

Analysis and Design Logbook Information Systems

Siti Mukaromah, Prisa Marga Kusumantara, Agung Brastama Putra, Arista Pratama

Dept. of Information System, Faculty of Computer Science
Universitas Pembangunan Nasional “Veteran” Jawa Timur
Surabaya, Indonesia
sitimukaromah.si@upnjatim.ac.id

Abstract—Laboratory management must be taken seriously and regularly, such as recording items in and out, students who enter to do scheduled lab work or free practicum and lecturer activities in conducting research can be recorded in a system. For this reason, an information system is needed to handle the recording of every visitor entering the laboratory. In this paper, the scope to be discussed was the analysis and design of a laboratory logbook.

Keywords—logbook, analysis, design system.

I. INTRODUCTION

A College in supporting the learning process requires a Laboratory. Based on government regulation number 19 of 2005 concerning national education standards, states that each department must have a laboratory as a means and infrastructure to support the learning process. Whereas based on the 2007 Minister of Education Regulation No. 24 the laboratory is a place used to develop skills.

Therefore, the Laboratory is a vital place in the development of science/research even in supporting the interaction of learning and teaching between students and lecturers so that they are expected to be able to produce a product.

Laboratory management must be carried out thoughtfully and regularly, such as recording of incoming and outgoing goods, students entering to do scheduled lab work or free practicum and lecturer activities in conducting research can be recorded in a system.

In laboratory management, it covers several aspects, i.e. planning, structuring, administering, as well as security, maintenance, and supervision:

With so much scope and the relationship between laboratories, teaching and learning activities, a system is needed that can be used for administration or recording data of goods, lecturers, and students who will or are doing activities in the laboratory so that all activities will be monitored. The scope of this research consists of the student/lecturer logbook sub-module for entering and exiting the laboratory, practicum student data and laboratory administration data that is used to record item data. This research will be tested in 3 laboratories in the Information System Department, Engineering Department Laboratory Informatics and Laboratory of Visual Communication Design Department at Universitas Pembangunan Nasional “Veteran” Jawa Timur.

Planning is the starting point of laboratory management to be professional and well organized. According to KBBI planning is a process of thinking and developing a dynamic object to achieve effective and efficient growth. Preparation

will involve structuring and administration; these three things are the first steps that must be carried out in conducting laboratory management and security, maintenance and supervision are processes that will protect the three elements.

Kartu Tanda Mahasiswa (KTM) or student ID is an identity card that shows that the cardholder is an active student, by maximizing the KTM that is owned by students, when students enter the laboratory must and must bring KTM and use it for attendance. Related to the above, the formulation of the problem in this study is how to analyze and design a logbook information system that uses KTM and barcode reader in the Information Systems Department.

The application of IT help companies to help some problems, the use of IT has been able to provide business strategic decisions support. [2]

II. METHOD

In this research using Iconix Process with the Unified Modeling Language (UML) approach and using visual studio for applications. ICONIX Process is a “cookbook” process in that it describes a series of specific steps that found work really well on many different projects [2, 3, 4].

ICONIX process is also ever been implemented before by some researcher like Honda [6] which it's title is Goal-Oriented Robustness Analysis. That paper presents an approach to transfer lots of information from KAOS requirements models to preliminary design of system behaviors in ICONIX. Their approach allows the use case models and robustness diagrams to reflect the requirements of the KAOS model with the refinement patterns. The other researchers also implement ICONIX process to design their software development. [5, 7, 8, 9, 10]

A. Method of Collecting Data

At this stage, data collection is take by interviewing software actors including lecturers, students and laboratory assistants.

B. Analysis Method

At this stage the following steps will be taken:

- Analysis of problems (required techniques etc.) in designing logbook information systems,
- Design the global architecture of the system to be built,
- Designing databases and user interfaces from information systems.
- Preliminary Design is the stage between analysis and design.

C. Iconix Process

Iconix Process is a minimalist and efficient approach that focuses on the area that lies between the use case and the code. The emphasis is on what needs to happen in the life cycle [3].

The ICONIX process is divided into dynamic and static workflows, which are very iterative. It can be through one iteration from the whole process for a small group of use cases to the source code and unit tests. For this reason, the ICONIX process is very suitable for agile projects, where fast feedback is needed on factors such as requirements, design, and estimates.

Iconix Process is a scenario-based approach; The main mechanism for decomposing and modeling systems is based on scenarios per scenario. But when using the ICONIX Process, the goal is to produce object-oriented designs that can be encoded. Therefore, it is necessary to connect the scenario to the object, by writing the use case using the domain model created.

