Position: Quality Assurance Engineer

Appliance: Smart Vacuum Cleaner in HACS

1. Introduction

In the HACS project, my role was that of a Tester (Quality Assurance Engineer). My main responsibility was to make sure that the Smart Vacuum Cleaner module worked correctly according to the requirements and the design. While the developer writes the code, the tester checks if the system is working as expected and also ensures that the system does not fail under unusual conditions.

The smart vacuum cleaner is a very useful appliance, but only if it is reliable. A small bug, such as failing to start cleaning on schedule or not giving an alert when the battery is low, can cause user dissatisfaction. Therefore, testing is an equally important phase of the project, as it ensures quality and correctness before the system is considered complete.

2. Role of Tester (Theory)

In software engineering, testing is defined as the process of verifying and validating that a system meets its requirements and works correctly. The tester plays a vital role by checking the following:

Verification: Does the system implement the requirements properly?

Validation: Does the system behave as the user expects?

Error Detection: Are there any hidden issues that can cause failures?

Quality Assurance: Is the system reliable, safe, and ready for use?

The tester is sometimes called the “user’s representative” because they check the system from the point of view of the end user. The goal is not only to confirm that the system works but also to try and find weaknesses before the user does.

3. Purpose of My Contribution

The purpose of my contribution in the project was to make sure that:

All the features of the Smart Vacuum Cleaner worked correctly.

The appliance met both the functional requirements (like ON/OFF, scheduling, and alerts) and non-functional requirements (like reliability and ease of use).

Problems were identified early so that the developer could fix them quickly.

The system gave the same result every time, without unexpected failures.

Testing was necessary to ensure that the appliance would be useful and trustworthy for the end user.

4. My Work in Testing

To perform my testing work, I followed a systematic approach. I began by reading the requirements written by the analyst. Then I studied the design prepared by the system designer. Finally, I looked at the features implemented by the developer. With this knowledge, I was able to prepare test cases for different scenarios.

The main functions I tested were:

ON and OFF operations – Checking if the vacuum cleaner started and stopped correctly.

Auto Cleaning Mode – Ensuring the vacuum cleaned the entire area without missing steps.

Spot Cleaning Mode – Verifying that only the selected area was cleaned.

Scheduling – Confirming that the vacuum started exactly at the scheduled time.

Pause and Resume – Testing whether cleaning could be paused and restarted smoothly.

Cancel Operation – Checking that the vacuum stopped immediately when cancelled.

Alerts – Testing conditions for low battery and dust bin full.

For each of these, I wrote simple test cases and then ran them to observe the results.

5. Types of Testing Performed

During the project, I carried out several types of testing:

Functional Testing: To check if each function worked as expected.

Boundary Testing: For example, scheduling the vacuum just before midnight and seeing if it still worked correctly.

Negative Testing: Entering invalid inputs, like scheduling in the past, to see how the system handled them.

Integration Testing: Making sure the vacuum cleaner worked properly when connected to the HACS server.

User-Oriented Testing: Looking at the system from the point of view of the end user and checking if it was easy and logical to use.

By using different testing methods, I was able to ensure that the system was robust.

6. Challenges I Faced

Testing was not without difficulties. Some of the challenges I faced included:

Incomplete Requirements: Sometimes requirements were written in general terms, and I had to clarify them with the analyst before writing test cases.

Unexpected Results: In some cases, the system behaved differently from what I expected. I had to carefully check if the issue was in the requirement, the design, or the code.

Multiple States: The vacuum cleaner could be ON, OFF, paused, or cancelled. Testing all possible state changes took extra time and careful planning.

Communication with Developer: When I found bugs, I had to work closely with the developer to explain the issues and re-test after corrections.

These challenges taught me the importance of patience and detailed observation in quality assurance.

7. Team Collaboration

My role as a tester was directly linked to the work of other team members. The analyst’s document guided me in knowing what features had to be tested. The designer’s diagrams helped me understand how the modules interacted. The developer’s work gave me the system to test.

When I found bugs, I reported them to the developer, and after corrections, I re-tested the features. This cycle of testing and fixing improved the system’s quality. The team collaboration made sure that the final product was not only functional but also reliable.

8. Outcome of My Contribution

The outcome of my contribution was a well-tested Smart Vacuum Cleaner module. I verified that the appliance performed correctly in all required situations. For example:

The vacuum started and stopped correctly.

Auto cleaning and spot cleaning worked as expected.

Scheduling started the cleaning at the exact set time.

Pause, resume, and cancel worked without errors.

Alerts were generated when the battery was low or when the dust bin was full.

Because of my work, the system was more dependable, and the risk of failure for the end user was reduced.

9. Learning and Reflection

From this project, I learned that testing is just as important as development. Without testing, even a small mistake can go unnoticed and cause problems later.

I learned about systematic testing methods like functional testing, negative testing, and boundary testing. I also learned that communication with the developer is key, because reporting and fixing bugs requires cooperation.

From a practical point of view, I realized that testing requires patience and attention to detail. Sometimes, repeating the same test multiple times was necessary to confirm results. From a theoretical point of view, I learned that testing is not about proving that the system works but about trying to find errors before the user does.

10. Conclusion

As the Quality Assurance Engineer, my contribution was to ensure that the Smart Vacuum Cleaner module of HACS worked correctly, reliably, and safely. I designed test cases, carried out different types of testing, identified problems, and worked with the developer to fix them.

My role was critical in making sure that the system was not only functional but also trustworthy for the end user. This experience taught me that quality assurance is not optional but essential in any software project. It also showed me how teamwork improves the final result, because only when each member performs their role carefully does the project succeed.