International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 12 Issue 2 ǁ February 2024 ǁ PP. 98-103

Generating Test Cases from Web based Information

Ashok Kumar Panda
College of Engineering Bhubaneswar, Biju Pattnaik University of Technology, Odisha, India.

Abstract: *As per the requirements specification, SaaS (Software as a Service) has to be running twenty four hours a day. Keeping this in mind, updating the related test cases is difficult. Managing requirements in this context has additional challenges. When requirements change, related test cases need to be updated accordingly. We claim that extracting and analyzing the usage of the SaaS can help to maintain requirements and test cases updated and contribute to improve the overall quality of the services provided. This paper presents an extension to REQAnalytics. The evolution involves extending the analysis performed over the sequences of functionalities and refining the data provided for Software Requirements Specification (SRS). This paper also tries to help the requirements engineers in the requirement maintenance activities, and to improve the overall quality of the services.*

Keywords: *Requirements engineering, test cases, web usage, software requirement specification*

Date of Submission: 04-02-2024 Date of acceptance: 16-02-2024

I. Introduction

The recent studies about internet users [1][2], there are four and a half billion Internet users in the world,representing62%of the world population, that have access to more than 192 million active websites. By using specialized tools, website owners can capture data related to all the users' interactions with their website.

In the context of electronic services, by analyzing this data, relevant knowledge is gathered that can help the service providers in improving the user experience with the service, identifying new development opportunities, observing user navigation patterns, predicting user behavior, or getting better understanding of the service usage.

However, in the context of requirements engineering, this data can provide useful knowledge necessary for assisting the requirements management activities and improve the overall quality of the service provided [6].

Nowadays, in Software Engineering, one of the biggest concerns has to do with software quality, i.e., ensure that software is built systematically, rigorously, measurably, on time, on budget, and within specification.

As software evolves along time, there is the need to define new requirements, update existing ones and implement changes accordingly. Change requests may originate from different stakeholders, such as developers, systems engineers, users and service integrators.

Changes in requirements, their uncontrolled evolution throughout the project and a weak requirements elicitation are the main causes for software project failures [2]. Also, whenever requirements change, related test cases must be updated accordingly.

Managing requirements helps to prevent the negative impacts of the uncontrolled requirements changes and assure that the stakeholder's needs are fulfilled. Also, ensuring the traceability between requirements and the system's artifacts contributes to the development of reliable software [18], especially in the context of a distributed development environment with multiple collaborating stakeholders [19].

By analyzing the effect of change and auditing, appropriate requirements management can be performed [3].

However, keeping the Software Requirements Specification documentation and related test cases updated is difficult, particularly on cases where the software project is a website/service that evolves during its lifetime.

This happens, despite of software organizations are trying to improve their processes used to collect, analyze, document and maintain their requirements in a structured requirements specification written in natural language [4]. As times goes by, a deficient requirements and test case management process leads to obsolete requirements and tests that do not express the actual state of development of websites, therefore becoming useless.

REQAnalytics extracts data about the usage of the software services and analyzes that data in order to provide recommendations. Among others, these recommendations help to maintain requirements updated and, ultimately, contribute to improve the overall quality of the software services provided.

Data gathered by web analytic tools together with a proper analysis of that data may produce recommendations to improve the overall software service.

This paper presents an extension to REQAnalytics. It uses the data gathered about the usage of the

software service and generates functional tests that can be used in the regression testing context and contribute to improve, even further, the software service provided along the time.

II. The REQ Analytics System

REQAnalytics is a novel approach/tool that aims to improve the requirements maintenance based on the usage of a website gathered by a web analytics tool and generating recommendations reports that may help the requirements maintenance and increase the overall quality of the software.

REQAnalytics, should not be labeled as a Requirements Management tool, despite having some features commonly available on such tools, since its goal is to complement and support the task of requirements maintenance providing several recommendations. A fully functional demo of REQAnalytics system is available at http://web.fe.up.pt/~reqanalytics.