III. RESULT AND DISCUSSION

In the results and the following discussion explains the results of the analysis and design of the system being built, the discussion in this chapter consists of database design and design of the Logbook Information System.

A. Database Design

The designed database design begins with Conceptual Data Model (CDM) and Physical Data Model (PDM) [6].

1). CDM

CDM is used to describe in detail the structure of the database in logical form, and still in the form of design, can be seen in the figure 1.

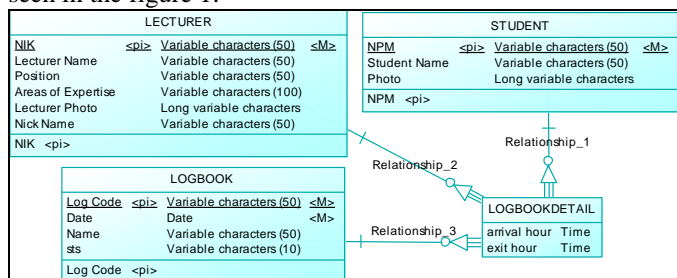


Fig. 1. CDM of Laboratory LogBook

As seen in the Figure 1 CDM of the Logbook Information System, in the CDM above, there is a relationship between tables, related tables are lecturers' tables with subjects and laboratory tables.

The connection between lecturer tables, courses and laboratories makes it easy to make the system or even develop the system in the future, some of the cardinality used in CDM in this study is one to many mandatory and one to many mandatory dependent.

2). PDM

PDM is a display with database details in physical form. The depiction of the PDM design shows the correct data storage structure in the database that is actually used (database), as seen in the figure 2.

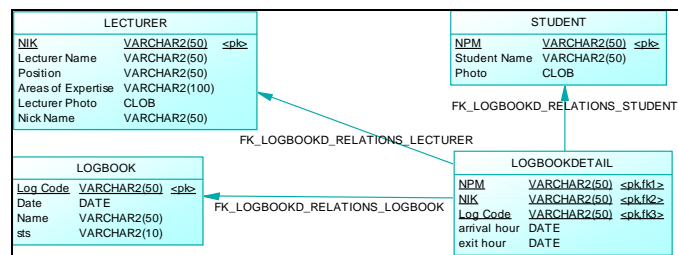


Fig. 2. PDM of Laboratory LogBook

As seen in Figure 2 is a physical display of the database that will be implemented, with the primary key and foreign key connections that are connected it will maximize the programmer when creating the database.

B. Domain Model

Domain Model is a task to build a project glossary, or a term dictionary used in a project (for example, an Internet bookstore project includes domain objects such as Books, Customers, Sequences, and Ordering Items). The aim is to make sure everyone in the project understands space issues in unambiguous requirements. The domain model for a project defines the scope and forms the foundation for building your use case. The domain model also provides general vocabulary to allow clear communication between project team members. [2]

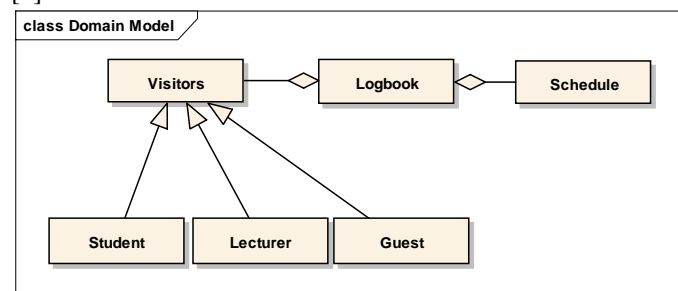


Fig. 3. Domain Model LogBook

As seen at Figure 3 it describes the Laboratory Logbook domain model where the terms that would be used in this study are visitors, logbook, and schedule. Visitors can be divided into 3 types of visitors, namely students, lecturers and guests. How to read this domain model is logbook has a schedule; logbook has a visitor; student is a visitor, lecturer is a visitors, guest is a visitors.

C. Use Case Model

Use cases describe how users interact with the system and how the system will respond.

At iconix process use case is divided into 2, namely use case text and use case diagram. Use case text will function as a specification of the runtime behavior shown in the diagram, in the form of text. The rules for writing a use case text are using 2 scenarios, a successful scenario (Basic Course) and a failed scenario (Alternate Course). Use case text exists from the use case model, model robustness, and model sequences.

