Intelligent rating tool for software quality attributes

Mara Diana Hajdu-Măcelaru

Abstract. Using a rating tool for software quality attributes is an important part of the development process of each software product, as it helps with the evaluation of the quality as by providing a good image of the quality attributes that should be taken in consideration. We will describe the challenges of developing a quality rating tool that is also adaptable to the project type and user importance.

1. Introduction

To evaluate the quality of a software product, it is very important to decide which of the quality attributes to be taken in consideration. Depending on the platform, on the project type, on the product intent the attributes that should count into the evaluation differ, as well as their rating importance. Software quality attributes represents an important step in the evaluation of the quality. Software quality attributes importance is different based on the application type that is build. In a mobile environment some attributes will be of a bigger importance than the ones from web, for example. Based on the application type we will define the software quality attributes importance and create a model for each application type. We will describe the challenges of developing on different platforms and which are the quality attributes that need to be taken into account when developing mobile or web or desktop application. Quality has different meanings for customers, users, management, marketing, developers, testers, quality engineers, maintainers, and support personnel. Many institutes and organizations have their own definitions of quality and their own quality characteristics and models. It is very hard to find a quality model that can be adaptable based on the user type, application type or environment type and to work well for all organizations. A first adaptation of the quality model can be made based on the application type. Depending on the application type, we have different quality factors based on each platform on which the application is developed. We cannot apply same quality model on each type of application, meaning that, in case we construct a web application we will have some quality factors that matters more for web application than for a mobile or desktop. Similar, if we construct a mobile application we will have some of factors that matters more in the mobile environment than in the web or desktop environment. In this way, we can state that based on the environment we have different sets of quality factors that needs to be taken into account more or less.

2. Specification

The tool intend is to be used inside a company to get from the people involved in the product evaluation, to rate the quality attributes that should be taken in consideration when evaluating the overall product. All the tools that exists nowadays are simple questions where the people involved just respond and in the end the attributes with higher

score are considered to be the most important ones. Of course this approach is not relevant at all, as we may have few people that are involved in the project, or some of them are developers, some are testers and we may have projects in which testers opinions matters more, others in which the users opinions is more important and so on. Also, we may have similar projects that were successfully embraced by users, and would be good to take in consideration the ratings from previous successful projects. For this purpose we have built a tool, that uses the rates from similar projects, and using the fuzzy logic we have created a tool that is relevant also based on user type. The application was built in PHP+MYSQL and we call it SQATool. The application tool has 4 user types: ADMIN, DEV, TEST and USR. The ADMIN is the user that manages everything, and the DEV represents the Developer type, TEST the tester type and USR a simple user. As we can see, all user types that should be involved in the development of the product are define. Logged as an administrator, we can add/edit/delete platform [by default we have mobile, web and desktop], can edit/add/delete users and attributes , can add/edit/modify projects ,can edit/add/delete the correlation between attributes, cand also add/edit/delete which users should rate the attributes for certain project from the project list- see Figure 1.
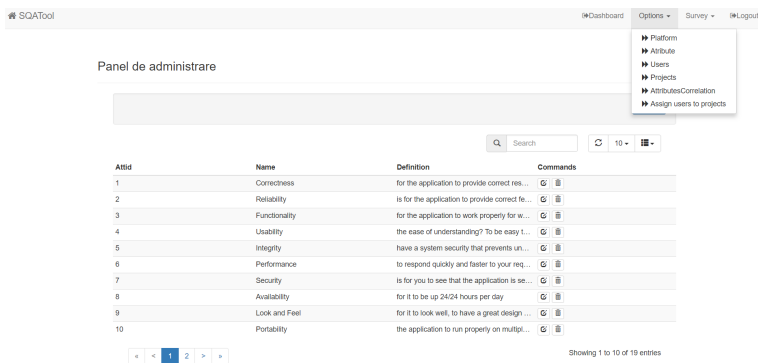

FIGURE 1. Admin SQATool dashboard

For each project, the admin also adds the project type [web, mobile, desktop], so that if we have a new project that is on mobile platform and want to use rates from all the mobile projects we had, in the final report we can select to have included in our final report all the mobile projects that we had just with a simple select. In the attributes list we can add the attributes that we want to have included in our model and a short definition of the attribute – see Figure 2.