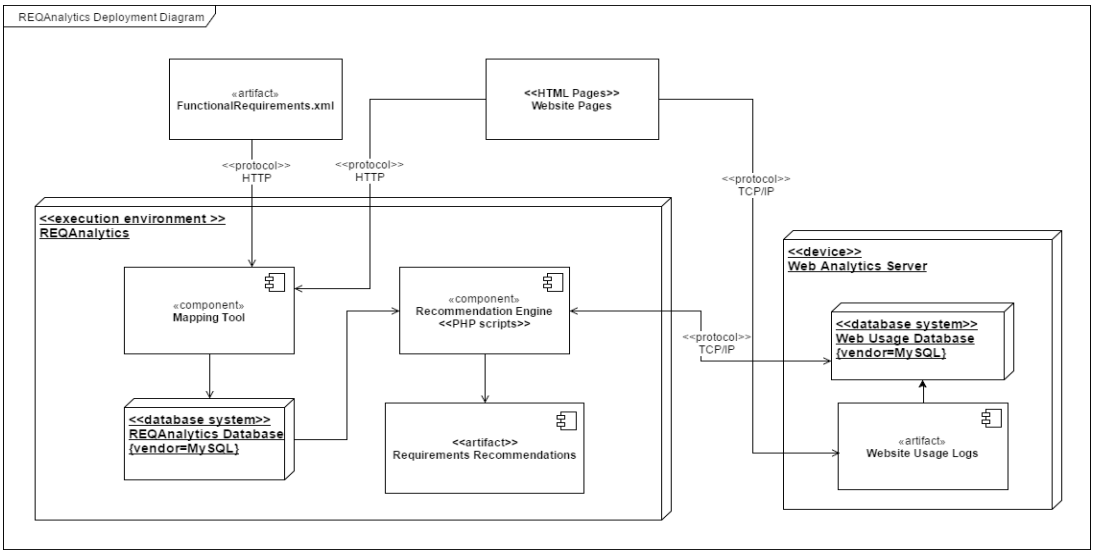

Fig.1. Deployment Diagram of REQAnalytics.

REQAnalytics Architecture

REQAnalytics was designed to be used through a web browser. It uses a web analytics tool, OWA -Open Web Analytics - in order to extract the web usage information from the web service under analysis. REQAnalytics is developed in PHP and uses MySQL databases as support. The Deployment Diagram of REQAnalytics is shown in figure 1.

The main components of REQ Analytics include:

- **Mapping Tool**. This component allows to relate functional requirements (provided as an XML file *FunctionalRequirements.xml* in Figure 1) with the web pages and UI (User Interface) elements of the website as described in[5]and based on a point and click process similar to the one presented in [8].The web elements related to a functionality are the ones that implement that functionality from the point of view of the user. The mapping information between functional requirements and UI elements is saved in *REQAnalytics Database* inFigure1.This web tool can work with any web application or websites in ceit was developed inside a bookmark let that works in any web browser.
- **Web analytics tool**. At this moment REQ Analytics uses OWA to collect data from the usage of the software service; however, another tool may be used. The data collected by OWA is saved in the *Web Usage Database* in Figure 1.
- **Recommendation Engine**. This component reads the mapping information saved in REQ Analytics Database and the usage information saved in *Web Usage Database* and performs different analysis in order to produce different recommendations.

REQAnalytics process

Aspresentedin[6][7],theREQAnalyticssystemisdividedinfourdifferentphases:

- Requirements mapping
- Collect Web usage data
- Analysis of the data collected
- Generation of recommendation reports

Mapping of requirements: The functional requirements of the website under analysis are mapped with the implementation artifacts like the URL of the web pages and HTML elements of the website. The mapping is established by a point and click process similar to the one presented in [8].
The information captured by this process is saved in a database. This mapping is a very important task in the process, since it will allow to establish correlations between the functional requirements and the implementation in order to allow REQAnalytics analyze and then generate recommendations in an abstract level closer to the analysts (at the functional level).

