Software Verification & Validation | Industry Experience

- Cyient works with top 6 Medical OEMs
- We have experience across all classes of medical devices, I, II and III
- Multi domain experience in Imaging, Monitoring, IVD, Cardiology Supported 200+ programs globally

- Cyient works with 4 top Rail OEMs
- We have “experience across 25+ non-safety & safety critical subsystems for rolling stock and signaling”
- Supported 20+ Rail programs globally and executed 350+ projects
- Subsystems ranging from SIL 1 - 4

- Cyient works with top 4 Tier 1 Automotive companies
- We have “experience across multiple subsystems”
- Supported 20+ programs globally
- Standard – ISO 26262, ASPICE

- Cyient works with **5 top Aero OEMs**
- Experience across **multiple subsystems**
- Established various **safety criticality compliance levels as per DO178 B/C**
- **FAA/EASA/DGCA/CAVIC**
- **80+ SOI/SOF audits**

Problem Statement

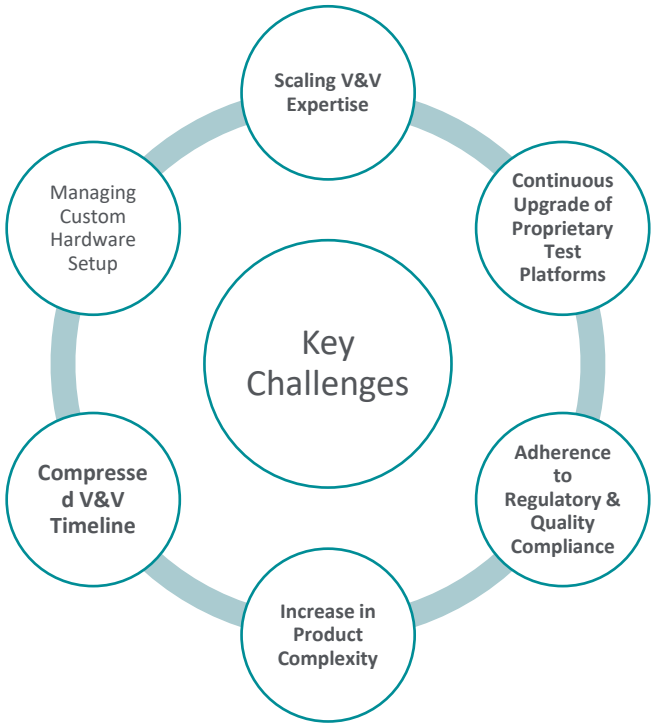
To ensure the software is thoroughly tested and dependable, having lengthy testing cycles, proficient test engineers, and allocating adequate time, effort, and funding has become necessary. Choosing an inappropriate test methodology can have severe repercussions on test efforts, and even minor modifications to the software or configuration can lead to substantial expenses for the organization.

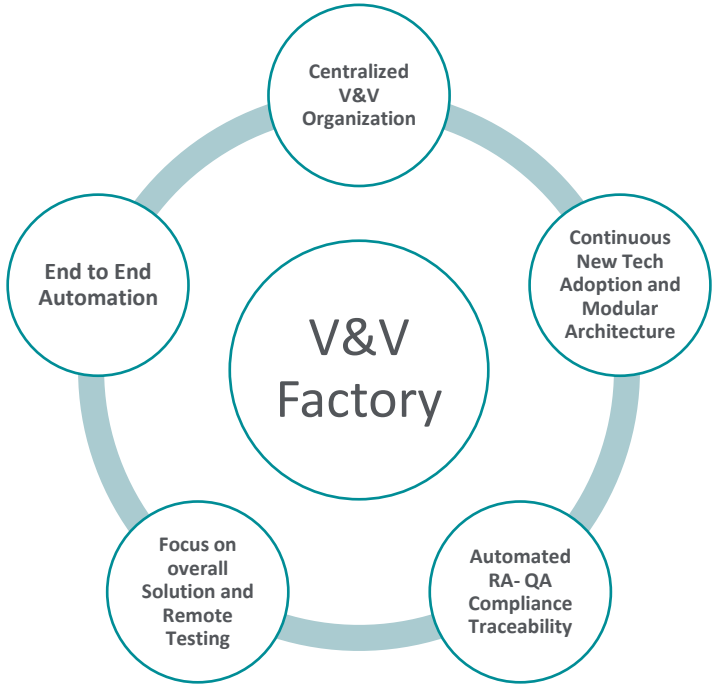
To effectively address these challenges in a sustainable manner, it is crucial to adopt a test automation solution that is efficient, dependable, and user-friendly. Such a solution should have the capability to anticipate, generate, and simulate business use cases with minimal or no human intervention.

