International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 06 | June 2019 www.irjet.net p-ISSN: 2395-0072

Automation Framework to Validate Interface Gateway Data Dictionary using python Scripting

Jayashree Ukkinagatti J[1], Azra Nasreen [2], Preemy Wilson [3]

[1]PG Student, Dept. of CSE, R V College of Engineering, Bangaluru, Karnataka, India
[2]Assistant Professor, Dept. of CSE, R V College of Engineering, Bangaluru, Karnataka, India
[3]Sr, Technical Product Manager, G E Indian Industrial Ltd., Bangaluru, Karnataka, India

---

Abstract - *The development of automation framework is immensely increasing nowadays to grab the benefits like less time consumption, less manual effort requirement, reusability, consistency, scalability, cost effectiveness, to avoid rework at later stages, to report errors easily, to avoid human errors like missing to validate test data. By means of automation framework for validation process it become easy to ensure the quality and reliability of any system. In this paper an automation framework is discussed that validates the data dictionary of interface gateway.*

***Key Words*: Interface gateway, Control system, Shared memory**

1. INTRODUCTION

To ensure the quality of system or software, validation is important, and testing should be conducted either manually or using automatically. When automated, separate software will be created which will take care of sequence of tests to be conducted and it also handles the process of comparing expected result with the actual result. An automation framework can be treated as an application that provides the guidelines for the automation of any system or product. It consists of set of functions, data to be tested, objects and different reusable modules. When automation frameworks are used for testing purposes then it helps to reduce cost, time and manual effort [1].

The primary objective is to validate the data dictionary of Interface Gateway. The Interface Gateway is a network device which provides real time data from the Control System to Positive Train Control. The Interface Gateway provides fault logs to the operators/customers. The control system used to capture approximately 300+ different parameters value at real time. It is required to check whether Control system is capable of capturing all the parameter values from as and when the values get change and whether Interface Gateway could able to accept the same values from Control system. So, test can be conducted by changing each parameter value sequentially.

To perform the test manually it has several demerits such as, requires much of manual effort and need lot of time and It will be a tedious job to test for each parameter if multiple time checking is required. To overcome from the demerits, an approach is proposed to create an automation frame work which can able to change all the parameter values sequentially and check whether Control system is able to change the parameter values and sent the same to the Interface Gateway.

2. RELATED WORK

A method for the validation of perception sensor models is proposed [2] which helps for advanced driver assistance systems (ADAS). This was suited for quantitative evaluation of sensor models and static environment. In this model validation takes place in stages. Initially it compares the experimental results with environmental model output. Then it compares real data and synthetic data from sensor model as input.

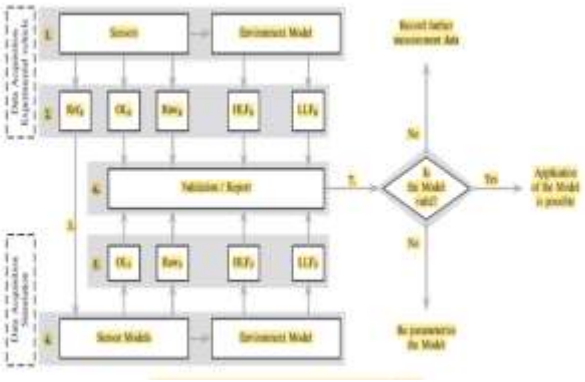

Fig – 1: Validation Model of ADAS [2]

A system is projected [3] that automatically collects GPS traces from mobile devices using GPS sensors. Correct the road map and validate it with the quality requirement provided by users. Used optimized sensing mechanism which helps to reduce energy consumption of mobile devices. Fig.2 illustrates Map Correct model.

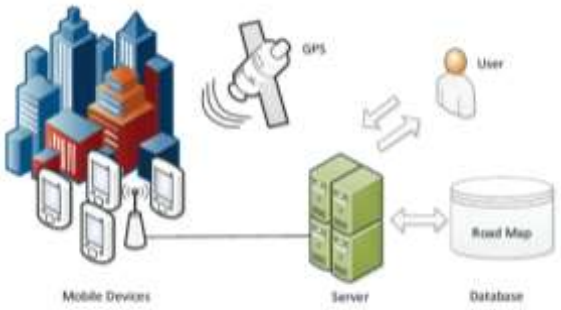

Fig – 2: Map Correct system model

An automation frame work proposed in this paper is developed to change all the parameter values sequentially in Control system and check whether control system can able to direct the changed parameter values to the Interface Gateway thereby minimizing the manual effort, to save time, to make the job easy to test for each parameter multiple times and to avoid human errors in missing out test data.

3. PROPOSED SYSTEM

This segment illustrates the working of proposed automation framework. This framework basically consists of two modules/applications.

- Interface gateway application
- Control system application

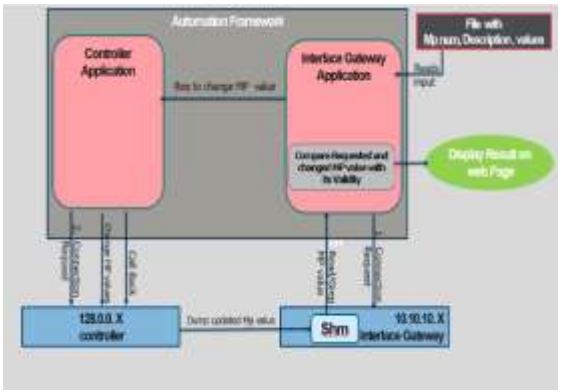

Fig - 3: System Architecture

Fig. 3 depicts the system architecture of proposed automation framework.