Collect Web usage data: The purpose of this phase is to collect the web usage data using the OWA [9]. It gathers different data like pages visited, DOM elements clicked, click paths, entry pages, exit pages and duration of session. This data is stored in a database for further analysis.

Analysis of the data collected: The REQAnalytics lists and analyzes the web usage data collected by the web analytics tool, with the mapping information stored in the first phase. The goal is to identify possible improvements for the website. So, it analyzes the paths taken by users while visiting the website to be able to identify, among other improvements, work flow changes [7],which are the most and least used functionalities, change the priority of the requirements, detect navigation paths' patterns [10]and detect new dependencies among requirements [11], among others [6].

Generation of recommendation report: After the analysis made in the previous phase, REQAnalytics generates several reports with recommendations and the results of the analysis performed, such as, Traceability Matrix Report [5]; Requirements Analytics (Statistics And Main Metrics) [6]; Recommendations to the Requirements Specification, among others [6] [7].

III. REQAnalytics Extension

Web analytics tools provide information about the sequence of actions performed by the users when interacting with the system.
These reports are difficult to read because they are based on the codes of the UI (User Interface) elements. REQAnalytics produces a report in a language closer to the analysts. It analyzes the data provided by web analytic tools and transforms the sequences of interaction on sequences of functionalities performed during the user sessions. Based on the mapping information gathered during the tool checks the subsequences of interactionthatcorrespondtofunctionalitiesandgeneratesnewsequencesbasedonfunctionalitiesperformedinstead of UI elements interacted with by the user. The algorithm used to perform this transformation is presented in[11]. A simplified version of this process is illustrated in Figure2.

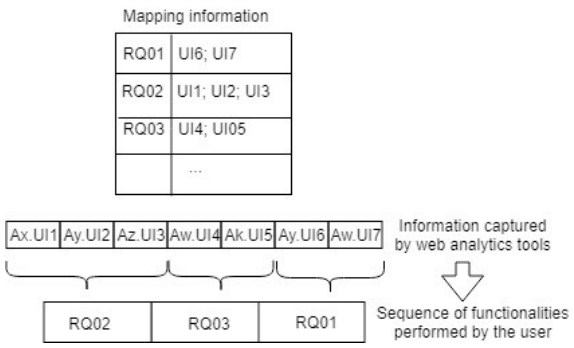

Fig.2. User actions transformed into functionalities.

The information about the way users use a software system may be useful to generate test cases for systems that evolve over time and for which there is the need for regression testing.
The REQAnalytics extension that we present in this paper is able to generate functional tests that can be continuously updated along time and adapted to the new functionalities of the Software Service provided. This extension can help in prioritizing and selecting the test cases, which is an important aspect of the overall software testing process [20] [21].
The test cases are generated according to three different criteria:

- Most frequent sequences of functionalities;
- Most frequent sequences of functionalities of a specific size;
- Most frequent sequences of functionalities to achieve a specific goal.

Most frequent sequences of functionalities

The tool lists the most traversed sequences of functionalities performed by the users. It starts by sorting the sessions by size and then it calculates iteratively the number of times each sequence is performed. At the end, it presents the listsorteddescendingbyfrequency.AsmaybeseeninFigure3, the tool shows the sequences performed, the number of times they were performed, and the percentage of times they were performed when considering all sequences of interactions. Also, when you move the mouse on top of a specific requirement, it shows a label with its description.

Fig.3. Most frequent sequences of functionalities.

Most frequent sequences of functionalities of a specific size

When selecting this test case generation criterion, the user needs to input the size of the sequences he wants to analyze. After that, the tool presents, in a descending order by frequency, the sequences of functionalities (of the size defined) performed by the users. It shows the sequences performed and the number of times each one was performed (see Figure 4).