1). Use Case Text

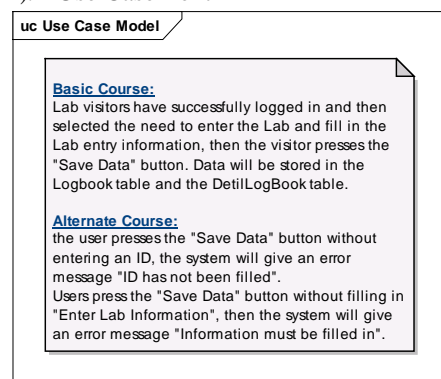


Fig. 4. Use case text Laboratory LogBook

As seen at Figure 4 is the use case text which is the narration of the use case diagram. Figure 4 explains there are two scenarios (Basic Course and Alternate Course) running on the system. The basic course is a scenario path if the steps for using the system are running correctly, that is, the user filled in the logbook that was previously successfully logged. Users (in the system called visitors) the system is divided into 3, namely students, lecturers, and guests.

Alternate course is a scenario path if the system usage steps are not appropriate. In Figure 4 explains if visitors fill a logbook without logging in first, the system will respond by displaying an error message. Another error scenario is that when the user does not fill in the laboratory entry information column, the system will respond by displaying an error message.

2). Use Case Diagram

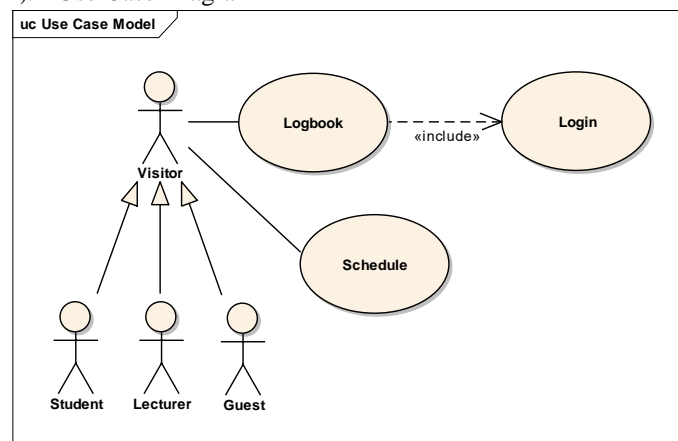


Fig. 5. Use case Diagram Laboratory LogBook

As seen at Figure 5 illustrates the use case in the logbook information system. As seen at Figure 5 shows the process that will be handled in the system is the login process and the logbook data storage process.

D. Robustness Model

Robustness Analysis used to get from use case to detailed design (and then to code), need to connect use case to object. Robustness Analysis helps bridge the gap between analysis and design by doing so. This is a way to analyze the use case text and identify the first set of objects for each use case.

A robustness diagram is an image of an object from the use case. The diagram of robustness and use case text must be appropriate, so the robustness diagram forces to tie the use case text to the object. This makes it possible to drive object-oriented designs from the use case.

Robustness Diagrams are used to check again that all possible actions have been discussed.

1). Robustness Diagram Login

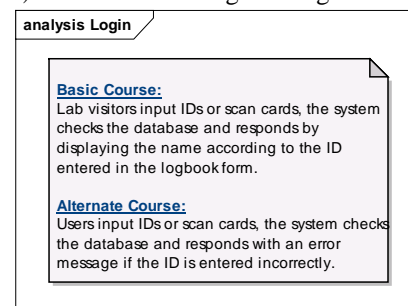


Fig. 6. Use Case Text Robustness Diagram Login SI LogBook

As seen at Figure 6 is a use case text from a robustness diagram for a login use case. The Basic course describes the login success scenario. Whereas the alternate course explains if the entered ID is not suitable, the system will respond by displaying an error message.

