Agile BY Pawan

* Agile methodology is used in service based company.
* changes in product is allow at any time

What is Agile?

- Agile is an approach to software development under which requirements and solutions evolve through the collaborative effort of self-organizing and cross-functional teams and their customer/end user.

- Both development and testing activities are concurrent unlike the Waterfall model.

(Parallel)

* There is no SRS document.

Dev and tester work parallely, not unlike waterfall

Agile model /Agile methodology /Agile process.

It is an Iterative and Incremental Approach.
Agile is an Iterative and Incremental Process.

Agile principles

1) Customer no need to wait for long time.
2) We develop, test and release peice of software to the customer with few number of features.
3) We can accept/accomodate requirement changes.

There will be good communication between Customer, Business Analyst, Developers & Testers.

Advantages

Requirement changes are allowed in any stage of development (or) We can accommodate Requirement changes in the middle of development. But in Waterfall requirement are fixed
Releases will be very fast(Weekly)
Customer no need to wait for long time.
Good communication between team.
It is very easy model to adopt.

Disadvantage:
Less focus on design and documentation since we deliver software very faster.
No srs document.

* Aarthat Bynes 80?
* Atos syntel - 8.5
* Collebra → Wipro

dec-20
1:7

V model

Requirement Customer

BRS document

SRS document

function Requirement given in SRS

Product Manager

Agile Methodology

Requirement stake Holdee

product Backlog

Sprint Backlog

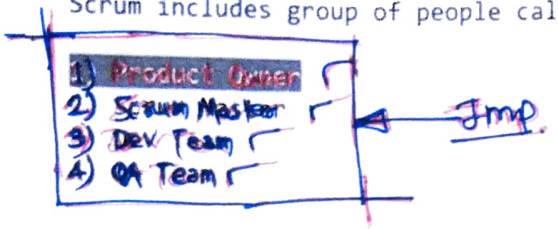
Function requirement in user stories

Scrum master

* Stake holdee are also customer but terms are used are different.

Scrum

Scrum is a framework through which we build software product by following Agile Principles.
Scrum includes group of people called as Scrum team. Normally contains 5-9 members.



Product Owner : (Imp)
Define the features of the product
Prioritize features according to market value
Adjust features and priority every iteration, as needed
Accept or reject work results.

Scrum Master: (project Manager)
The main role is facilitating and driving the agile process. He take care about process as well as product

Developers and QA:
Develop and Test the software.

-Product owner define feature of product in the form of user stories and BA not involve in agile. requirement directly come from product owner and he getting from customers..

Scrum Roles

- **The Scrum Master**
  - A scrum master is the facilitator for an agile development team.
  - Enable close cooperation across all roles and functions

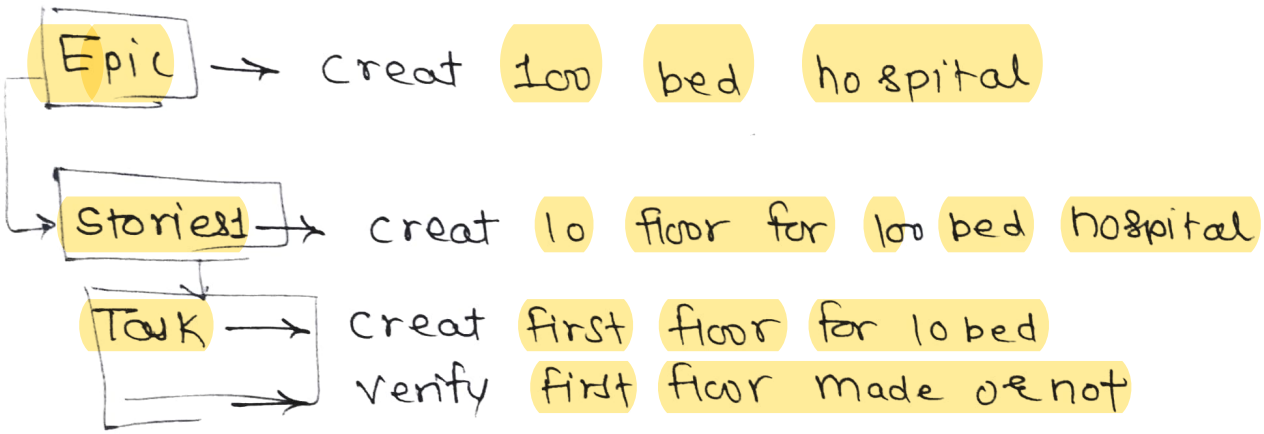
- **Product Owner**: (Stories)
  - Define the features of the product
  - Decide on release date and content
  - Prioritize features according to market value
  - Adjust features and priority every iteration, as needed
  - Accept or reject work results.

- **Scrum Team:**
  - Developers and QA

Epic → Its high level requirement, its large body of work that can be broken down into a small-small stories or sometimes called issue in jira.
its delivered into multiple Sprints

~~Ticket~~

Project



Stories 2 → arrang All hospital equipment

* epic contain mutiple stories and each stories contain multip task
* once you ~~comple~~ / Product owner complete or load user stories into project backlog the we can add then into sprint backlog as per priority
* We can creat spring and simply add and remove our stories by drag and drop and start Sprint
then see dashboard
We can assign this all stories to various Persons
Task done, check go to Active Sprints ~~Story~~ (i.e. Complete this story) → when we go to complete sprint it show to convert into other stories

Scrum master is not talking care of product its only take care about process.drive complete agile process.he is immediator between team and product owner

Sprint planning or Estimation meeting :- involved

1) PO 2> development lead 3) Tester Lead

Scrum Terminology

User Story : A Feature/module in a software

Epic : Collection of user stories.

Product backlog : Contains list of user stories. Prepared by product owner.

Sprint/Iteration : Period of time to complete the user stories, decided by the product owner and team, usually 2-4 weeks of time.

Sprint planning meeting: Meating conducts with the team to define what can be delivered in the sprint and duration.

Sprint backlog : List of committed stories by Dev/QA for specific sprint.