The challenges in automated testing are,

- Struggling communication ana collaboration in Team
  - Collaboration between testers and developers is essential as miscommunication and lack of understanding of the automation tool may lead to complications during the testing process.
- Finalizing the Right Automation Tool
  - Choosing the right testing tool from the numerous options available, including free and commercial tools, each suited to particular scenarios and requiring thorough research and budget considerations.
- High Upfront Investment Cost
  - While automation boosts testing speed, it demands substantial initial investment, and the payback period can be uncertain or prolonged.
- Selecting Proper Testing Approach
  - Identifying the right test automation approach to match the needs especially important if you are working in an Agile environment where the tested application changes regularly.
- Demanding Skilled Resources / Scaling V&V Expertise
  - Acquisition struggles of resources with strong programming skills and in-depth knowledge of test automation tools.
- Continuous Upgrade of Proprietary Test Platforms
  - Every now-then new test framework/tools are getting added to solve various testing pain areas
- Adherence to Regulatory & Quality Compliance
  - Aero, Rail, Medical, Auto domains are highly regulated the test tools and product need to be in compliance
- Increase in Product Complexity
  - More commented devices are adding more complexities to product
- Compressed V&V Timeline / Time To Market
  - Competitive market putting pressure on organization to release good quality product in market as early and regularly as possible
- Managing Custom Hardware Setup
  - As product getting more heterogenous and complex need for customer hardware to solve unique problem has becoming more evident