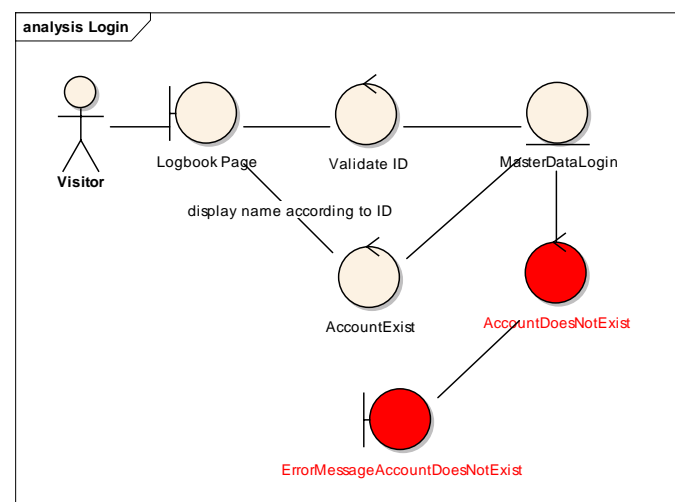


Fig. 7. Robustness Diagram Login Laboratory LogBook

As seen at Figure 7 illustrates the robustness of the login process, where images with transparent background are basic course depictions (successful scenarios) while symbols that use red background are alternate courses (failed scenarios). The basic course in figure 7 shows if the system finds in the visitor data database based on the ID entered, the system will display the visitor's name according to the ID on the logbook page. Whereas if a visitor enters the wrong ID or the entered ID is not in the database, the system will display an error message.

2). Robustness Diagram Logbook

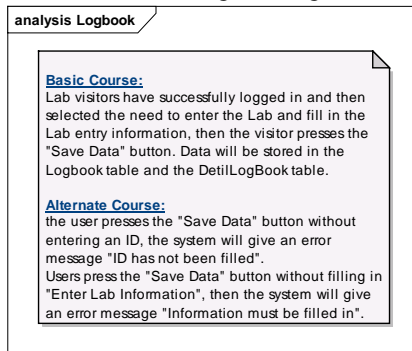


Fig. 8. Use Case Text Robustness Diagram Logbook

As seen at Figure 8 shows the use case text of a robustness diagram for a use case logbook. The basic course shows that after logging in, visitors can fill in the laboratory entry information and save the filled data, the system will store the data that is loaded from the logbook form to the database, in the logbook table and Logbook table details.

The Alternate Couse in figure 8 explains if the user immediately presses the save button without logging in first, the system will respond with an error message. The second alternate course if the user has successfully logged in but the laboratory entry information field is forgotten, the system will respond with an error message.

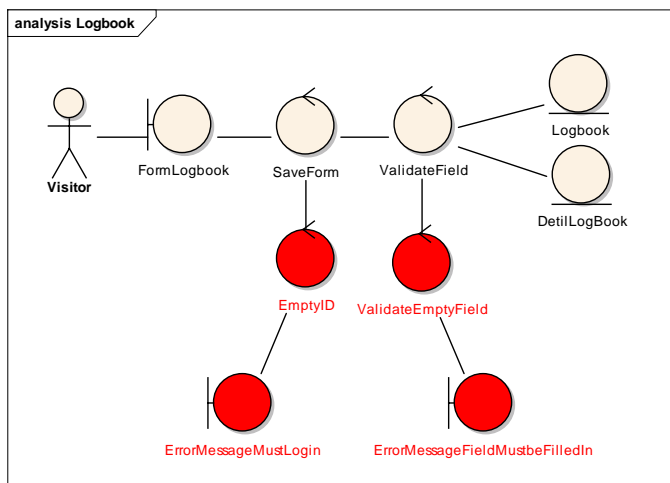


Fig. 9. Robustness Diagram Logbook LogBook

Figure 9 illustrates the robustness diagram of a use case logbook, where images with transparent background are basic course drawings (successful scenarios) while symbols that use red background are alternate courses (failed scenarios). The basic course in figure 9 shows that if a visitor has successfully logged in then they can enter the laboratory entry information and press the save data button, then the system will save the data loaded from the logbook form to the database in the logbook table and Logbook table details.

Whereas if the user immediately press the save button without logging in first, the system will respond with an error message. Alternate course (symbol with red background color) the second if the user has successfully logged in but the laboratory entry information field is forgotten, the system will respond with an error message.

E. Sequence Model

The Iconix process uses sequence diagrams as the main vehicle to explore system detail design in scenario-by-scenario. In object-oriented design, most system buildings are correctly related to finding the optimal allocation of functions to the class (aka behavior allocation). The point of this is to draw message arrows in sequence diagrams and allow modeling tools to automatically assign operations to the target object class that receives runtime messages.

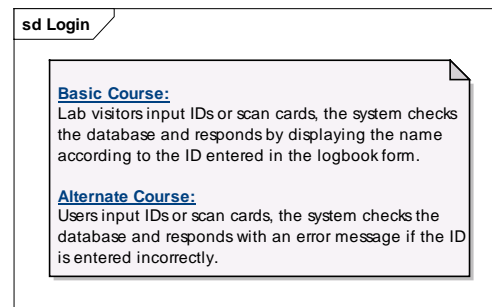


Fig. 10. Use Case Text Sequence Diagram Login LogBook

As seen at Figure 10 shows the use case text from sequence diagrams for login use cases. The Basic course describes the login success scenario. Whereas the alternate course explains if the entered ID is not suitable, the system will respond by displaying an error message.

