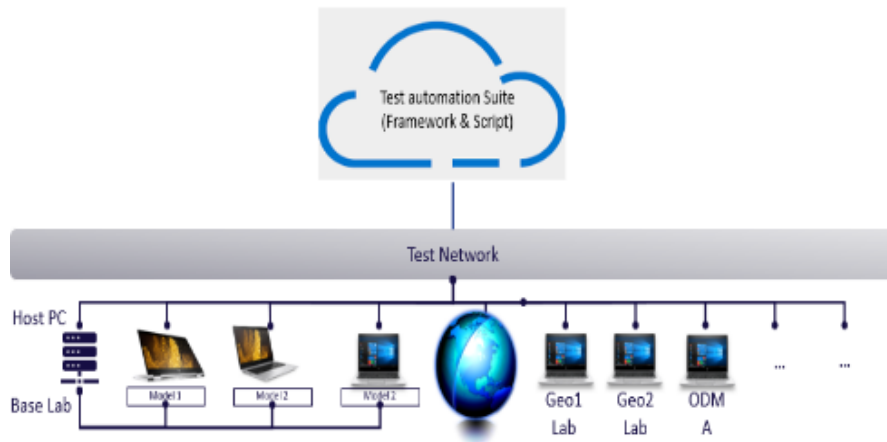




Test Automation Framework for computing devices



BENEFITS

- Enhanced the test execution speed and increased the productivity of the organization
- Flexibility to add new features or technology to the existing solution
- Reusability of common features, functionalities and actions for various applications
- Reduced human error

RESULTS

- Improved the overall product quality.
- Increased productivity thereby better time to market

CUSTOMER CHALLENGES

- Customer wanted to verify the functionalities of various applications/BIOS /Windows driver/performance testing/stress testing etc. in laptops/desktops.
- Test plans available for manual testing across various devices was time consuming.
- Execution of test cases in different OS versions for each release was difficult and often did not meet the release schedule.

SCOPE

Proposed to create a Test Automation Framework that could

- Execute different types of test cases in various machines simultaneously.
- Include reusable components which can be used in other modules.
- Configure various types of test machines
- Efficiently undertake exception handling.
- Integrate to test automation scripts easily

SOLUTION

- Developed a validation automation suite with a vision of comprehensive automation strategy
- Provided a Test Manager component which execute/schedule test jobs to run whenever the test resource or Unit Under Test (UUT) is available
- Created a Test Center which can distribute test jobs to the Test Managers in different test sites.
- Built a common library module which enhances the reusability perspective of the framework
- Created a Utility component and Configuration component which can restart/hibernate/sleep the testing devices and configure the devices for test execution
- Tools & Technologies : Robot framework, Python, Selenium, PywinAuto

FEATURES

- Automation with open source scripting
- Parallel execution on multiple test devices
- Efficient way of Exception handling and Error Logging
- Test cases are independent