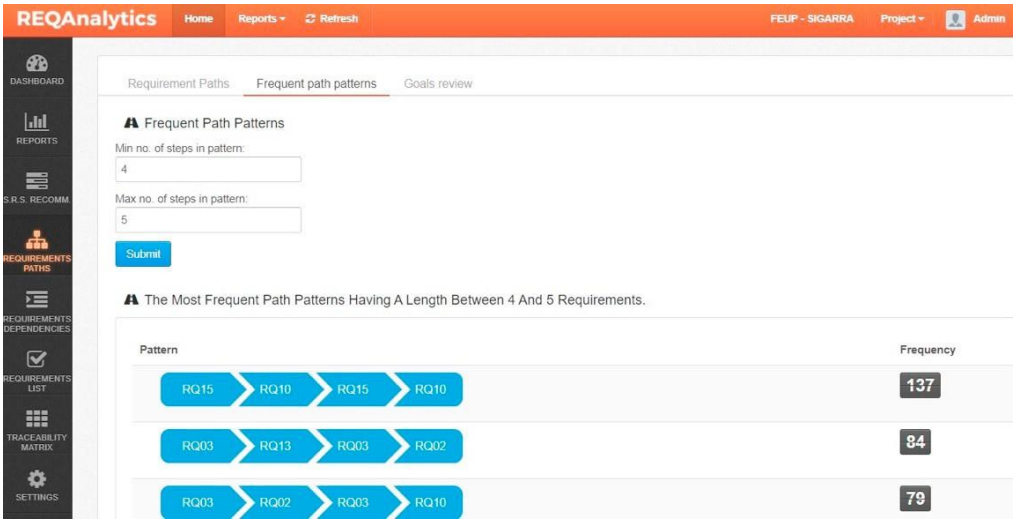

Fig.4.Mostfrequentsequencesoffunctionalitiesofaspecificsize.

Most frequent sequences of functionalities to achieve a specific goal

When choosing this criterion, the tool generates sequences of interactions performed by the users to achieve a specific goal: the smallest; the largest; and the most frequent sequences of functionalities performed to achieve the goal defined (see Figure 5). As can be seen, all sequences contain the goal specified (in this case, RQ03).

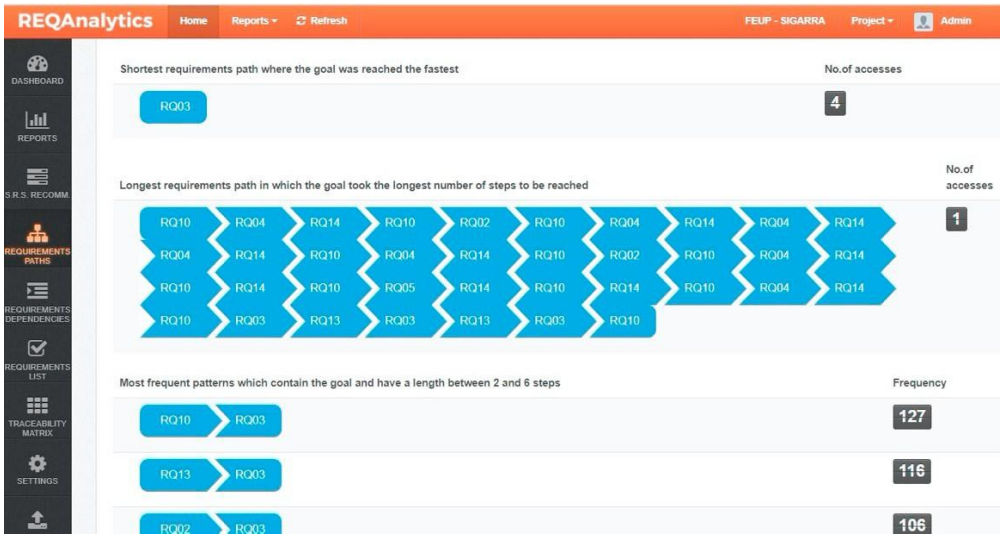

Fig.5.Most frequent sequences of functionalities performed to achieve a goal.

