button
  + Click on #1 in Build History Section
  + Click on Console Output

## Git Commands

1. **Creating a remote repository (One-time activity)**
   1. Login to Github
   2. Create a new Repository
2. **Setting the local repository and connecting to Remote (One-time activity)**
   1. Open Terminal/CMD
   2. Go inside the project folder which we want to push to remote repository
   3. Git init
   4. git remote add origin <https://github.com/ajaym19/LTITDDFramework.git>
3. **Transferring Codes and changes from local to remote**
   1. Git add filename or git add .
   2. Git status
   3. git commit -m "commitmessage"
   4. git push origin master