To validate that interface gateway data dictionary, initially interface gateway application reads the input file line by line. Each line contains individual monitor parameter details. After reading, interface gateway application sends request message to control system application which consists of monitor parameter's number, control signal name and value to be changed. On receiving the request message, controller change the value of monitor parameters and checks whether it got changed. Dumps the changed value of monitor parameters in the shared memory of interface gateway. Interface gateway application retrieves monitor parameters value from shared and compares it with expected value. Finally launches the result with the summary on web Page.

4. RESULT ANALYSIS

The applications developed in framework are analyzed with respect to response time and accuracy. The developed framework took about 20 minutes to validate 32 monitor parameters values by changing their valid values (92 different values) which usually takes 8 to 10 hours of manual effort. This shows the developed framework saves about 95% of time compared to manual procedure. As the task is automated in the developed framework, it did not miss any test case and the result obtained was accurate.

Table -1: Comparing Time engaged by developed system and manual method

Input	Automation Framework	Manual method
10 Monitor parameters values	10 minutes	2 hours
30 Monitor parameters values	32 minutes	8 hours
50 Monitor parameters values	55 minutes	10 hours
72 Monitor parameters values	80 minutes	15 hours

Table.1 evidently illustration that the automation framework works well to validate monitor parameter by engaging very small time.

The experimental results are also analysed with respect to system accuracy. The developed automation framework works very accurately while delivering the validation results. It does not miss any data to be tested and clearly displays the result in a short time.

Table - 2: Accuracy of proposed system

	Predicted Fail	**Predicted Pass**	**Accuracy**
Actual Fail:	TN=02	FP=02	(90+2)/94 = 0. 97
Actual Pass:	FN=00	TP=90	

Table 2 depicts the accuracy of proposed system. Here totally 94 values were changed for different monitor parameters and expected 92 will produce positive result and 2 will produce negative result. The System end up creating 90 positive results and 4 negative result. So, this indicates, system is approximately 97% accurate.

The following are Screen Shots of proposed Automation Framework.

RESULT

Fig – 4: Monitor parameter validation result

When system validates MPs value interface gateway application updates each MP values validated result on page. Figure 4 depicts the monitor parameters validation Result. The reason for failure is given on the comment i.e. with respect to tag number 0x300B value got changed successfully but validity is not 7F so resulted as Fail.

5. CONCLUSION

The Automation Framework intended to validate the data dictionary of Interface Gateway and verify whether all the data changes encountered at control system are efficiently transferring to the interface gateway's shared memory. Using 'paramiko' and 'telnetlib' python libraries interface gateway and control system devices were connected to local system. To change monitor parameter values control signals were executed in control system. QNX commands were used to retrieve data from interface gateway's shared memory. HTML tags were used to create result webpages. This developed automation framework benefitted in several way like.

It reduced the manual effort and saved time of validation process. It became ease to conduct validation process multiple times as and when required. Early detection of defects avoid rework in later stages of project. It avoids human errors in missing out test data. Compared to Manual validation process the accuracy has been increased by 40% and the time taken to perform validation has been reduced up to 95% by considering all test data.

REFERENCES

[1]Kumar Mandula,Ramu Parupalli ,CH.A.S. Murty , E. Magesh ,Rutul Lunagariya,"Mobile based home automation using Internet of Things(IoT)", 2015 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), pp.381-3, 2016.

[2] AlexanderSchaermann, AndreasRauch, NilsHirsenkorn, Timo Hanke, Timo Hanke, Ralph Rasshofer, Erwin Biebl, "Validation of Vehicle Environment Sensor Models ", IEEE Intelligent Vehicles Symposium P.NUM: 405-411, 2017.

[3] Patrick Baier, Harald Weinschrott,Frank D¨urr, Kurt Rothermel, "MapCorrect:Automatic Correction and Validation of Road Maps Using Public Sensing" , 36th Annual IEEE Conference on Local Computer Networks pp: 58 – 66,2015.

[4] Priyanka Paygude, P. R. Devale, "Automation Of Data Validation Testing For QA In The Project Of DB Migration", International Journal of CSE and IT Research (IJCSEITR), pp:15-22, 2015.

[5] Stanislav Stresnjak, Zeljko Hocenski," Usage of Robot Framework in Automation of Functional Test Regression", The Sixth International Conference on Software Engineering Advances, pp.30-34, 2011.

[6] G. Lindstrom, "Programming with Python", **Published in:** IT Professional ,Volume: 7, pp.10-16,2005

[7] R. Nageswara Rao, " Core Python Programming", Dreamtech Press, 2nd Edition, 2018.

[8] Berners-Lee, Tim, Connolly, Daniel , "Hyper Text Markup Language (HTML) Internet Draft version 1.1". *IETF IIIR Working Group,1993*