In this way, the question to rate the attributes will be taken from the attribute list, by also providing a short description of the attribute that needs to be rated. For each attribute the user that rates them, need to choose from four options: "Very Important", "Important", "Not so important", "Doesn't matter at all" - see Figure 3.

In the admin panel, the admin user can see all responses that were given, can select to see responses from a user for a specific project, or to see all responses from a user, or all responses from a project, depending on what is more relevant for him. Also, a statistic is provided to see how many responses were given by users, and how many are in pending-meaning that it waits for users to give rates for specific projects. This way the admin can have a better idea related to the statistic of the rates provided. On the user side (logged as a dev/test/user) , after login, the user has display the remaining rated survey for specific
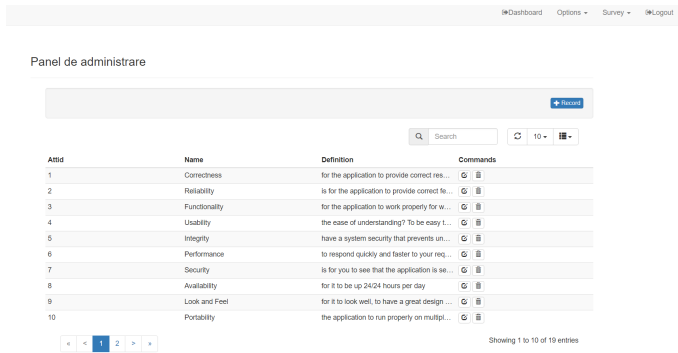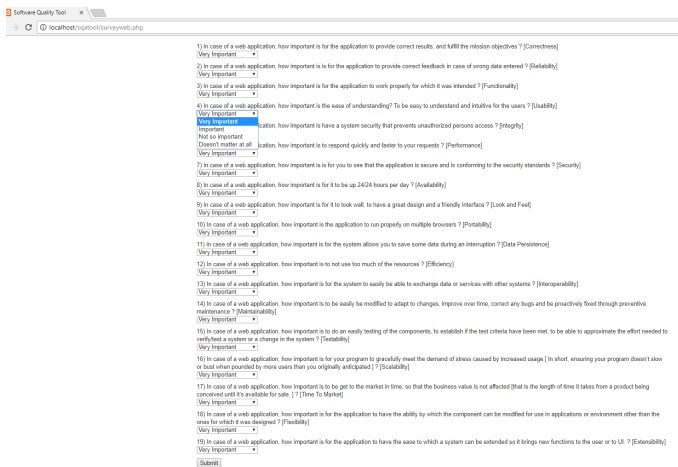
FIGURE 2. Admin SQATool Attribute list

FIGURE 3. Survey to rate the attributes

project, for which he is involved – see Figure 4, and he can also see his responses from previous rated projects.
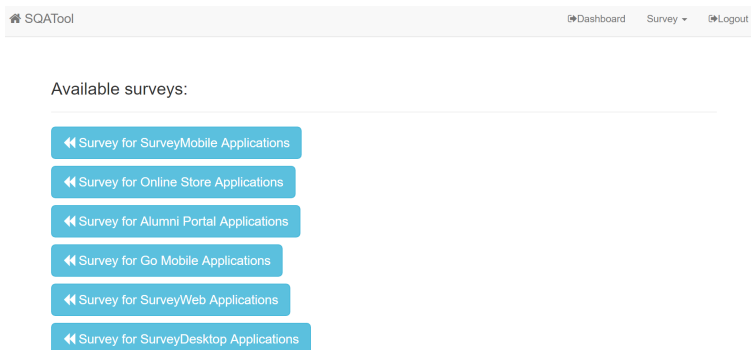

FIGURE 4. User SQATool dashboard

By just a simple click on the dashboard for the remaining projects to be rated, he can start the survey and chose what he thinks is suitable for that specific project. Bu default the questions and attributes to be rated are:

″ 1) In case of a web application, how important is for the application to provide correct results, and fulfill the mission objectives? [Correctness]

