An Identification of Software Requirement for Web Based Judging System

N.A.A.Zahari, R.AJM.Gining, S.S.M.Fauzi*, T.R.Razak and M.N.F.Jamaluddin
Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Perlis Branch, Malaysia.

shukorsanim@perlis.uitm.edu.my

Abstract. The paper aims to identify the requirement for the judging system. The requirement identification of the system involves two phases; requirement gathering and requirement analysis. Several activities are done in each of the phases and the result is discussed. This study translated the requirement gathered from the interview and document analysis into several diagrams; use case diagram, activity diagram and entity relationship diagram.

1. Introduction

Every product in an innovation competition needs to be evaluated. The assignation of judges to the products or booths is done a few weeks before the competition. During the day of the innovation competition, the judges need to identify themselves to the secretariat, and they will be briefed and provided with evaluation forms for their assigned product. After the briefing, all judges start to conduct the judging processes. by visiting their assigned booth and filling in the evaluation form. Once the judging process is completed, the judges need to return the evaluation form to the secretariat. The secretariat then enters the suggested scores by the judges into a selected platform - usually Microsoft Excel. Finally, all the scores are sorted to identify the winners.

The above-mentioned process will lead to several issues and drawbacks. Firstly, during the judging process, judges had to carry several evaluation forms on hand and sorting them would be hard, as they need to go through all of the forms to assign scores. Sometimes, the manually written scores are misunderstood by the secretariat as the judges would have varied writing styles. There is also a possibility that the evaluation form will be misplaced, damage and loss. Secondly, other than needs more time and resources, manual data entry is also burdensome [1]. Moreover, the problem of missing data from its records has been proven as a main problem for this approach [2-5]. The data entry from physical to electronic undertaken by a human is also a slow. If this situation continues, a massive delay in announcing the winners in innovation competition surely will occur.

Therefore, to improve the judging process, a web-based judging system is developed. The objective of the paper is to identify the software requirements for the judging system. The software requirement identification is a crucial process to ensure the success of the system. The requirements are
presented in a form of diagrams as it will be used as the main source for reference in the next phase - the development phase.

This paper is alienated into several sections. The first section introduces the issues and motivation of the study. Followed by the second section which discusses the methodology used. Results and discussion are presented in the third section and the paper is concluded in section 4.

2. Methodology

The methodology used in the system requirement identification involves two phases which are requirement gathering and requirement analysis. Each phase consists of activities that lead to the requirement collection and affirmation before the development process.

2.1. Requirement Gathering

The first phase focuses on the interview activity, which conducted among several experienced innovation competition committee and judges. An information-gathering interview is a dialogue with a specific goal that uses a question-and-answer format. A recorded formal or informal interview can be used for study purposes. The interview content may comprise of a semi-structured and open-ended question which to be answered by the respondent. The purpose of conducting the interview is to gather information about the current judging system used in an innovation competition. The judges for innovation competition who are responsible for giving scores to the participants were interviewed to gain an in-depth understanding of the current system used in the judging process. These involve the flow of the judging process and the problems that judges frequently face. A face to face conversation is preferred in this activity since it allows the respondent to answer the question spontaneously and any unclear information could be clarified immediately.

Other than the interview, the activity of analysing existing documentation has also been done in a way of obtaining an evaluation form used by the respondents in an innovation competition. Documentation assists the requirement identification process in term of understanding the overall flow of the judging process. Information required such as the product, judges, and the evaluation criteria are also analysed and extracted from the available documentation.

2.2. Requirement Analysis

The second phase of the method is the Requirement Analysis phase. This phase utilises several modelling tools and diagram such as Use Case Diagram, Activity Diagram, and Entity Relationship Diagram. These diagrams are used to visualise the user requirement compiled in the first phase for better understanding. Specifically, it is developed to aid the developers to visualize, specify, document and construct the artefacts of the system to be developed.

2.2.1. Use Case Diagram

A use case is a methodology used in to recognize, elucidate, and organize system requirements [6]. The use case consists of a set of possible interactions flows between users and systems in a specific environment and associated with a specific goal. It is composed of three main elements; actors, use case and association.

2.2.2. Activity Diagram

Activity Diagram purpose is to model the step-by-step actions taken in a portion of a larger activity [7]. Activity Diagram is essentially an improved version of the flow chart that demonstrate the flow from one activity to another activity. By explaining the flow of actions in a process, it also indirectly gives a view of the system behaviour. An Activity Diagram is denoted by shapes that are linked by arrows that illustrate the sequential order of executed activities.

2.2.3. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) is also known as an entity-relationship model, is a graphical representation of an information system that portrays the
relationships among objects, people, concepts, places, or events within that system [8]. In Database Management System (DBMS) terminology, an entity is a table or attribute of a table in a database, thus by displaying the relationship between tables and their attributes, ERD illustrates the complete logical structure of a database. ERD also can be used as the foundation for a relational database, which is the type of database that will be used in the development process.

