**1.1. Essentials point:**

**API testing**

Test the endpoints:

* Verify correct status code
* Verify response payload
* Verify response headers
* Verify performance sanity

Scenarios like:

* Positive test
* Negative testing with valid input
* Negative testing with invalid input
* Security, authorization and permission test

**Web app testing:**

Validate the new user stories doing functional testing related to acceptance criteria.

* Check all the GUI elements for size, position, width, length, and acceptance of characters or numbers. For instance, you must be able to provide inputs to the input fields.
* Check you can execute the intended functionality of the application using the GUI
* Check Error Messages are displayed correctly
* Check for Clear demarcation of different sections on screen
* Check Font used in an application is readable
* Check the alignment of the text is proper
* Check the Color of the font and warning messages is aesthetically pleasing
* Check that the images have good clarity
* Check that the images are properly aligned
* Check the positioning of GUI elements for different screen resolution.

1.2. Which will be the main challenges for testing in this Project?

The main challenge will be to coordinate the activities to get a fast feedback related to the current developments. We will need test early as possible because the time is short and is extremely important to solve issues as far as possible and in this way take corrects direction and measured were we should improve and maintained the focus.

1.3. How you approach the test Automation strategy in this project?

We must define the goals and plan the test approach to decide the level of testing to provide the maximum value. In point, 1.1 I choose some important items to automate API level and UI level. These tests must assure usability, functionality, reliability, performance and security.

Decide the rules and guidelines to follow like methodology and framework then choose the team members experience with coding and testing, as well as the type of architecture the code is based on.

The next point is select the right testing tools and set the test Automation environment and the frameworks.

Now we can start creating test cases, execute it, and maintain it.

Measure and take decisions.

1.4. How would you approach the integration of Automation test into the Project CI/CD pipelines?

One times we have already setting the Automation environment to develop and to test we can use some CI/CD tool like Team city to create pipelines to deploy the last product version in to the test environment and then run the test Automation there using different roles to identify the different kind of testing and goals for example API and Smoke web Automation and before run obtain details report.

2.1/ 2.2. Witch methodologies would you recommend for this Project development? Why?

I recommend SRUM with sprints of two weeks to get fast feedback and can take decision early.

Functional and Non-fusional testing to bring a good coverage.

BDD can be a good idea to write test case fast and us it to automate directly and to get and receive feedback for the stakeholders.

Related to Automation we can set some patterns to work like Page Object Model Page Factory Model.

3.1. How would configure the test team for this Project?

In this case, we will be seven in QA team members:

Four manual tester 2 in each team.

Two QA Automation Engineer working in both teams focus one in API automation and the other in Wen Automation

And 1 QA Lead helping to the others QA members organizing the tasks and solving problems and showing the progress to the stakeholder (Can help how SCRUM master is if needed).

3.3. Witch mindset and attitudes would you look for in your team members?

* DevOps mindset.
* Goal oriented.
* Good problem solving.
* Capacity to self-organize.
* Enthusiastic.
* Creative.
* Good team player.

4.1 How would you identify the test scenarios or test cases needed?

The QA engineer must have a good understanding of the business and functional requirements of the application to do this we can follow these guidelines:

1. Go through the requirements document carefully and try to understand the same.
2. Do not just guess the requirement! If you need clarifications, talk to the project manager or to any other person who understands the requirement really well.
3. Understand the project workflow and wireframes (if available) and relate the same to the requirement.

And we can used some Test case Design Technique.

Deriving test cases directly from a requirement specification include:

* Boundary Value Analysis (BVA)
* Equivalence Partitioning (EP)
* Decision Table Testing
* State Transition Diagrams
* Use Case Testing
* Corner cases (occurs outside of normal operating parameters)

4.2. How you keep traceability between the test scenarios/cases and requirements?

I can make a Forward traceability matrix to solve the traceability.

4.3. How would you Split the testing responsibilities between developers and Tester?

The developers can make the unit tests and help to the QA collages to create the pipelines and the functionality to deploy the product in different environment and work together in the entire DLC sharing ideas and knowledge.

The Tester have the responsibility to analyze the requirement (User Stories) to find issues and missing information from the beginning. In this way, the developers have a good documentation to start theirs activities.

5.1. Witch tools would you use for test management.

I would use Jira/Zephyr or VSTS.

5.2. Witch tools would you use for Performance Testing?

I choose Jmeter.

5.3. Witch tool/frameworks would you use for API testing and API Automation?

I choose Postman to manual testing and Rest-assured to Automation.

5.5. Witch others tools would you use?

* We can use Framework like Serenity that could improve our speed to start the automation.
* Sourcetree or Bitbucket to manage the code.
* Confluence to manage the SRUM board and documentation.
* Eclipse or IntelliJ.
* Planning Poker.
* Slack to meetings and chat.