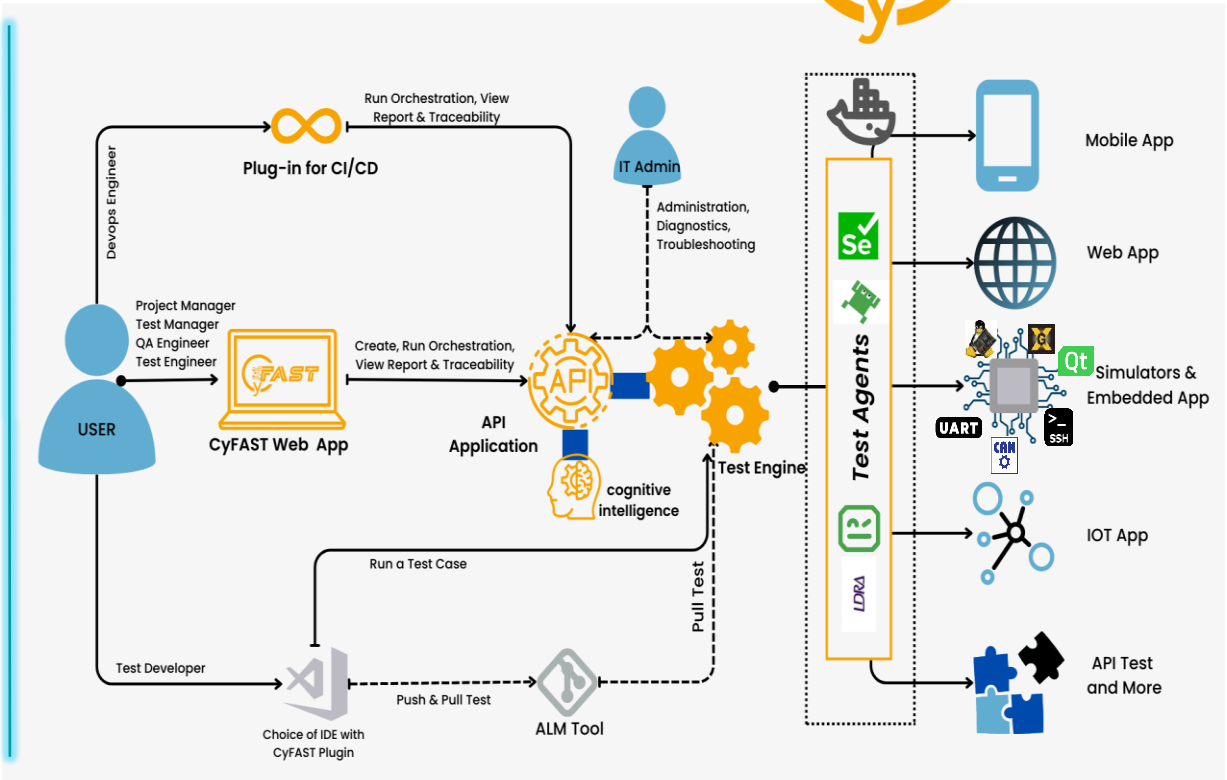
Challenges and Approach





CyFAST as Solution

- **Single Test platform** to serve automation testing on various platforms such as Embedded, Web, Mobile, Desktop, API and Simulators
- Reuses **existing automation assets** to enable end to end solution testing
- **Risk Based** prioritizing test case execution and planning
- **Easy to Integrate with ALM & CI/CD tools**
- Plan and schedule **multisite and parallel/distributed execution** of test cases.
- **Report** is generated in multiple format
- Easy to use, scale or develop **embedded device testing libraries** and hooks



With CyFAST You can,

- **Manage test** automation of multiple projects **at single place** for a customer
- Easy to **Setup and Manage** multisite remote / Native **test environments** at one place
- **Prepare, configure Test Execution Plans** (Orchestrations) for various test environments
- **Orchestrate** testing automatically **on demand** sequentially on configured test environments or targets
- **Maintain and download** execution **logs**, console logs, and test summary **reports** for all orchestration/test cases that are executed
- Establish **traceability** between **test cases, requirements, execution results**, risks, features, and revisions of the software
- **Support multiple embedded devices** with a reusable keyword library and communication strategy
- Analyze **project health, Test coverage, and trends** using **customizable** graphs and charts
- Perform impact analysis and **Auto-Define regression scope** for change in specification, software, and risk scope with cognitive intelligence developed over traceability data
- **Orchestrate** testing automatically **on schedule**, or both sequentially and **simultaneously** on configured test environments or targets

CyFAST Key Differentiator

Cross-platform testing

- One place to test across multiple platforms like Web, Mobile, API, Embedded etc
- Easy to integrate Off-The-Shelf test automation framework
- Easy setup to automation testing for embedded application

Distributed Testing

- Plan and schedule multisite and parallel execution
- Faster, easier, flexible testing of heterogeneous distributed systems
- Agnostic to geographic location

Traceability Management

- Requirement based Traceability
- Risk Based Traceability
- Managing Risk and Requirement coverage

Customized Report

- Report generation based on Customer QA templates
- Built-in Cyient Templates for US FDA compliances
- Built-in Cyient Templates for EU compliances
- Support to Word, PDF, HTML format

Risk-based Testing

- Testing order created intelligently, giving priority to high-risk and CTQ tests
- Identify and Auto-include of flaky tests
- Prioritizing defect-verification tests for regression

Cognitive Intelligence

- Analyze requirements, tests, results and traceability changes
- Perform impact analysis
- Recommend regression plan
- Create deep insights

Features of the offering

Growth Offering

- End-to-end system & software testing
- Cloud based remote execution of tests with ease
- Planning and scheduling multisite and parallel test execution
- Plug-and-play integration with ALM tools (Azure Extension)
- Report customization
- Cognitive intelligence for optimized test orchestration, risk-based testing, and Regression analysis.
- Ready to Use Embedded assets and library to execute tests on embedded systems

Benefits to Customer

- Supports wide range of Applications including embedded applications
- Reduction in License cost
- Increase in Test Coverage
- Lower operational costs
- Improve testing productivity
- Reduction in time-to-market up to 50%
- Plug & Play Model with minimal customization
- Ability to use existing test assets