3. Result and Discussion
The results of the interview sessions indicate most of the judging approach used by the respondents in their participated innovation competitions are still using the manual approach - using paper-based evaluation form to record scores and then the responsible judging committee would transfer the score to their selected platform - usually to Microsoft Excel. This approach could increase the tendency for errors, mostly on the committee and such as mistype or inserting data in the wrong column, and they might misunderstand the score given by the judges as a manually written score sometimes lack clarity. Not only that, the possibility that the evaluation form to be misplaced exists, in this case, the judges may need to re-evaluate the innovation product and the result may be different from the score given previously.

Therefore, the respondents prefer an easy alternative for judging which could reduce their workload on evaluating and calculating the score for each participant or booth. The use of an information system is highlighted in some of the interviews as it could help to store the needed data and calculate the score automatically.

Through the review of the input gathered from the interview sessions and document analysis, the requirement for the judging system has been determined. The requirements are analysed and translated into modelling tools diagrams; Unified Modelling Language (UML) diagram, Entity Relationship Diagram (ERD) and sitemap. The UML diagram presents both the Use Case Diagram and Activity Diagram.

The requirement analysis phase classified the users’ requirements into two; functional and non-functional requirements. Listed below are the requirements for the system. Each requirement is assigned with an ID and in the priority column, the following abbreviations are used:

M – Mandatory requirements (something the system must do)
D – Desirable requirements (something the system preferably should do)
O – Optional requirements (something the system may do)

| No. | Requirement ID | Requirement Description | Priority |
|-----|----------------|-------------------------|----------|
| JS _01 | Login into the system (Judge and Admin) | Judge and admin must enter email and password to login. | M |
| JS _01_01 | | Message will be prompted if email and password are not match. | D |
| JS _01_02 | Could click on Forgot Your Password if user forgot their password. | O |
| JS _02 | View All Group (Admin) | Admin could display the list of groups that participate in the innovation competition. | M |
| JS _02_01 | | Admin could edit the group name, product name, product type and group member. | D |
| JS _02_02 | | Admin could add group member by clicking on Add Group Member button. | M |
| JS _02_03 | | Admin could select product type by using drop down list. | D |
| JS _02_04 | | Message will be prompted when the number of group member exceed five. | O |
| JS _02_05 | | |

Table 1. Requirement of the judging system.
| No. | Requirement ID | Requirement Description | Priority |
|-----|----------------|-------------------------|----------|
| 9.  | JS_02_06       | Admin could add new group by clicking Add Group. | M        |
| 10. | JS_02_07       | Alert message will be prompted when blank mandatory field detected (Group name, Product type and Product name) | O        |
| 11. | JS_02_08       | Admin can delete group record by clicking Delete Group button. | M        |
|     | JS_03          | View All Booth (Admin) |          |
| 12. | JS_03_01       | Admin could display the list of booth number as well as the product assigned to it. | M        |
| 13. | JS_03_02       | Admin could edit the booth number and product name. | D        |
| 14. | JS_03_03       | Admin could add booth by clicking on Assign Booth. | M        |
| 15. | JS_03_04       | Alert message will be prompted when blank mandatory field detected (Booth number, Product Name). | O        |
| 16. | JS_03_05       | Admin can delete booth record by clicking Delete Booth button. | M        |
|     | JS_04          | View Assigned Judge to Booth (Admin) |          |
| 17. | JS_04_01       | Admin could display the booth number along with the list of assigned judges and group name. | M        |
| 18. | JS_04_02       | Admin could edit the assigned judge. | D        |
| 19. | JS_04_03       | Admin could add judges to booth. | M        |
| 20. | JS_04_04       | Admin could select judges by using drop down list. | D        |
| 21. | JS_04_05       | Admin could view the group details during assigning the judge. | D        |
| 22. | JS_04_06       | Admin could delete the assigned judge. | M        |
|     | JS_05          | Register New User (Admin) |          |
| 23. | JS_05_01       | Admin could register new user (Admin and Judge). | M        |
| 24. | JS_05_02       | Display message for successful registration. | D        |
|     | JS_06          | View Assigned Booth (Judge) |          |
| 25. | JS_06_01       | Judge could view the assigned booths. | M        |
| 26. | JS_06_02       | Judge could access the evaluation form for each group. | M        |
| 27. | JS_06_03       | Judge could view the rubric during the evaluation process. | M        |
| 28. | JS_06_04       | Judge could view the scores they gave to each group everytime they access the evaluation form. | D        |
| 29. | JS_06_05       | Judge could edit the score they gave in the evaluation form. | D        |
|     | JS_07          | Report (Admin) |          |
| 30. | JS_07_01       | Admin could view the scores obtain by all groups. | M        |
| 31. | JS_07_02       | Admin could view the list of groups that already sorted by higher score to lower. | M        |

**B. NON-FUNCTIONAL REQUIREMENTS**

| No. | Requirement ID | Requirement Description | Priority |
|-----|----------------|-------------------------|----------|
| 32. | JS_08          | Usability Issues | M        |
| 33. | JS_09          | Efficiency Issues | D        |