Scrum meeting : Meating conducted by Scrum Master everyday 15 mins. Called as scrum call/Standup meeting.

What did you do yesterday?

What will you do today?

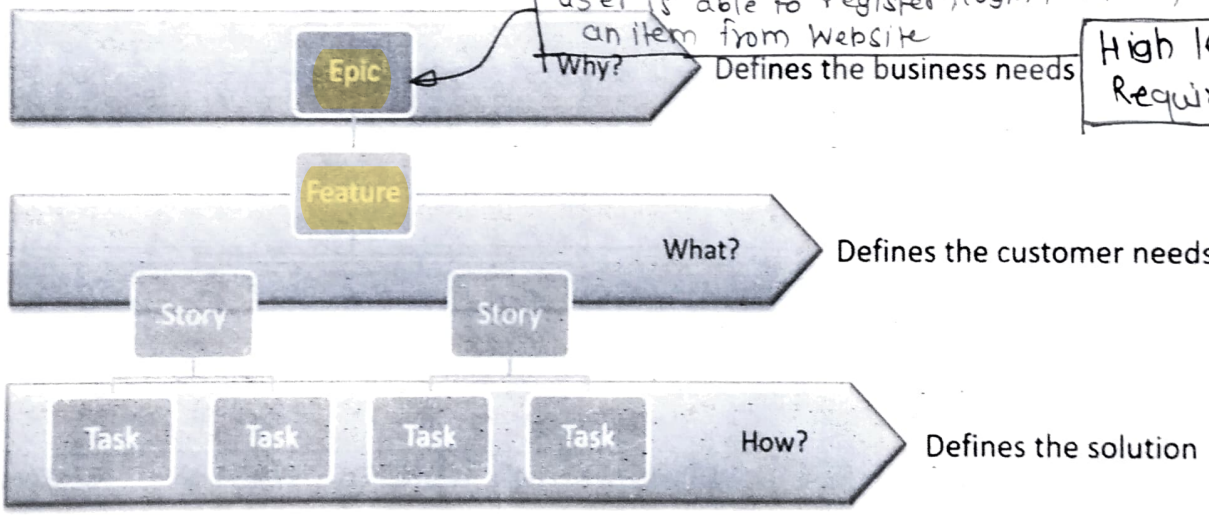
Are there any impediments in your way?

Agenda of this meeting "What is progress of sprint or project"

cheer person scrum master

Epic, Story & Task

* To build Hospital with 100 Beds
* To build hostel
* For new e-commerce website to launch the highest business value will be when a new user is able to register, login, search, and purchase an item from website



Epic: High level(Huge Requirement) collection of user Stories

Ex: As a user I need to use Online Banking Apllication

Story1: As a user I can login into online baking application

Story2: As a user I can check my balance in my account

--

--

Sprint planning meeting :- In this meeting decided that how much stories to take into Sprint by P.A and dev but priority is decided by product owner.

* User stories → its are requirement from stackholder sometime product owner instruct to dev or project manager to make user stories.
* it contain description and acceptenle criteria
* ID of user stories started with US001

Example :-

Name of user story :- credit card Payment

ID :- US001

i.e. US001 : credit card payment

Description : As a customer, I want ability to pay with credit card so that it can confirm my order

Acceptance Criteria :-

1) Discover Card
   When we enter card number, it should display name of company of card i.e. Rupay or master card
2) Validate credit card information
3) Then Validate expiry date & CVV
4) Validate adress
5) Then generate success or failure result.

Story1: As a user I can login into online baking application

Dev tasks

Review the story
Estimate the story
Desing
Code
Unit testing
integration etc....

Task-its action to perform to complete stories.

In agile there is not BA.there is Product owner who is responsible for getting requirement from customers.

Epic-high level ,larger requirement

Story-small requirement

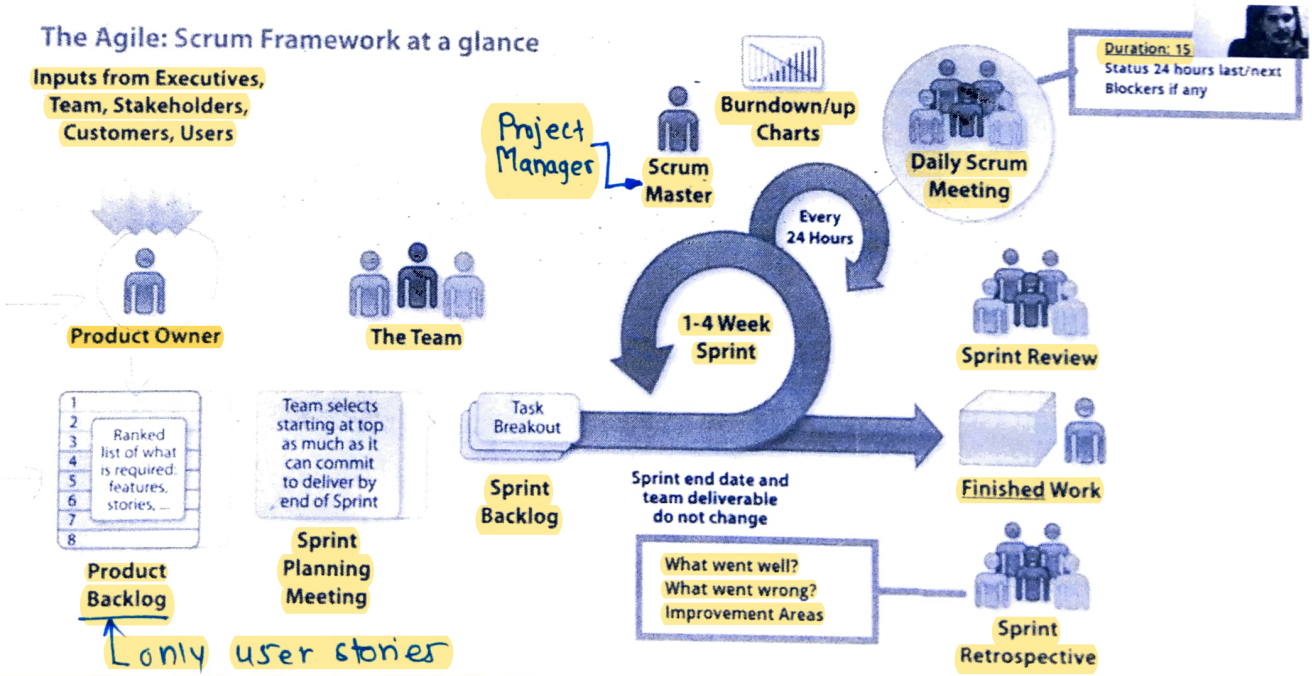
Task-its action performef to complete stories

Sprint retrospective meeting: conducts meeting after completion of sprint. The entire team, including both the ScrumMaster and the product owner should participate.