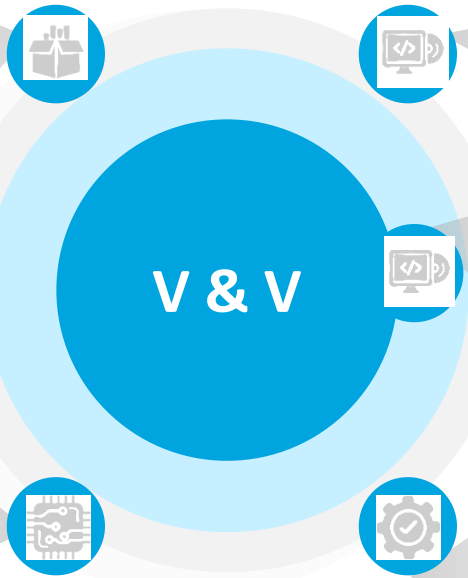
Offerings | Verification & Validation Spectrum

Product

- Functional validation, OTS/COTS Verification, System Reliability Analysis
- Regulatory compliance (CE, CSA, CENELEC, DO, ISTA), Safety and Essential Performance
- Co-Existence Testing, Environmental testing, No. of Cycles Until Failure, Biocompatibility
- Documentation –Compliance report generation

Hardware

- EMI/EMC Testing
- Hardware Fault Insertion Testing, Reliability Testing, Failure Modes and Effects Analysis, Hardware Test Coverage Analysis
- Test infrastructure development
- Interface Testing, Leakage Current and Electric loading, Magnetic Felid



Software

- Static Analysis, Unit testing
- Integration (Manual/Automated)
- Regression (Manual/Automated)
- Functionality (Manual/Automated)
- System Testing (Manual/Automated)
- Tool qualification

Usability / Human Factors

- User Research
- Analysis – safety, hazard, and user Error
- Design and Formative Study in iterations – Low fidelity, High fidelity expert review, User Interface usability test
- Summative study – usability test and risk update

