Agile Explain:

- Agile is an iterative and incremental approach.
- Agile methodology is a value driven process.
- Whatever the client is requiring on a priority we have to make sure that it is deliverable at that instance. Deliverable should not get delay is a main principle.
- We can develop, test and deploy the piece of software to the stakeholder with few features.
- We can accept the frequent requirement changes.
- There is a good communication between product owner, stakeholder, and scrum master and development team.
- Agile defines continues development & continues testing will happen in application/ build.
- If any change request comes at any point of time then we will consider these change request.
- When will accept these CR then we will check impact on current development, Testing & production.
- If change request has more impact on development/ testing/ production then we will inform to client.
- If change request has less impact on development/ testing/ production then we will develop & testing and we will deploy to client.
- In my organization , deployment process in 2 week.
- Agile methodology is having no of subtypes:
    - **XP (extreme programming) –** Development & no testing.
    - **Scrum -** (Bunch of requirement then Sprint wise development & Testing & Delivery).
    - **Kanban –** Support team (tickets/ Existing issue/ bugs/CR).
    - **Lean -** Support team.
    - **FDD –** Future driven development.
- I have worked in Scrum agile methodology.
- It is most popular in IT industry followed by 90% of companies.
- In agile scrum methodology , project team worked on sprint.
- They have to design, develop, test and deploy the software at the end of sprint.
- **It is start with, Requirement gathering** product owner collect all the requirements from the stakeholder. Then make prototype to show them how the software the look like.
- Prototype is nothing but the visualtion of functionality before development. After the approval from client side then team will work on it. As it made with rally tool.
- In requirement gathering product owner converts product backlog document is sprint backlog which means all the high level requirements of stakeholder get converted into user stories.
- This document is share to the team for understand purpose. If they will not clear the requirement they will arrange the meeting with BA.
- In scrum methodology we have no of meeting which are hosted by scrum master.
- Scrum master is person who monitor each and every activity of the project also handled the project.
- In this meeting everybody is involved over there.
- Firstly we have grooming session, in this meeting product owner discuss about all the user in detail then team have to understand all the requirement in detail. If have any doubt discuss with team.

- Then we have sprint planning meeting, in this meeting scrum master will ask to developer as well as tester to give estimation to assign user stories.
- Estimation is based on 3 factors such as knowledge, effort and complexity.
- **Knowledge:**
- Whenever team formation is done, each team member should have knowledge about domain of the product.
- **Efforts:**
- High level management decides how much efforts & resources are required. As per dependency of module. They select user story.
- **Complexity:**
- This is the important factor in Estimation for defining complexity. They consider time, cost, resources requirements, Bandwidth of team.
- They give estimation based on the Fibonacci's series such as 1 story points is equal to 8 hrs. .
- As the sprint start,
- Development team will develop the code that time testing team will find out scenarios as per the requirements or functionality.
- As the development goes in process testing team will conduct review on the test cases such self, peer, internal, external review it depends?
- Every day we have daily standup call or scrum call. In this meeting everybody from the team have to give to work update to the scrum master.
- Here we have to tell what we did yesterday, what you planned for today.
- If any blocker or roadblock in the work discuss with team and get the solution.
- By this time scrum master do prepare the burn down and burn up chart.
- As soon as they received the build from the dev team start executing the test cases.
- If any defect found they will raised in defect management tool such as JIRA.
- Developer will work on assign defect, fix it and sent to the testing team to perform retesting on it.
- Once the testing activity completed they will resume it.
- Then sprint review meeting, we have to give demo to the stakeholder on what we have completed within the sprint.
- They provide review on it.
- If there is any kind of change request is happen by stakeholder then we have to accept it. Team have to work on it.
- Once the development and testing is completed on the change request we have to provide demo again.
- Once they approve it. Then we have deploy to the requirement to them.
- Then they can use this.
- At the end of sprint we have sprint retrospective meeting, it this meeting team will discuss how the sprint went well, what was the mistake that we made if any improvement needed we have to set goal for next sprint.
- If any team member from the team who works well appreciate him.
- That's all about the agile methodology that is followed in my organization.