Story point : Rough estimation of user stories, will be given by Dev & QA in the form of Fibonacci series.

0 1 1 2 3 5 8.....

1 story point = 1 hour/ 1 day(6 hours)



Only user stories

Anyone can add item to product Backlog, PO can take stories from stackholder, customers also he can instruct to development to help hime while creating stories

Waterfall / V-model – 1 releare = 3 month

Product backlog is created by Product owner contain epic and user stories.

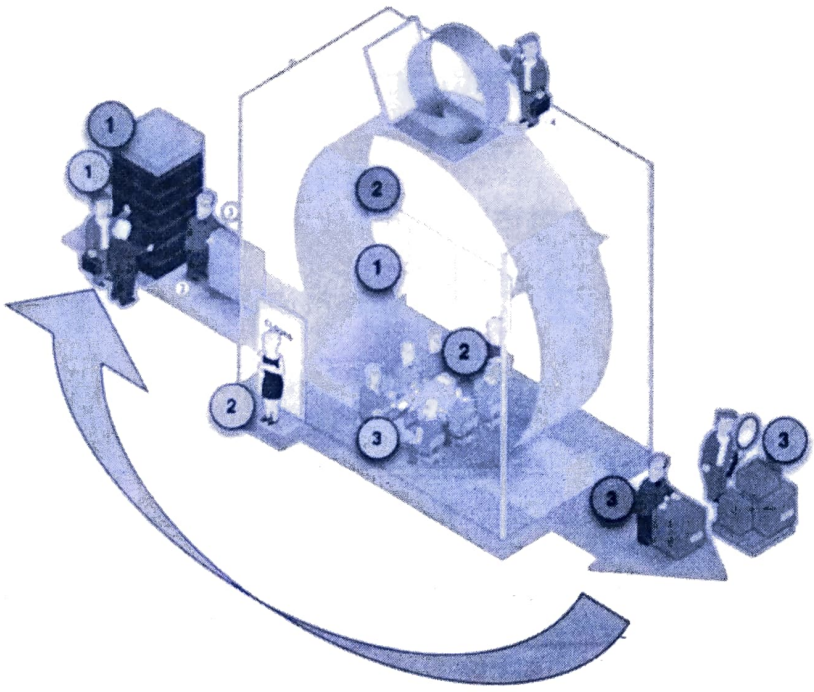
Release period is 3 month /1 month check and remark

Sprint period 2-3 week

SprintBacklogus-its are stories to complete within sprint duration

Sprint backlog prioritized by product owner

And to take stories for developmet in sprint is decided by scrum master of developer also.



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

Artefacts
1. Product Backlog
2. Sprint Backlog
   Burndown chart

Ceremonies
1. Sprint Planning
2. Daily Scrum
3. Sprint Review

Story Points ← its unit of stories

- In most cases a story point uses one of the following scales for sizing:
- Fibonacci sequence: 1,2,3,5,8,13,21

• Sprint duration is depends on story points.
• We can give it into hrs and day • We can give in jira for particulare story

Story point estimation done in sprint planning meeting.

Product backlog-it's a document created by product owner which contain requirement from customer-stories and epic

Sprint Backlogs-it contains committed stories in sprint planning. story inside sprint backlog choose by qa and dv.but priority is set by product owner. suppose there are more feature in sprint planning then we can reduce feature by convensing to product owner in sprint meeting.we can said its not possible to accommodate all features within short period. But once we plane sprint but we cant make all feature as we have commited due to some challenges ,in such situation QA Dev all have to jstify reasons ,why they are not able to develop stories.And remaining stories add in next s print.

If team is not able to complete sprint stories within commited time then team have to give reason for that to product owner.

For every story dev and qa perform their task

Tester task-Understand user stories

-find out high level testing scenario and write test cases for each scenario by studing stories

-Make test planning report

-Make testing environment to be ready etc.

Scrum Board

Scrum board its show progress of stories.

Standup call-sometimes PO may be attend meeting

Once done dev and testing then we have to give demo to product owner before software relese generally demo is given by qa to product owner.

After releaseing software there is one meetimg is now as retrospective meeting.what we done wrong and which area we have to improve.

Definition of Ready (DoR) & Definition of Done (DoD)

Definition of Ready (DoR)

- User Story is clear
- User Story is testable
- User Story is feasible
- User Story defined
- User Story Acceptance Criteria defined
- User Story dependencies identified
- User Story sized by Development Team
- Scrum Team accepts User Experience artefacts
- Performance criteria identified, where appropriate
- Team has a good idea what it will mean to Demo the User Story

Definition of Done (DoD)

- Code produced (all 'to do' items in code completed)
- Code commented, checked in and run against current version in source control
- Peer reviewed (or produced with pair programming) and meeting development standards
- Builds without errors
- Unit tests written and passing
- Deployed to system test environment and passed system tests
- Passed UAT (User Acceptance Testing) and signed off as meeting requirements
- Any build / deployment / configuration changes are implemented / documented / communicated
- Relevant documentation / diagrams produced and / or updated
- Remaining hours for task set to zero and task closed

Definition of Ready (DoR) vs. Definition of Done (DoD)