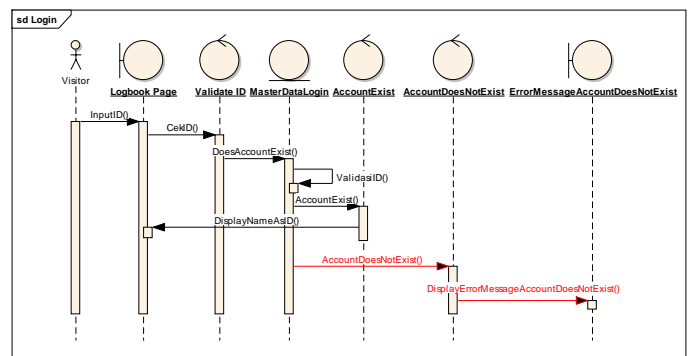


Fig. 11. Sequence Diagram Login LogBook

Figure 11 illustrates sequence diagrams for the use case logbook. The process begins with the user inputting the ID or by scanning the card on the logbook form, then the input is handled by the ID validation control which checks the input ID with the MasterDataLogin table. After the data is found, the name of the ID is displayed on the logbook form.

If the ID is not found in the database, the control will display an error message that the ID entered is incorrect. In iconix process the step is given a red color.

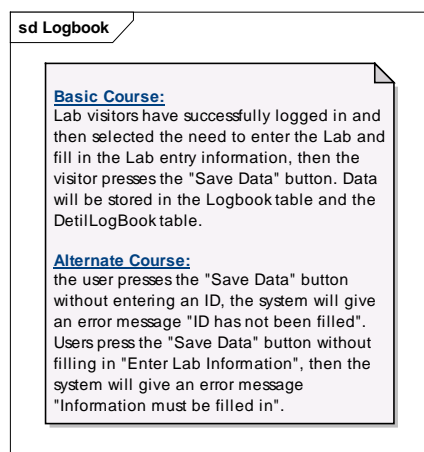


Fig. 12. Use Case Text Sequence Diagram Logbook

Figure 12 shows the use case text of the sequence diagram for the use case logbook. The basic course shows that after logging in, visitors can fill in the laboratory entry information and save the filled data, the system will store the data that is loaded from the logbook form to the database, in the logbook table and Logbook table details.

The Alternate Couse in Figure 12 explains if the user immediately presses the save button without logging in first, the system will respond with an error message. The second alternate course if the user has successfully logged in but the laboratory entry information field is forgotten, the system will respond with an error message.

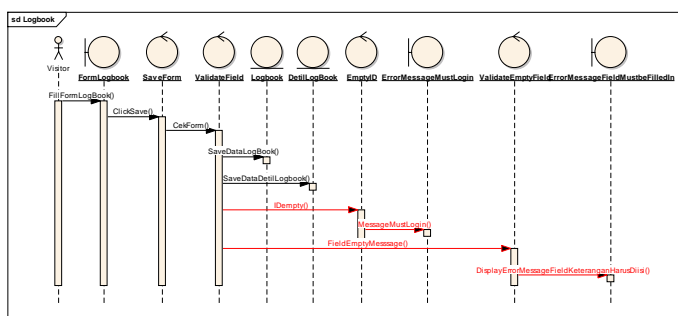


Fig. 13. Sequence Diagram Logbook LogBook

Figure 13 illustrates the use case logbook sequence diagram. The process begins with the user logged in successfully (for the login sequence in Figure 11) which then fills in the information on the requirements for entering the laboratory in the logbook form. After filling in the next visitor press the save data button, then the process will be handled by the SaveIsiForm control which is then validated in the fields that must be filled by the ValidationField control. After all fields are filled in, the data will be saved to the database in the Logbook table and LogBook details table.

While the red line that describes the alternate course. There are 2 alternate courses that may occur, the first is when the visitor without filling in the data login ID or directly filling out the laboratory entry information field and pressing the save button, then the IDKosong control will command to display an error message that the visitor must fill in the ID first.

The next alternative course is when the visitor has logged in but still has not filled in the laboratory entry information field, then the ValidateFieldKosong control will display an error

message that the laboratory entry information field must be filled.