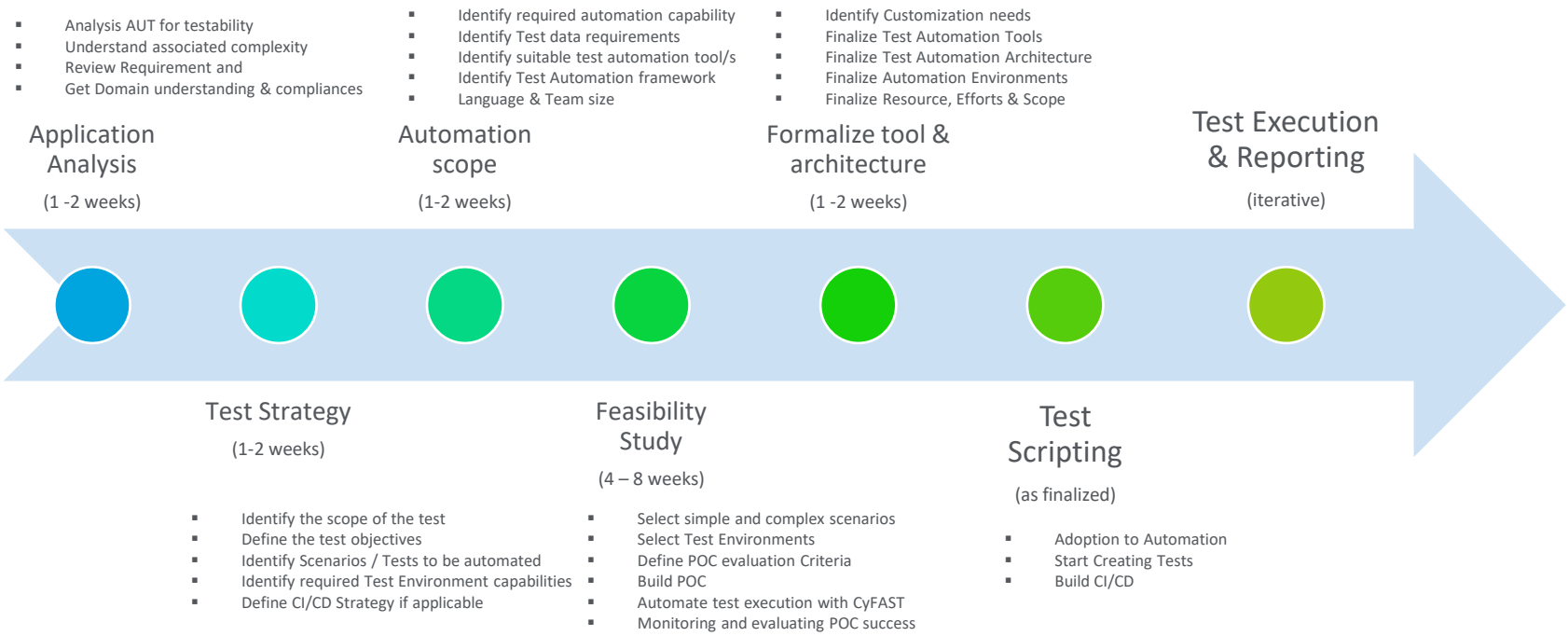
Test Automation

- Consulting services – Testing strategy, Tool assessment & selection
- Transformation services – Test automation implementation
- Documentation services – Compliance report generation
- Accelerator - CyFAST

Engagement Models

- Testing-as-a-Service
- Test CoE Set-up

Customer Onboarding Process



Service/ Engagement/ Cost model

CYIENT

Case Studies

Case Study

CyFAST - Automating Embedded UI Testing for Azure RTOS GUIX Applications

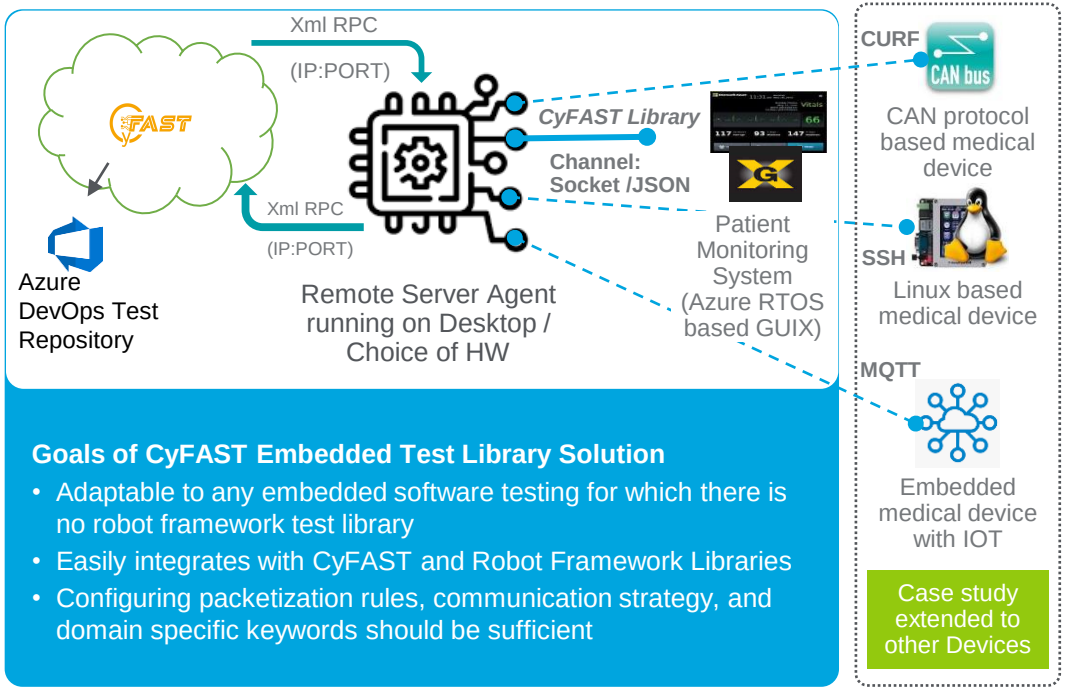
Case Study: The project goal is to reduce costs and increase productivity by adopting test automation for embedded UI applications with CyFAST Embedded Test Library

Challenges

- **Unavailability of embedded UI test libraries** for Azure RTOS GUIX applications
- **Automating remote test execution** for Embedded UI applications