- **Definition of Ready (DoR)**
- User Story is clear
- User Story is testable
- User Story is feasible
- User Story defined
- User Story Acceptance Criteria defined
- User Story dependencies identified
- User Story sized by Development Team
- Scrum Team accepts User Experience artefacts
- Performance criteria identified, where appropriate
- Scalability criteria identified, where appropriate
- Security criteria identified, where appropriate
- Person who will accept the User Story is identified
- Team has a good idea what it will mean to Demo the User Story

- **Sample Definition of Done**
- Code produced (all 'to do' items in code completed)
- Code commented, checked in and run against current version in source control
- Peer reviewed (or produced with pair programming) and meeting development standards
- Builds without errors
- Unit tests written and passing
- Deployed to system test environment and passed system tests
- Passed UAT (User Acceptance Testing) and signed off as meeting requirements
- Any build / deployment / configuration changes are implemented / documented / communicated
- Relevant documentation / diagrams produced and / or updated
- Remaining hours for task set to zero and task closed

Advantages of Agile Scrum

- We can save time and cost of the project.
- The quality can be ensured because each and every sprint will be tested multiple times.
- The requirements change can be accepted at any level of the project maintenance.
- All are participating in Scrum meting so that transparency can be maintained.
- Each and every sprint we are delivering to the client so we can maintain the customer's satisfaction and we can avoid delivery risk of the project.

Sample Product Backlog

Project Name	OpenCart (Frontend)
Client	OpenCart
Created By	Name of the Product Owner
Creation Date	DD-MM-YYYY
Approval Date	DD-MM-YYYY

Epic	User Story ID	Feature/Title	User Story	Status	Acceptance Creteria
OpenCart_Epic_001 : **For a new e-commerce website to launch, the highest Business Value will be when a new user is able to buy an item from the website.**	US001	Registration	As a First-time visitor to the e-commerce website, I want to register my account, So that I can login to application.	New	New user should able to Register account with valid data.
	US002	Login	As a registered user, I want to login to the website, So that I can see my account details etc..	New	System must validate user credentials and allow login if credentials are correct.
	US003	Logout	As a registered user, I want to logout from website, So that no one else can't access my account.	New	System must logout after login.
	US004	User search products	As a user, I want to be able to search items, So that I can add them to cart and do payment.	New	User should able to search products and add them to cart.

Small Project

Project	Dealmart.com
Stake holders /Customers/users	abc, xyz etc..
Created By	Name of the Product Owner

Epic	User Story ID	Feature/Title	User Story	Status	Acceptance Cr
For a new e-commerce website to launch, the highest Business Value will be when a new user is able to buy an item from the website.	US001	First time visitor can make the payment as a guest user without having to register	As a First-time visitor to the e-commerce website, I want to be able to buy a listed product, So that I can use the product I buy	New	
	US002	New user can register on the website	As a First time visitor to the website, I want to be able to register on the website, So that I can browse and buy listed products from the website	New	
	US003	Registered User can Browse items listed	As a Registered Website User, I want to be able to browse listed products from the website, So that I can make my choice and buy a listed product from the website.	New	
	US004	Registered User can add items to the cart	As a Registered Website User, I want to be able to add items to the cart, So that I can buy a listed product from the website	New	
	US005	Registered User can make payment for the items added to the cart.	As a Registered Website User, I want to be able to buy listed products, So that I can use the product I buy	New	

Sprint_Backlogs

Project	Dealmart.com
Srum Team	Dev, QA, PO, Scrum master, DecOps etc..
Created By	Name of the Product Owner

Epic	User Story ID	Feature/Title	User Story	Story Points	Sprint
For a new e-commerce website to launch, the highest Business Value will be when a new user is able to buy an item from the website.	US001	First time visitor can make the payments as a guest user without having to register	As a First-time visitor to the e-commerce website, I want to be able to buy a listed product, So that I can use the product I buy	8	1
	US002	New user can register on the website	As a First time visitor to the website, I want to be able to register on the website, So that I can browse and buy listed products from the website	3	2
	US003	Registered User can Browse items listed	As a Registered Website User, I want to be able to browse listed products from the website, So that I can make my choice and buy a listed product from the website	5	2
	US004	Registered User can add items to the cart	As a Registered Website User, I want to be able to add items to the cart, So that I can buy a listed product from the website	5	3
	US005	Registered User can make payment for the items added to the cart	As a Registered Website User, I want to be able to buy listed products, So that I can use the product I buy	8	3

Task

QA Tasks	Dev Tasks
Reviewing User stories	Reviewing User stories
Creating Test cases	Setting up Dev Environment
Reviewing Test cases	Developing Programs
Setting up Test Environment	Code Review
Creating test data	Creating Unit Test cases
Executing Manual tests	Executing Unit tests
Automation Setup	Merging Code
Automation Design	Creating builds
Automating test cases	Fixing Bugs
Re-testing Bugs	
Regression testing	
Demo user story	

SELENIUM FRAMEWORK

Basically framework is nothing but systematic and sequential way to write and automate the script for the application. And it is a structural way for building and automating the application.

So, we are having **Hybrid driven framework in our project**, Basically As I am Automation testing engineer, involving in the testing of Automation.

So basically we are using a framework that is **Test Data Driven (TDD)** for using TDD approach we need to take a help of **Maven Project.**

In this we are Having **project Object Model (POM)** where we can have well design structure, where we contain most of the things like src/main/java, src/test/java, src/main/resources, src/test/resources and we will get out put folder and target folder we can create different folder as per our need and lots of things is there.

In this we can add dependencies into the maven project, dependencies are nothing but it is third party tools which we are using in our project.

And inside that we use **page object model (POM)** we have maintained a class for every web page each web page has a separate class and that class holds the functionality and member of that web page.

We have separate packages for Pages and Tests. All the web page related classes come under the Pages package and all the tests related classes come under Test's package Separate classes for every individual test.

As per the maven project all the tests are kept in the src/test/java.

And by using **page object model design patterns with page factory** in that POM uses **Encapsulation concept** feature of oop's.

where **variable are private** and **methods are public** and we use **getter** and **setter** method .but there is a disadvantage of POM that it cannot find the hidden element so we use **Page factory class** to overcome disadvantage of pom.