The requirement ID is used for identification based on the activities done in the system. JS in the ID is the abbreviation of Judging System and the numbering of the ID is provided as in number increment. The priority classification of each requirement is determined from the result of the interview. The non-functional requirements centre around usability and efficiency but mostly it is just preferred features which are not mandatory to the system.
A use case is a methodology used in to identify, interpret, and organize system requirements [7]. There are two actors involved in this system; Admin and Judge. Admin is referring to the secretariat of the innovation competition. The Admin actor is associated with six use cases; 1) login into the system, 2) register new user, 3) manage the innovation booths, 4) manage group, 5) manage judge and 6) view all scores obtained by all groups. On the other hand, the Judge actor is connected to four use cases; 1) login into the system, 2) fill in the evaluation form, 3) view rubric and 4) view list of assigned booths. The figure below shows the Use Case Diagram of the judging system.

![Use Case Diagram for the judging system](image)

**Figure 1.** Use case diagram for the judging system.

The Activity Diagram shows the detailed process for both actors (Admin and Judge) interacting with the system. For the sake of simplicity, only two Activity Diagram is included in this paper - Login and Adding New Group diagram.
The figure above illustrates the process occurs when both actors login into the system. After the actors click on the login button, the page for the user to enter their email and password will appear, which is the Login Page. The actors need to enter a correct combination of email and password to be directed to the Home Page. If any of the information provided is incorrect, an error message will appear, and user needs to re-enter the correct email and password.

The figure below shows the processes occurs during the creation of a new group or participant in the innovation competition. This process is done by the admin. When Admin clicks on Add Group menu, the corresponding form will be displayed. Several data needed to be entered such as the group name, product name, product type and the group members before submitting the form. There is a restriction for the numbers of group member - which is limited to five members, adding more and an error message will appear. After the Admin submits the form by clicking on the Add Group button, the system will check for validation, if all the data is valid, it will be saved in the storage and the Admin will be shown a 'successful' pop-up message, otherwise, the Admin need to re-enter any invalid data.
Figure 3. Activity diagram for add group.
The figure above shows the Entity-Relationship Diagram (ERD) of the judging system. It consists of nine entities and two of these are a bridge entity. The Booth_User table is the bridge between Booth and User table while Criteria_Scale table is the bridge between Criteria and Scale table. Both of these tables become a pivot table or bridge because of a many-to-many relationship. Every table has its own primary key which is id except for Booth_User table that have two primary keys, user_id and booth_id.

The relationship between the Group and Group_Member is one-to-many because one group can have many members and one member can only have one group. Group and Product have a one-to-one relationship because one group can own one product and one product can only belong to one group. Group and Booth also are connected with one to one relationship. This is due to one group can be assigned to one booth and one booth can only have one group. The relationship between Booth and User is many to many therefore there is a pivot table named Booth_User. This is because one booth can be assigned to many judges and one judge can be assigned to many booths. Criteria and Scale are connected with a many-to-many relationship, therefore there is a pivot table named Criteria_Scale. This is due to one scale can be assigned to many criteria and one criterion can have many scales. Booth_User and Criteria_Scale have a one-to-many relationship because one judge can insert scores into many evaluation forms while one evaluation form can only be filled by one judge.

These requirements are the main foundation of the judging system that will be developed in the future as it acts as a communication tools and references among the developer and designer of the system. The success and failure of the development process rely on the requirement collected during the requirement identification process.

4. Conclusion
The main purpose of this study is to identify the software requirements for the web based judging system. 31 functional and 4 non-functional requirements were identified. The identified requirements were classified into 9 use cases, and 2 actors. The proposed ERD consists of 9 entities, which two of them are bridge entities. Future works would be the development of the proposed system based on the identified requirements.
References

[1] Vlada, M., Babiy, I., & Ivanescu, O. (2010). ABBYY recognition technologies – ideal alternative to manual data entry. Automating processing of exam tests. Star, 3(1), 3–8.

[2] Tufo, H. M., & Speidel, J. J. (1971). Problems with medical records. Medical care, 509-517.

[3] Dawes, K. S. (1972). General Practice Observed: Survey of General Practice Records. Br Med J, 3(5820), 219-223.

[4] Zuckerman, A. E., Starfield, B., Hochreiter, C., & Kovasznay, B. (1975). Validating the content of pediatric outpatient medical records by means of tape-recording doctor-patient encounters. Pediatrics, 56(3), 407-411.

[5] Romm, F. J., & Putnam, S. M. (1981). The validity of the medical record. Medical care, 310-315.

[6] Rouse, M. (2007, April). use case. Retrieved from https://searchsoftwarequality.techtarget.com/definition/use-case

[7] Bell, D. (2003). UML basics Part II: The activity diagram. IBM Global Services, Rational Software.

[8] Rouse, M. (2018, July). entity relationship diagram (ERD). Retrieved from https://searchdatamanagement.techtarget.com/definition/entity-relationship-diagram-ERD