Technical Solution

- Automation of Embedded applications with CyFAST **using CyFAST Embedded Test Library** with a test hook for Azure RTOS GUIX.
- Developing an **agent (interface)** that communicates with the CyFAST cloud service and the Embedded device under test using **customizable communication channel**.
- Developing **reusable generic keyword library** to test Embedded UI Application with a test hook for Azure RTOS GUIX ***(a CyFAST Library)***



Goals of CyFAST Embedded Test Library Solution

- Adaptable to any embedded software testing for which there is no robot framework test library
- Easily integrates with CyFAST and Robot Framework Libraries
- Configuring packetization rules, communication strategy, and domain specific keywords should be sufficient

Case Study

CyFAST - Automating Flight Attendant Panel Application Test using CyFAST

Case Study: It is the objective of this project to migrate legacy test approaches to Behavior Driven Scripting and automate the testing process.

Challenges

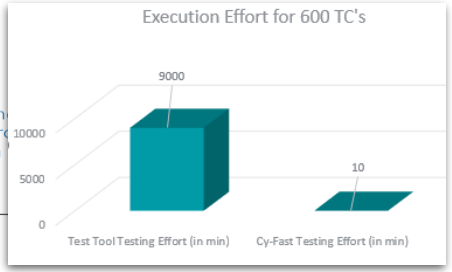
- A 12+ -year-old Java-based Flight Attendant Panel application is being tested semi-automatically
- Writing tests in proprietary assembly language style PDL (Page Description Language) is difficult for novice engineers
- A manual test execution process i.e., Verifying the expected and entering the results.

Value Addition

- Demonstrated remote execution of tests using CyFAST on FAP hardware, enabling simultaneous telnet, FTP, and socket communication
- Streamlined test script in BDD style from the legacy complex approach
- *Reduced execution time of 900 scripts from 900 min to 10 min*
- Increased reliability with Evidences as log and screen captured

Before

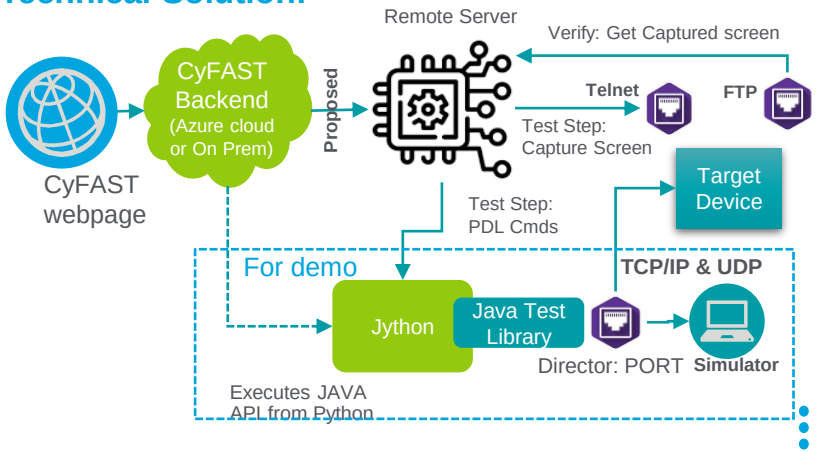
```
P 0x60
EP

UP "Verify that the Prerecorded announcem
displays tabs as PRAM Group 1 and PRAM Gr
UP "Verify that the file selection button area
```



After (BDD)

```
Verifying_tab_titles_on_Prerecorded_Announcement_page
    Given The user opens Prerecorded Announcement page
    And End Page
    And Wait for 60 seconds
    Then Tabs name PRAM Group 1 and PRAM Group 2 are displayed
    Then File Selection button area has folder selection
```

Technical Solution:



Case Study

CYFAST - Automating Hardware testing using CAN Analyzer

Objective of this project, is to migrate to Scripting and automate the testing process for Application Testing.