In page factory we use @findby annotation above your web element and we have to use static method like initelement to initialize the data member in page factory

Inside our project we are using TestNG framework for generating the results. And most of the selenium users use TestNG more that Junit because of its advantages like, **generating the report in a proper format including a no. of test cases, run, passed, failed, and skipped test cases.**

And multiple test cases can be grouped easily by converting them into Testing.xml file its also known as **suit/regression suit.**

With the help of **TestNG Annotations** we can control the flow of method in the test script they make selenium test script more manageable and effective. And its define before every methods .

@Test annotation in testNG methods annoted by @Test it is a part of test case, we can use @Test annotation in single testNG file for multiple times.

And the sequence of annotation in our script like @BeforeSuit,@Beforetest,@BeforeClass,@BeforeMethod,@Test,@afterMethod,@AfterClass,@AfterTest,@AfterSuit.

There are some usefull testNG Keywords.

In which we can set the priority for test cases, which test case should execute first and which is last by using Priority Keyword.

And we can execute test case multiple times using Invocation count keyword. If we want to skip some method we can use enabled keyword with false value.@Test(enabled=false)

If one the test case is taking a long time due to which other test cases are failing. To overcome that situation we use Timeout keyword with specific time.@Test(Timeout=60000)

If one the method is depending on other method then we use depends on method keyword.

As I say we can create testNG test suit or regression suite. It is collection of test cases, testNG enables you to run test method, test classes and test cases in parallel inside the project by performing **parallel execution** .And we can reduce the execution time as test are started.

Also **we can run multiple classes/packages in single xml file** by creating suite it contains configuration , suite name, class name, test names, also we can select thread count which is used for parallel execution based on the no of suite it will execute in sequential order.

Also testNG allows you to perform **Grouping** of different methods.

And we can include as well as exclude that group in xml file and we can also specify a group within another group called **meta group**.

With the help of **testNG assertion** we can compare between Actual results to expected results. There are two types of assertion first is Hard assertion and second is Soft assertion.

Hard assertion: When any test case is fail then hard assertion throw an assert error immediately and break the execution. various assert methods in that like AssertEqual,AssertNotEqual,Asserttrue,Assertfalse,AssertNull,AssertNotNull,

Soft assertion: Sometimes we want to execute whole script even if test fail, so this is not possible in hard assertion to overcome this problem we need to use soft assertion in testNG we have to include its corresponding class (as softAssert()) in the script and end of the @test we will use Assertall() method.

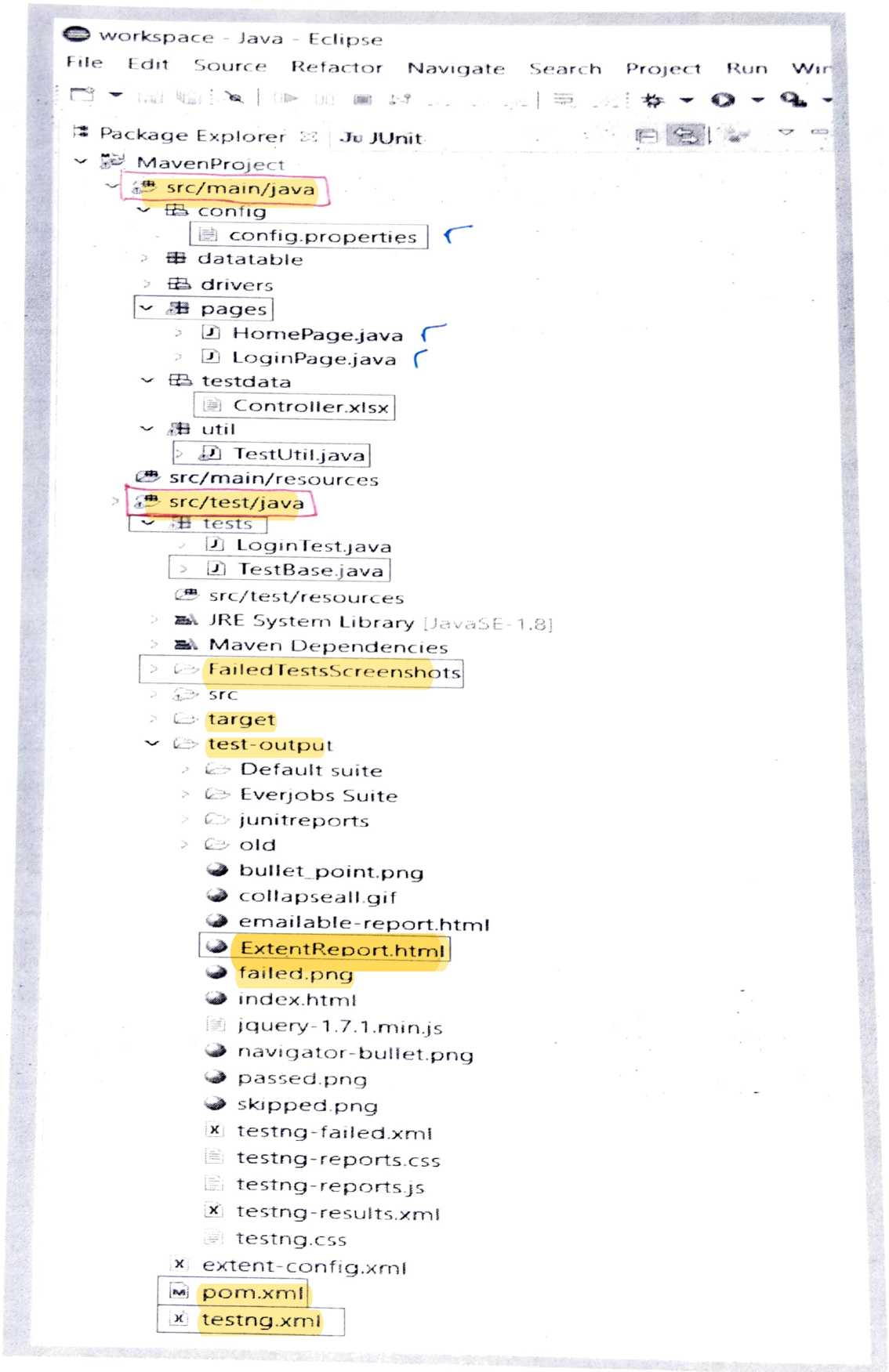
Inside our project to customise logs or reports of testNG we use **testNG Listeners** we can fully customize log using listener .

