Where you used oops concept in your framework?

1. Inheritance:

-Selenium architecture

-Base class method inherited into test class and POM

2. Polymorphism:

Method overloading:

* Iframe
* Collection -> add() method
* Implicit wait

3. Encapsulation:

POM Class

3. Generalization:

-> Collection

-> Selenium Archiecture -> browser implementation classes

**Interface**

SarchContext

WebDriver

WebElement

TakeScreenshot

JavaScriptExecutor

Collection: List, set , Queue

ITestReuslt

Iterator

Enumarator

Alert

Action -> interface

Static method:

**Selenium:**

All methods in “By” class -> id, class, xpath, tagName, css etc

initElementMethod-> Pagefactory

**ApchePoi:**

all methods in Workfactory class -> create, getSheet, getRow(), getCell(), getStringCellvalue(),

**Java:**

sleep -> Thread

setProperty -> System

Copy -> FileHandler

**TestNG:**

All methods in Assert class (HardAssert) -> assertEqual, assertNotEqual etc.

Our framework:

takeScreenshot () -> Utility

readExcelsheetData () -> Utility

**Static variable:**

Configuration class -> appUrl, driverPath, screenshtpath, excelsheetPath etc.

**Non-static methods:**

* findElement()
* findElements()
* all methods in select class (selectByVisibleText, selectByIndex, selectByValue)
* WebDriver method ex: close, quit, maximize, minimize, etc
* WebElement Methods
* Methods in SoftAssert (ex: assertEqual, assertNotEqual, assertTrue, assertFalse, assertNull etc.)
* Methods in Action class
* Method in POM class
* Method Implementation class (ArrayList, vector, LinkedList,TreeSet etc)
* Methods in Date class

**Classes:**

**Java:**

Date

System

FileHandler

Thread

Exceptions

FileInputStream

Scanner

ArraList, Vector, LinkedList, Hashset, LinkedHashset,TreeSet

HashMap

ArithmeticException, NullPointerException, ArraIndexOutofBounExcpetion

File

**Selenium Lib:**

By

Pagefactory

Select

Actions

RemoteWebDriver

ExplictWait

WebDriverWait

Point

Dimension

ChromeDriver, FirefixDriver etc.

NoSuchElementException, NoSuchFrameException, TImeOutException, StaleElementExcpetion

Keys

**ApchePoi:**

Workbookfactory

**TestNG:**

Assert

SoftAssert

**Our Framework:**

BaseClass

POM

Configurations

Utility

TestClass

Wrapper

Exception topmost -> Throwable

Collection -> Iterable

Java – object

Mutable and immutable String

Team Structure:

|  |  |
| --- | --- |
| Team-A | Team-B |
| Purely Tester | Developer + Tester = 12 |
| Manual Tester: 5 to 6 | 8 Developer (BA, scrum master, po) |
| Automation Tester: 2 to 3 | 4 tester (Manual + Automation) |

Daily Routine:

* Daily standup / daily call / scrum call
* Report analysis

Failure reasons:

1. Internal Fix

2. Test Data

3. Defect : report to respective respective team

* Start the new scope automation

Roles and Responsibility of automation Engg. :

1. Identify automatable test case :
2. Understand flow of automatable test.
3. Prepare test data as per framework requirement
4. Identify locator
5. Write POM Class
6. Write test class as per flow
7. Make sure automation script is working fine and test case is Pass.
8. Make sure proper report is generated is generated.
9. Push code on remote repository and create Pull Request.
10. Create test suite

Have you designed any framework?

No, I have worked on already designed framework.

Two types of roles

Pure automation: get the manual test cases and automate it

Manual + automation: Design and execute manual test case and Automate it.

What are challenges you are facing in automation:

1. Locator change for dynamic element
2. Dynamic element need to handle in dynamic way
3. Framework understanding is bit challenging
4. Handel unwanted popup is taking some time
5. Exception need to be handle.
6. Cross browser need different browser configuration and version

Sprint : 15 days (2 weeks) – Role: Manual + Automation

1 st Week: Manual TCs Design and Execute

2nd Week : Identify automatable TCs. And write automation script for automatable TCs.

How Many Test case you have automated till now?

2+, 2.5, 3+ Experice:

In technical interview:

Automation is of 06 month for initial 4 to 5 interview => 40 to 50 TCs

-After 4 to 5 interview automation exp is 1 year => 60 to 70 TCs or 70 to 80 TCs

Exp less than 1 or 1.5 or up 2 Yrs

In technical interview:

Automation is of 06 month for initial 4 to 5 interview => 30 to 40 TCs

-After 4 to 5 interview automation exp is 1 year => 50 to 60 TCs or 60 to 70 TCs

How many test cases you can automate in one day?

It depends on complexity of test cases:

Low complexity: 4 to 5 test case (one or 2 web page)

Medium complexity : 2 to 3 test case (it consist of 3 to 4 web pages)

High complexity : 1 per day or some time 1 Test case will take 2days (Highcomplexity: which cosists of more than 4 web pages)

Weekly:

Low complexity: 15 to 20

Medium: 10 to 15

High complexity: 4 to 8

Automation will be generally performed for Regression test cases NOT on SIT test case, because in SIT Tests build is unstable.

* new keyword => Heap area
* If new keyword is NOT used => Stack area

|  |  |
| --- | --- |
| String class | StringBuffer / StringBuilder |
| Immutable => Non changeable behavior | Mutable => changeable behavior |
| When we perform some operation on string then its original behavior of object is not changing is called as immutability of string | When we perform some operation on string then it original behavior will change that original behavior of object is changing is called as mutability of string. |
| More Memory will require as compare to StringBuffer class | Less memory will require as compare to String class |
| Performance is High | Low |
| String pool area | Heap area |

**package** stringProgram;

**public** **class** Program15 {

**public** **static** **void** main(String[] args) {

String s1 = **new** String("Test Program");

System.***out***.println(s1);

}

}

**package** stringProgram;

**public** **class** Program16 {

**public** **static** **void** main(String[] args) {

StringBuffer s2 = **new** StringBuffer("Test Message");

System.***out***.println(s2);

System.***out***.println(s2.append(" with java "));

System.***out***.println(s2);

}

}

Modules: (1%)

1. Login
2. Admin / profile (optional)
3. HomePage / DashBoard
4. [Travel Insurance](https://travel.policybazaar.com/?newpq=1&utm_term=newjourney&utm_content=newpq)
5. Health Insurance
6. Car insurance
7. Group Health Insurance
8. Family Health insurance

Total automated module: 08

Total module in project: 20 to 25

Total POM classes: 50 to 60 POM classes

Test Classes: 50 to 60 or 40 to 50 or 55 to 60

Test Classes: minimum 4 TCs and max 10

Total TCs for 8 Module (Team contribution): around 250 to 300

Extent Report and Email able Report