Challenges

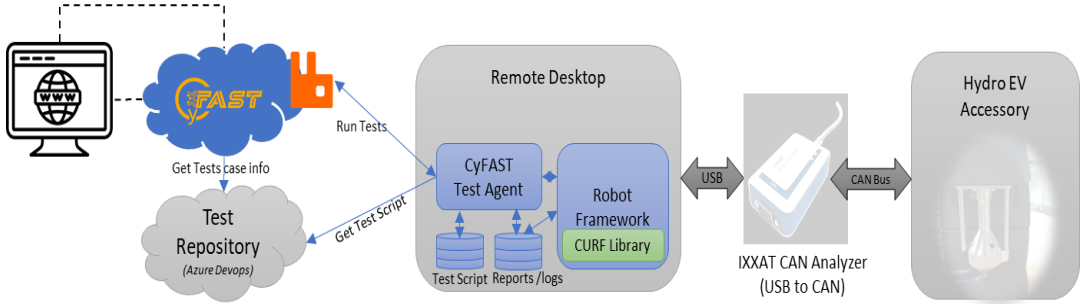
- Manual testing consuming more time, by creating data frames on CAN communication with CAN analyzer tool.
- Each CAN message has its independent behavior.
- Analyzing data on CAN frame using manual testing is most challenging task.
- Maximum 5 CAN frames can be used for testing using CanMini3 Analyzer

```
Can Transmit command for light ON
    ${canlist} =      Send Receive CAN     0X1604A     37 1
    # Log to console     ${data}
    # ${data} =  extract data       ${data}
    ${data}  get from list     ${canlist}      1
    ${canmsg}  get from list     ${canlist}      0
    ${data}=  convert to string     ${data}
    #${op}=     Evaluate      type($ op)
    Log to console     ${canmsg}
     IF    "${data}" == "25"
         log to console  light is ON Successfully
     ELSE
        log to console  light is OFF Successfully
     END
```

Value Addition:

- Single time creation of test script using CyFast(Robot framework) will reduce time for testing on different hardware Module used in CAN based Safety Critical Embedded Application.
- Four different hardware modules test result log can give an easy solution for identifying problem.
- This script can avoid separate automation tools usage.
- This kind of scripts can be used for hardware/Firmware testing which contains application communication.
- 50% to 60% of test cases can be automated using CyFast, resulting in time savings of 50% to 60%.

Technical Solution:



CyFAST - Automating Testing of VueJS Based Web Application on Chrome Browser

Case Study: Project goal is adoption of test automation strategy with Scalable & Maintainable Test Architecture

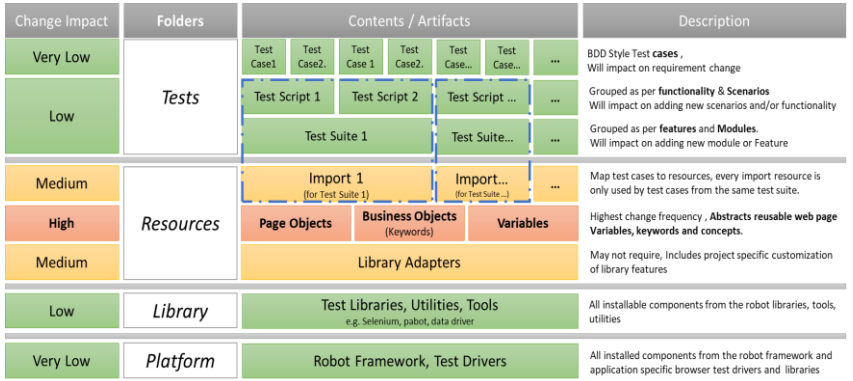
Challenges

- **Automation of UI tests for web applications** based on VueJS
- **High Maintenance costs** due to frequently changing requirements
- **Adaptability** of existing automation solutions and automation scripts **to the latest features and platforms**

Technical Solution

- **Automation** of web applications with **CyFAST using Selenium/Robot Framework**.
- **Gherkin-based test scripts (BDD style)** to make the test scripts technology agnostic
- Creating a **scalable and maintainable test architecture** by following **SOLID guidelines** for scripts, resources, and data.
- **Cognitive Intelligence**-based regression analysis using traceability & test summary reports generated.

Proposed Test Architecture



CyFAST Enables

- Utilization of existing automation assets to enable end-to-end testing
- Prioritizing the execution and planning of test cases
- Integrate ALM tools to update test cases, test scripts, test execution results, and traceability.
- Multisite and parallel execution of test cases can be planned and scheduled.
- The report can be generated in a variety of formats

THANK YOU

#DesigningTomorrowTogether

cyient.com

Follow us on