2) In case of a web application, how important is for the application to provide correct feedback in case of wrong data entered? [Reliability]

3) In case of a web application, how important is for the application to work properly for which it was intended? [Functionality]

4) In case of a web application, how important is the ease of understanding? To be easy to understand and intuitive for the users? [Usability]

5) In case of a web application, how important is have a system security that prevents unauthorized persons access? [Integrity]

6) In case of a web application, how important is to respond quickly and faster to your requests? [Performance]

7) In case of a web application, how important is for you to see that the application is secure and is conforming to the security standards? [Security]

8) In case of a web application, how important is for it to be up 24/24 hours per day? [Availability]

9) In case of a web application, how important is for it to look well, to have a great design and a friendly interface? [Look and Feel]

10) In case of a web application, how important is the application to run properly on multiple browsers? [Portability]

11) In case of a web application, how important is for the system allows you to save some data during an interruption? [Data Persistence]

12) In case of a web application, how important is to not use too much of the resources? [Efficiency]

13) In case of a web application, how important is for the system to easily be able to exchange data or services with other systems? [Interoperability]

14) In case of a web application, how important is to be easily be modified to adapt to changes, improve over time, correct any bugs and be proactively fixed through preventive maintenance? [Maintainability]

15) In case of a web application, how important is to do an easily testing of the components, to establish if the test criteria have been met, to be able to approximate the effort needed to verify/test a system or a change in the system? [Testability]

16) In case of a web application, how important is for your program to gracefully meet the demand of stress caused by increased usage.[In short, ensuring your program doesn't slow or bust when pounded by more users than you originally anticipated.] ? [Scalability]

17) In case of a web application, how important is to be get to the market in time, so that the business value is not affected [that is the length of time it takes from a product being conceived until it's available for sale.] ? [Time To Market]

18) In case of a web application, how important is for the application to have the ability by which the component can be modified for use in applications or environment other than the ones for which it was designed ? [Flexibility]

19) In case of a web application, how important is for the application to have the ease to which a system can be extended so it brings new functions to the user or to UI. ? [Extensibility]″

The questions are automatically composed from the attribute list using the attribute name and definition. To have a more accurate results, we use the fuzzy logic, and triangular fuzzy sets to mark the importance criteria. The concept of Fuzzy Logic was first conceived by Lofti Zadeh in 1965, who presented it as a way of processing data by allowing a partial membership set rather than a crisp membership set or non-membership. Fuzzy Logic incorporates a simple, rule-based "If X and Y then Z" approach for solving the problem rather than solving it mathematically. The Fuzzy Logic model is completely

empirical and relies on the experience of the operator rather than the technical under-standing of the subject. The technique of triangular fuzzy has been adopted in this paper. Fuzzy Logic is a powerful problem-solving methodology that can be used for applications in many areas such as embedded control and information processing. Fuzzy Logic provides an easier way to infer definite conclusions from highly imprecise, vague, and ambiguous information when compared with classical logic. Fuzzy Logic brings us close to human decision making, enabling one to analyze approximate data to precise solutions.

Fuzzy Multiplication - Let (a, b, c) and (x, y, z) be two triangular fuzzy sets, then the fuzzy multiplication for triangular fuzzy sets is defined as

$$(a, b, c) \times (x, y, z) = (a \times x, b \times y, c \times z)$$

Fuzzy Addition - Let (a, b, c) and (x, y, z) be two triangular fuzzy sets, then the fuzzy addition is defined as

$$(a, b, c) + (x, y, z) = [\max(a, x), \max(b, y), \max(c, z)]$$

For example, we can use the following values as fuzzy degree:

User importance criteria	Fuzzy degree:
Very Low	(0.0,0.0,0.2)
Low	(0.0,0.2,0.4)
Medium	(0.2,0.4,0.8)
High	(0.4,0.8,1.0)
Very High	(0.8,1.0,1.0)

Dev importance criteria	Fuzzy degree:
Very Low	(0.0,0.0,0.3)
Low	(0.0,0.3,0.6)
Medium	(0.3,0.6,0.9)
High	(0.6,0.9,1.0)
Very High	(0.9,1.0,1.0)