Listener is defined as a interface which is used in selenium by implementing listener interface.

we can implement interface i.e ITestListener which has method like onstart(),Ontestsuccess(),OntestFailure(),OnTestskipped(),OnFinish().

And it can be implement on **class level as well as suit level.**

Test automation framework structure such as



The above screenshot illustrates a standardized maven project. As per the above maven project, all the tests are kept in the 'src/test/java' and remaining files (such as config.properties, element locators (POM classes), utility files, test data, etc.,) kept under 'src/main/java'.

Test Base Class: Test Base class (TestBase.java) deals with all the common functions used by all the pages. This class is responsible for loading the configurations from properties files, Initializing the Web Driver, Implicit Waits, Extent Reports, and also to create the object of FileInputStream which is responsible for pointing towards the file from which the data should be read.

Utility Class (AKA Functions Class): Utility class (TestUtil.java) stores and handles the functions (The code which is repetitive in nature such as waits, actions, capturing screenshots, accessing excels, sending email, etc.,) which can be commonly used across the entire framework. The reason behind creating a utility class is to achieve reusability. This class extends the TestBase class to inherit the properties of TestBase in TestUtil.

Properties file: This file (config.properties) stores the information that remains static throughout the framework such as browser-specific information, application URL, screenshots path, etc.

All the details which change as per the environment and authorization such as URL, Login Credentials are kept in the config.properties file. Keeping these details in a separate file makes it easy to maintain.

Screenshots: Screenshots will be captured and stored in a separate folder and also the screenshots of failed test cases will be added to the extent reports.

Test Data: All the historical test data will be kept in an excel sheet (controller.xlsx). By using 'controller.xlsx', we pass test data and handle data-driven testing. We use Apache POI to handle excel sheets.

TestNG: Using TestNG for Assertions, Grouping and Parallel execution.

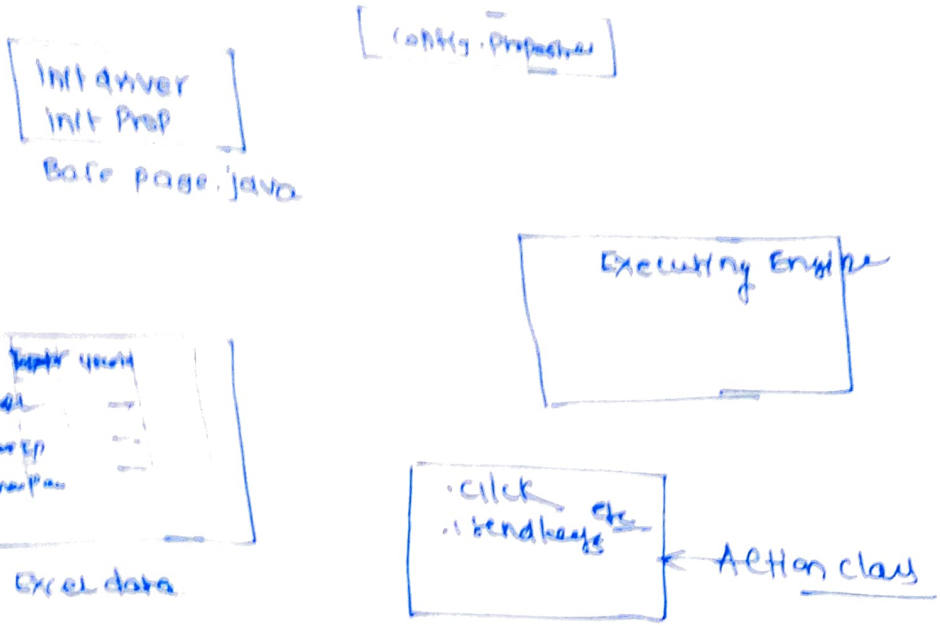
Maven: Using Maven for build, execution, and dependency purpose. Integrating the TestNG dependency in the POM.xml file and running this POM.xml file using Jenkins.

Version Control Tool: We use Git as a repository to store our test scripts.

Jenkins: By using Jenkins CI (Continuous Integration) Tool, we execute test cases dailyand also for nightly execution based on the schedule. Test Results will be sent to the peers using Jenkins.

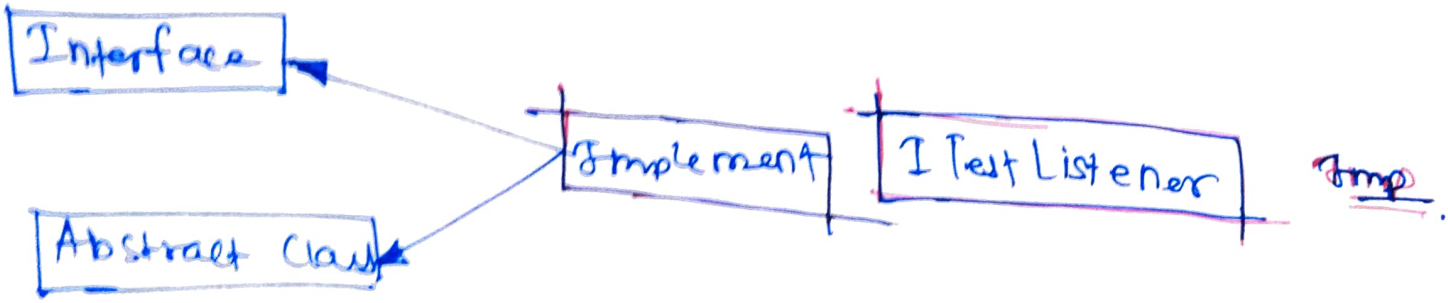
Extent Reports: For the reporting purpose, we are using Extent Reports. It generates beautiful HTML reports. We use the extent reports for maintaining logs and to include the screenshots of failed test cases in the Extent Report.