Framework Explanation:

- Basically framework is nothing but the systematic way to write test scripts to automate the web based application.
- So currently I am a working on the hybrid driven framework in which I am involved in data driven framework.
- As name suggest data driven means executing test cases with the help of test data which is stored in some specific format like table or in excel sheet.
- In data driven framework we have several components such as we are using java as a programming language, for test script writing we are using *Eclipse IDE,* Also for the test data we have separate folder for *excel sheet* because *test data* is in the table format.
- Also we are using *properties file* that stores the information that remain throughout the framework *browser specific information*, *application URL* , *security questions* and *answers , screenshot path , password.*
- Each parameter in the properties file is stored as *pair of strings*, in key-value pair format *(key=value)* where each key is on one line.
- We have another component as reports or log. We are using *extend report* for reporting.
- For maintaining the log we have *log4j API implementation* which maintains logs for test execution.
- For scripting we are using the *maven project* which is used to build and manage the projects.
- If we want to build the java projects *into executable jar files* then we can use maven project. Able to work with multiple project at same time so maven is basically build automation tool.
- In maven we can add *dependencies* from *remotes repository* of maven and stored into local cache according to our requirements. The dependencies are download automatically over the internet.
- In maven project, here we have separate packages for tests and pages. All the web page related classes are comes under the page packages and all the test classes are comes under the test packages.
- Maven is based on page object model. We are using page object model with page factory to avoid *stale_element_reference_Exception.*
- AS we know POM is design pattern to create repository for web UI elements. It has advantage of to reduce code duplication and improves test maintenance. There is a clean separation between test code and specific code.

- In page factory, we are using *@FindBy annotation* to declare the web element. We initialize that web element in the constructor.

- In POM, we have maintained a class for every webpage. Each web page has a separate class and that class holds the functionality and members of that web page. Also we have separate classes for every individual test classes.
- As per the structure of maven project all the test are kept in *src/test/java* and remaining files such properties file, POM classes, test data, utility files are comes under the *src/main/java.*
- POM supports multiple concepts of OOPs in java like
- We are declaring the *webelement as* variable and we declare the variable as *private* and use that variable in *public* method so here we achieve *encapsulation.*
- Also we have *base class* for initialize the browser , waits, properties files etc. and we extends the *base* class so here we achieve *inheritance*.
- Also in *waits* we have time configurations in *seconds, minutes*, and *hours* so here we achieve *polymorphism.*
- Also we using the *web driver* which is *an interface.* Also we perform *upcasting* by creating the *reference variable* of the *web driver* interface as driver and making the object the browser class.
- For test script we are using *testNG* which is inspired by Junit by adding new functionalities which made testNG more powerful than other framework.
- TestNG have some features like we can run the *test cases in groups* according to the requirements, we can *control the execution of test cases*.
- We can executes the test classes or *test cases parallel* by doing modifications *in the xml file.*
- In testNG we have *annotations and keyword.* Annotations *control the line of code* that we have *written in the method.*
- So we are using *@symbol* before the method.
- By using keywords we are executing the test cases *on priority,* we can *dependencies* over the test cases, and also if we *don't want to consider the test case* to execute here we are using *enabled keyword.*

- If any of test case got *failed due to time issue*, if it requires *more time* then we use *timeout,* also if we want to execute the particular test case multiple time then we use *invocation count.*

- TestNG does not require a main method to run and methods written need not be static.

- Also we are *Jenkins tool* for run the test *script or project over remote server*.

- It is basically an open source *continues integration/continuous delivery* and *deployment (CI/CD)* automations *software DevOps tool* written in the Java programming language.

- And for maintaining the source code we are using version control tool such as *Git remote repository.*
- That's All About my framework on which I have worked. Thank You…!