F. Graphical User Interface



Fig. 14. GUI main menu Logbook

Figure 14 shows the main menu of logbook, there are 3 menus at this form. “Log Tamu” means if any visitor who enter the laboratory., “Log Dosen/Mahasiswa” means if any lecturer or students attend class at laboratory, “Cetak Data” means print information from this application.

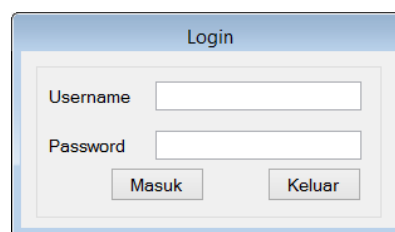


Fig. 15. GUI Login

Figure 15 shows that administrator who wants to add data to database, so that everyone who scan/input their ID at this application could be save as Laboratory Log.

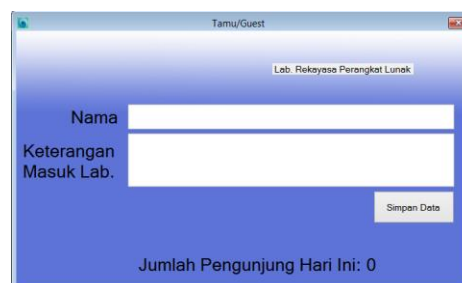


Fig. 16. GUI Guest Log

Figure 16 is the user interface when guests enters the Laboratory, because their data are not exist on database unless they save their data each time they want to attend some occasion at Laboratory.

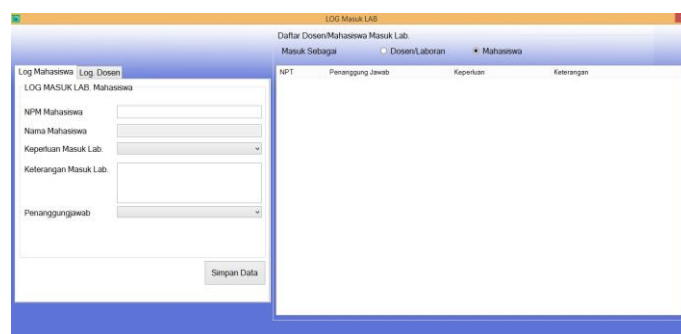


Fig. 17. GUI Lecturer or student Log

Figure 17 is form whether students or lecturers should fill this form before enters laboratory. This form will save to data store the information the purpose of students and lecturers enters Laboratory.

Fig. 18. GUI Lecturer Master Form

Figure 18 is form for administrator who wants to save the master data from Lecturers and Laboran (person who assist lecturers at Laboratory)

IV. CONCLUSION

The conclusion of this study is that the analysis and design with Iconix Process using the UML approach has been

completed which is then continued in the application development process with visual studio .Net 2005.

REFERENCES

- [1] S. Mukaromah, and A.B. Putra. "Maturity Level at University Academic Information System Linking it Goals and Business Goal Based on Cobit 4.1." MATEC Web of Conf. Vol. 58. EDP Sciences, 2016.
- [2] D. Rosenberg & M. Stephens, "Use Case Driven Object Modeling with UML: Theory and Practice", Inc, 2007.
- [3] D. Rosenberg, M. Stephens, & M. Collins-Cope, "Agile development with ICONIX process", 2005.
- [4] D. Rosenberg & K. Scott, "Driving Design with Use Cases", 2000.
- [5] M. Collins-Cope, M. Stephens, & D. Rosenberg, "Agile development with the ICONIX process: people, process and pragmatism", 2005.
- [6] K. Honda, H. Nakagawa, Y. Tahara, & A. Ohsuga. "Goal-Oriented Robustness Analysis." IosPress, 2016.
- [7] A. Sidky, J. Arthur, & S. Bohner, "A disciplined approach to adopting agile practices: the agile adoption framework". Innov. in syst and softw. eng., vol. 3(3), pp. 203-216, 2007.
- [8] A. Qumer, & B. Henderson-Sellers, "A framework to support the evaluation, adoption and improvement of agile methods in practice." J. of Syst. and Softw., vol. 81(11), 1899-1919, 2008.
- [9] M.El-Attar, & J. Miller, "Developing comprehensive acceptance tests from use cases and robustness diagrams". Requir. Eng., vol.15(3), pp 285-306, 2010.
- [10] N. Elmasri., "Fundamentals of Database Systems", Sixth Ed. Pearson Edu., Inc, 2011.