Keyword driven framework



data hidding

Upcasting WebDriver driver = new ChromeDriver();



==========

Accenture QA Automation interview real time que.

- **Can you brief me about yourself?**
  Hi, my name is Pankaj.

  I started my career as a Testing Executive 4.5 years back with Infosys currently I am working as Test Engineer.
  My responsibility is to understand Business Requirement Specification and High-Level scenarios and to convert them into test cases & Automation scripts if required.
  Execution of test cases and reporting of defect to the developer if there any and get them fixed.
  I have experience on Functional, Automation, Regression, Smoke, Sanity, Web accessibility, Web Analytics, Mobile Testing.

  In my previous project I have worked on Automation testing where we have used Selenium with java and TestNG Cucumber framework for BDD approach. We have used Page object model where we have separated our test cases with page objects, and we performed testing on the same. For build management tool we are using Maven for version controlling we are using Git and for automating our jobs for nightly run or any schedule we are using Jenkins,.

  For defect management & test case management we have used JIRA, TEST RAIL & HP ALM.
  I have worked on tools like BrowseStack, DeviceAnywhere, Toadsql,

  I am working on Agile environment we have daily standup call and we have 2-week sprint cycle. I am part of 8-member team out of which we are 3-Tester, 2- dev, 1- manager, 1-scrum master.

- Tell me your Day to Day activities as QA?

  First thing I do after login in my system. I check the active sprint in Jira for our project code. There I can see my assigned open tasks. After that I will check my mail if there is any important mail I need to take action on. Then we have our daily scrum meeting where we used to tell our previous day actions what we did, what we are planning for today and if we have any blocker to discuss. Product owner and scrum master help us to resolve that blocker. After that I need to take the pending task and do needed action whether creating test case, Execution, Defect retesting if any.

- **Do you have created framework from scratch, or you have maintained that?**
  I have not created Framework from scratch by myself but yes, I was part of framework creation and created some part of it.

- **How much you rate yourself in Java out of 10?**
  Out of 10 I will rate myself 6 in java as QA Automation engineer.

- **Can you tell me Oops concepts and relate it with your Framework?**
  We have Polymorphism, Inheritance, Encapsulation and Abstraction in Oops. So, we will start with
  1) **DATA ABSTRACTION**

Overloading

Assert.assertEqual(expected, Actual);

Assert.assertEqual(expected, Actual, text);

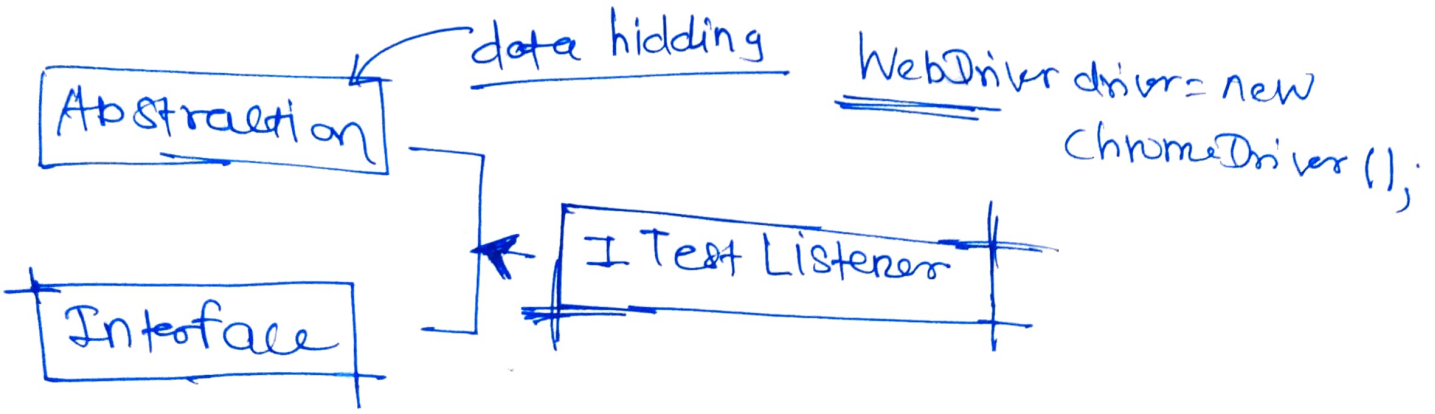
Assert.assertEqual(int Actual, int Expected);

Overriding :->

- driver.findElements(By.xpath "_____");

Encapsulation :-> Page Object model

page layer



WebDriver driver = new ChromeDriver();

Selenium 4 — Oct 2021 — Release

Use → Dec 2021

Data Abstraction means to handle complexity by hiding unnecessary details from the user. In java, abstraction is achieved by interfaces and abstract classes. We can achieve 100% abstraction using interfaces.

In Selenium, WebDriver itself acts as an interface. Consider the below statement:

WebDriver driver = new ChromeDriver();

We initialize the Chrome Browser using Selenium Webdriver. It means we are creating a reference variable (driver) of the interface (WebDriver) and creating an Object. Here WebDriver is an Interface and ChromeDriver is a class.

We can apply Data Abstraction in a Selenium framework by using the Page Object Model design pattern. We define all our locators and their methods in the page class. We can use these locators in our tests but we cannot see the implementation of their underlying methods. So we only show the locators in the tests but hide the implementation. This is a simple example of how we can use Data Abstraction in our Automation Framework.

ITestListener — Implementation of interface

Page Object Model design pattern — Imp

2) ENCAPSULATION — Pom

Encapsulation is defined as the wrapping up of data under a single unit. It is the mechanism that binds together code and the data it manipulates. Encapsulation can be achieved by: Declaring all the variables in the class as private and writing public methods in the class to set and get the values of variables.