IV. Related Work

There are different approaches and techniques to build test cases, either manually or automatically.
The problem with manual generation is the difficulty to build a set suite that is able to achieve a high level of coverage. Regarding automatic test case generation, the most common are the generation of random tests and Model-Based Testing (MBT) [12].
Random testing may be useful to detect crashes, but it may be difficult to achieve high level of coverage. If we have 100 events, each event has a 1/100 probability of occurring.

The main problem with Model-Based Testing is the need for a model which, most of the times, becomes useless because it is not updated continuously to describe the changes the software services require along the time.

To overcome these problems there are approaches which combine reverse engineering with testing [13] [14].The goal may be to extract automatically part of the model that needs to be analyzed/completed/validated before being used as a base for test case generation. Even diminishing the modeling effort, this approach may require too much effort in a context where the software system under analysis is constantly changing.

Another, not so common, way to generate test cases that may fit better for software services in continuous adaptation is to generate test cases from web usage data.

In [15] the authors present an approach for Android that follows a Record - Mine -Generate - Validate approach to generate scenarios or test cases. They record the actions performed by the users when interacting with the apps; then they apply data mining techniques to Web click stream data in order to extract usage patterns and, afterwards, theyanalyzetheresultsproduced.Regardingtheapplicationofdataminingtechniquestoanalyze Web data, there are also different approaches [16].

However, none of these approaches link web usage data with functionalities as REQAnalytics does and so, are not able to generate test cases at a functional level, i.e., sequences of functionalities to perform.

In a context of a software service that must be updated according to change requests and being working 24 hours per day and 7 days per week, maintaining specifications and test cases updated is of utmost importance to guarantee high levels of quality.

REQAnalytics generates recommendations to improve the software services and test cases (with the extension presented in this paper) that may be generated automatically and adapted when updates are implemented.

V. Conclusions

This paper presents an extension to REQAnalytics, a recommender system that maps the functional requirements of a software service with the features elements of the GUI that implements them in the point of view of the user. By relating this information with the web usage data, REQAnalytics is able to generate recommendations to improve the software service provided. These test cases may be generated according to different criteria, such as, most frequent, most frequent of a specific size and most frequent to achieve a goal. At the end of the test case generation process, the tester can get the sequences of interactions according to these criteria and save them as high level test cases for reproducing whenever some change request is implemented in the system.

In the future, the test cases generated by this approach may be executed automatically. For that, it is necessary to perform the inverse process, i.e., transform the functionalities into sequence of user actions. That is possible because we save them aping in formation in a database. By transforming the high-level functionalities into atomic user actions it is possible to build test scripts (for instance, using Selenium) that can be executed automatically.

References

[1] Internet World Stats, 2020, World Internet Users and 2020 Population Stats. Available:http://www.internetworldstats.com/stats.htm.[15May2020].

[2] StandishGroup,"CHAOSReport,"Group,pp.11- 13,1995.[Online].Available:http://www.standishgroup.com.[15May2020].

[3] N.Ibrahim, W.M.N. WanKadir, and S.Deris," Propagating requirement change into software high level designs towards resilient software evolution," in Proceedings - Asia-Pacific Software Engineering Conference, APSEC, 2009, pp. 347–354.

[4] K.E.Wiegers,"AutomatingRequirementsManagement,"1999.

[5] J.E.Garciaand A.C.R.Paiva," A Requirements-to-Implementation Mapping Tool for Requirements Traceability, "Journal of Software, vol. 11, no. 2, pp. 193–200, 2016.

[6] J.E.GarciaandA. C.R.Paiva," An Automated Approach for Requirements Specification Maintenance," inWorldCist'16 -4thWorld Conference on Information Systems and Technologies, mar 2016.

[7] J.E. Garcia and A. C. R. Paiva," REQAnalytics :A Recommender System for Requirements Maintenance," International Journal of Software Engineering and Its Applications, vol. 10, no. 1, pp. 129–140, 2016.