Test importance criteria	Fuzzy degree:
Very Low	(0.0,0.0,0.25)
Low	(0.0,0.25,0.5)
Medium	(0.25,0.5,0.75)
High	(0.5,0.75,1.0)
Very High	(0.75,1.0,1.0)

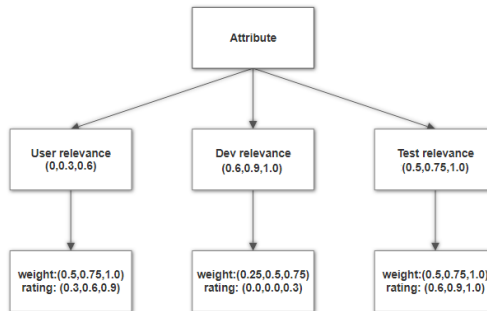FIGURE 5. Fuzzy criteria for user/test/dev evaluation importance

FIGURE 6. Fuzzyfication for an attribute

In doing the final evaluation we will multiply the corresponding triangle sets in order to evaluate and we will make the defuzzyfication in order to have real time data.
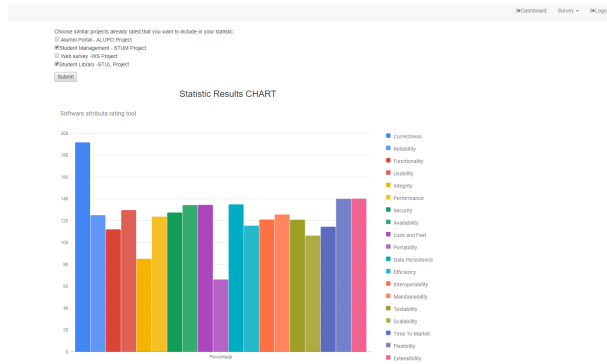
FIGURE 7. Statistic final results

The results are displayed as a statistics, but they are adaptable if the manager wants to add rates from previous projects or if he / she wants to have different weights for user type importance. This importance can be applied to all projects or just to the current project and get the rates as they were from other projects.

3. CONCLUSION

We have created a web application rating tool, which can be used inside a company for getting feedback from the people involved in a project about the attributes importance that should be taken into consideration when evaluating the product. The rating tool can use the experience form previous projects and also provides a better statistics idea by the usage of the fuzzy logic in the user type evaluation criteria. Bu using this tool, we do not only keep track of the rating importance in different projects, but we also create a more precise statistics for the attribute importance.

REFERENCES

[1] Azzeh, M., Neagu, D. and Cowling, P. I., *Analogy-based software effort estimation using Fuzzy numbers*, J. Systems and Software **84** (2011), no. 2, 270–284

[2] Hajdu-Măcelaru, M., *Software Quality Evaluation Using Fuzzy Theory*, Sci. Stud. Res. Ser. Math. Inform. **23** (2013), no. 1, 49–53

[3] Hajdu-Măcelaru and M., Zelina, I., *An adaptable software quality model*, Creat. Math. Inform. **27** (2018), no. 1, 31–35

[4] Huang, S-J. and Chiu, N-H., *Applying fuzzy neural network to estimate software development effort*, Appl. Intell. **30** (2009), no. 2, 73–83

[5] Féris, M. A. A. and Zwikael, O., Gregor, S., *QPLAN: Decision support for evaluating planning quality in software development projects*, Decision Support Systems, **94** (2017), 92–102

[6] Ivan, I., Zamfiroiu, A., Doinea, M., Despa, M. L., *Assigning Weights for Quality Software Metrics Aggregation*, Procedia Computer Science, **55** (2015), 586–592

[7] Pinciroli, F., *Improving software applications quality by considering the contribution relationship among quality attributes*, 3rd International Workshop on Computational Antifragility and Antifragile Engineering, 2016

DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE
NORTH UNIVERSITY CENTER AT BAIA MARE
TECHNICAL UNIVERSITY OF CLUJ-NAPOCA
VICTORIEI 76, 430122 BAIA MARE ROMANIA
Email address: mara.macelaru@cunbm.ro