All the classes in an Automation Framework are an example of Encapsulation. In Page Object Model classes, we declare the data members using @FindBy and initialization of data members will be done using Constructor to utilize those in methods.

3) INHERITANCE

Inheritance is the mechanism in java by which one class is allowed to inherit the features (fields and methods) of another class.

We can apply Inheritance in our Automation Framework by creating a Base Class to initialize the WebDriver interface, browsers, waits, reports, logging, etc. and then we can extend this Base Class and its methods in other classes like Tests or Utilities. This is a simple example of how we can apply Inheritance in our framework.

Imp — in starting hooks we use Interface

4) POLYMORPHISM

Polymorphism allows us to perform a single action in different ways. In Java polymorphism can be achieved by two ways:

- **Method Overloading**: When there are multiple methods with same name but different parameters then these methods are said to be overloaded. Methods can be overloaded by change in number of arguments or/and change in type of arguments.

In Selenium Automation, Implicit wait is an example of Method Overloading. In Implicit wait we use different time stamps such as SECONDS, MINUTES, HOURS etc.

- **Method Overriding**: It occurs when a derived class has a definition for one of the member functions of the base class. That base function is said to be overridden.

In Selenium Automation, Method Overriding can be achieved by overriding any WebDriver method. For example, we can override the findElement method

findElement — Imp

In assertion we have used overload because in assertion we used to like asset.true(actual, expected) and second time we can use same assert.true(actual, expected, message).

Imp

- **How can you use interface and how it is different from Abstract class?**

Abstract class may have Abstract and concrete methods, and there is not any compulsion in adding abstract method in abstract class. But in Interface, we do have only abstract methods and we don't need to write abstract keyword in Interface this is by default public and abstract.

Imp

Variable are static & final

- **What do you mean by Static keyword in Java?** data Management
  Static means it is at class level not at instance level, we have static method, static variable & static inner class. When we have any variable as static so it will remain same for all the instance of our classes, and static/Private/Final methods can't be over-ridden like if we have initialized any method as Static so we cannot override it in any child class.

- **How to call static method and variable in java?**

  Direct calling, Calling by class name.

- **Can I access Static method by using object reference?**
  Yes we can, but we got one warning that you need to access it via Direct or By class name.

- **How to call non-static method and variable in java?**
  For calling non static method we need to create object first.

- **Can we overload & override main method?**
  Overload-Yes, Override-No

- **What do you mean by wrapper class and how will you do data conversion?**
  Wrapper class in java are used for data conversion. In data conversion if user wants to convert Int to string, String to int, Boolean, double then we use Wrapper class.
  integer.parseInt(); - To convert string to Integer
  Double.parseDouble(); - To convert string to Double
  Boolean.parse Boolean(); - To convert string to Boolean
  String.valueof(); - To convert Integer to String.

- **Can you convert string a ="110a" in integer?**
  No we got NumberFormatException while converting the above string.

- **What do you mean by Call by Value & Call by Reference in Java?**
  Call by value means suppose we have created one sum method with input parameter int a, int b. So while calling the creating the object and running we provide values that is know as call by value.

- **What do you mean by Exceptions in Java?**
  Exception is like any interruption in our normal flow. Like if we are running anything and we got issues in our script this is we called exception, we have 2 types of exception Run Time & Compile Time. (checked & Unchecked exceptions)

- **Can you tell me about difference between Throw and Throws keyword?**
  Throw is a keyword used inside a body of function. And Throws used while initializing any method. By using Throw we can throw only one exception while for Throws we can declare multiple exceptions which might occur in that particular function. Throws keyword followed by instance name and Throw keyword is followed by class name of that exception.

- **How much you rate yourself in selenium out of 5?**
  Out of 5 I will rate myself 3.5 in selenium.

- **Which locator you are using in your framework and why?**
  Mostly we used ID and Xpath because Id is the fastest and unique one and after that we prefer Xpath. Anyways we have other locators as well like css , class name, tag name, Link text, Partial Link text.

- **What is the difference between findelement & findelements?**
  findelement will give the first appearance of that element which matches our locator, whereas findelements will give us list of all the elements which is present over the webpage and matching our locator. And if we don't find the element findelement will give us nosuchelementexception whereas findelements will return NULL/Empty list.

- **Can you tell me how you will handle multiple window in selenium.**
  We have windowhandle & windowhandles function for handling Multiple windows. Windowhandle will give the string value of only the active window that is open whereas windowhandles will give set of all the windows that are open in browser.

- **How you will move from one window to another?**
  First we will check what all windows are open by using driver.getwindowhandles, to get set of opened windows , then I use iterator to iterate over each of the pages and inside for loop will check like Current URL matches with the excepted page, if match then switch to that window by using driver.switchTo(Destination window) -> to return back to main parent window use driver.defaultwindow.

- **Tell me the difference between Implicit & Explicit wait?**
  Implicit wait applies for all the elements and all the tests like if we give 10 sec of implicit wait it will wait for 10 sec for each element before giving nosuchelement exceptions.
  While Explicit wait can be applied for any particular step for which you want extra wait time so we can use explicit wait. We can use mix of both waits to depend on the situation of the step.

- **Can you tell me some exceptions in selenium?**
  NoSuchElementException, NoSuchWindowException NoSuchframeException, StaleElementReferenceException, TimeoutException.

- **Can you tell me about StaleElementReferenceException?**
  Stale means old or decayed, here it sounds like element which was present on that page is no longer there or decayed. To handle this, we can refresh the webpage before pointing to that element. We can write script for waiting via explicit wait by writing expected condition.refresh. Or we can go with page object model in that we can over come this stale element exception.

- **What do you mean by User Defined Exception?**
  User Defined Exception or custom exception is creating your own exception class and throws that exception using 'throw' keyword. This can be done by extending the class Exception. ... The keyword "throw" is used to create a new Exception and throw it to the catch block.

- **Can you tell me what is assert in TestNG?**
  Assert is like verification where we check like expected thing and actual thing are same or not.