[8] A.C.R.Paiva,J.C.P.Faria,N.Tillmann,andR.A.M.Vidal,"Amodel-to-implementation mapping tool for automated model-based GUI testing, "in ICFEM'05 Proceedings of the 7th International Conference on Formal Methods and Software Engineering, ser.Lecture Notes in Computer Science, K.-K. Lau and R. Banach, Eds., vol. 3785.Berlin, Heidelberg: Springer Berlin Heidelberg, nov. 2005, pp. 450–464.[Online]. Available: http://dl.acm.org/citation.cfm?id=2099878.2099918

[9] "Open Web Analytics. Available from http://www.openwebanalytics.com/,"2015.[Online]. Available:http://www.openwebanalytics.com/

[10] A.Garcia and A.C.R.Paiva," SaaS Usage Information for Requirements Maintenance, "in 16th International Conference on Enterprise Information Systems (ICEIS), 2014.

[11] J.E.Garcia and A.C.R.Paiva," Manage software requirements specification using web analytics data,"in Trends and Advances in Information Systems and Technologies-Volume2[WorldCIST'18, Naples,Italy,March27-29,2018],2018,pp.257–266.[Online]. Available: https://doi.org/10.1007/978-3-319-77712-225

[12] M.Utting and B.Legeard, Practical Model-Based Testing:A Tools Approach. Morgan Kaufmann,2006.

[13] I. C. Morgado and A. C. R. Paiva, "Mobile GUI testing, "Software QualityJournal,vol.26,no.4,pp.1553–1570,2018.[Online].Available:https://doi.org/10.1007/s11219-017-9387-1

[14] I.C.Morgado, A.C.R. Paiva, J.P.Faria, and R.Camacho,"GUI reverse engineering with machine learning,"in Proceedings of the First International Workshop on Realizing AI Synergies in Software Engineering, RAISE 2012, Zurich, Switzerland, June 5, 2012,2012,pp.27–31.[Online].Available:https://doi.org/10.1109/RAISE.2012.6227966

[15] M.Linares-Vásquez, M.White,C.Bernal-Cárdenas,K.Moran,andD.Poshyvanyk,"Mining and roid app usages for generating action able gui-based execution scenarios, "IEEE International Working Conference on Mining Software Repositories, vol. 2015-August, pp. 111–122,2015.

[16] R.s. Rao and J. Arora, "A survey on methods used in web usage mining" in International Research Journal of Engineering and Technology(IRJET), vol. 4, no. 5, pp.2627-2631, 2017. [Online]. Available: https://www.irjet.net/archives/V4/i5/IRJET-V4I5669.pdf

[17] Netcraft, 2018, July2020 Web Server Survey. Available from https://news.netcraft.com/archives/category/web-server-survey/ html [27 July2020].

[18] K. Yoshino and S. Matsuura, "Requirements Traceability Management Support Tool for UML Models" in Proceedings of the 2020 9th International Conference on Software and Computer Applications (ICSCA 2020), pp.163-166, 2020. [Online]. Available: https://doi.org/10.1145/3384544.3384586

[19] R.Wohlrab, E.Knauss,J. Steghoferetal.,"Collaborative traceability management: a multiple case study from the perspectives of organization, process, and culture "in Requirements Engineering, vol. 25,pp.21-45,2020. [Online]. Available: https://doi.org/10.1007/s00766-018-0306-1

[20] M.Abbas, I.Inayat, M.Saadatmandand N. Jan, "Requirements Dependencies-BasedTestCasePrioritizationforExtra-FunctionalProperties"in 2019 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW), pp. 159-163, 2019. [Online]. Available:https://doi.org/10.1109/ICSTW.2019.00045

[21] R.Butool, A. Nadeem,M. SindhuandO.u. Zaman, "Improving Requirements Coverage in Test Case Prioritization for Regression Testing,"2019 22nd International Multi-topic Conference (INMIC), Islamabad, Pakistan, 2019, pp. 1-6, doi: 10.1109/INMIC48123.2019.9022761.