PAPER • OPEN ACCESS

Software testing on e-learning Madrasahs using Blackbox testing

View the article online for updates and enhancements.

Software testing on e-learning Madrasahs using Blackbox testing

S Sutiah[1,*] and S Supriyono[2]

[1]Department of Islamic Education, Faculty of Tarbiyah and Teaching Training, Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang, Indonesia
[2]Department of Informatics Engineering, Faculty of Science and Technology, Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang, Indonesia

*sutiah@pai.uin-malang.ac.id

Abstract. Distance learning in the pandemic era of COVID 19 must meet the needs of the system. E-Learning Madrasah is an application program developed from the Ministry of Religious Affairs of the Republic of Indonesia. The learning that can be done in the pandemic era of COVID 19 corresponds to government directives. E-Learning Madrasah has an essential role in distance learning at the basic education level up to secondary education. To improve the software functionality required testing system. The following research uses the Blackbox Testing approach to e-learning Madrasah software testing. The testing scenario includes several scenarios, including login form testing, class setting testing, computer-based test testing, student learning evaluation testing. The scenario can be used to test functionality so that the recommendation results can be used to develop e-learning madrasah in the future. The resulting results showed e-learning Madrasah using the BlackBox testing approach to testing software functionality levels with the need to reach 90%. These results show from the experiment of some of the scenarios.

1. Introduction
Distance learning is fundamental in the era of the COVID 19 pandemic. The success of educators in the COVID 19 pandemic can be measured by how effective and efficient the learning method is using e-learning. Related research discusses the use of e-learning at Senior High School in Bengkulu City [1]. Related research uses Moodle to improve performance on e-learning. The development of technology and the world of education today encourages innovation and development to get knowledge quickly without limitation of place and time. One of the good breakthroughs in the world of education is the existence of e-learning or electronic learning media, with the development of the internet and the ease of information, e-learning is now quite common as a means of getting knowledge quickly and easily. The use of Hybrid learning can increase the use of e-learning in improving learning technology [2].

E-learning Madrasah is a free application for Madrasah products that are intended to support the learning process in Madrasas from Raudatul Athfal (RA), Madrasah Ibtidayah (MI), Madrasah Tsanawiyah (MTs) and Madrasah Aliyah (MA), to make it more structured, attractive and interactive. E-learning Madrasah contains content starting from the beginning of the learning process, making

competency standards, learning implementation plans, learning materials, teacher journals, daily assessment processing, computer-based exams to processing report cards. Core competencies in student learning can be measured according to the competency level needs of students [3]. To produce more optimal software, appropriate technology that can be used in distance learning is needed. The optimal results can be seen from the results of the system testing that has been done. For more optimal testing, the BlackBox testing method is needed.

The development of learning media is an essential aspect of learning technology. In testing the learning media application, it has a pretty good level of accuracy [4]. Related research uses design-based research (DBR) method in integrating technology in a lot of interactivity in a student's class [5]. Web-based application testing uses a testing scenario with various parameters by evaluating the use of genetic algorithms used to generate testing requirements [6]. Software testing is used to measure the level of complexity of an application program used includes functional and non-functional testing [7]. Software testing has been discussed in research related to using tools that aim to verify software that has been developed [8].

The software testing that is carried out imperfectly will certainly have an adverse effect on the quality of the software produced. Ineffective and incomplete software testing can cause various problems when the software is used by end-users. It can automatically improve the efficiency of the testing process to identify which parts of the software are prone to failure. In the following research, testing the E-Learning Madrasah software with the BlackBox Testing approach. Based on the method used, it will be known weaknesses in the software after testing using the Black Box method and how to find out which output is considered valid.

2. Blackbox testing

System testing aims to see whether the system that has been created is by the original purpose of manufacture and is suitable for use. Testing on the system uses the BlackBox Testing method. The goal is to determine that the parts in the application system correctly display error messages if an error occurs in data input.

2.1. Equivalence partitioning

The following testing techniques are used to reduce the number of test cases. The test parameter is obtained from the input value in each part of the tested domain [9]. Equivalence Partitioning Testing is a method of testing software with the BlackBox Testing approach that breaks or divides the input domain of the program into data classes so that test cases can be obtained. The design of test case equivalence partitioning is based on the evaluation of equivalence classes for input conditions that describe a set of states that are valid or not. The input condition can be a numeric value, a range of values, a related set of values , or a Boolean condition.

2.2. Boundary value analysis / limit testing

The Boundary Value Analysis method is a method of detecting errors by limiting input to a specific limit, a value above or below a limit [10]. Boundary value analysis is one type of testing technique with the Blackbox Testing approach, which tests the upper and lower limits of a value inputted into the application. The main requirement of boundary value analysis is complementing the equivalence class testing technique, which only pays attention to the input value. At the same time, the boundary value analysis also pays attention to the output value. Boundary value analysis examines the input at the upper and lower limits of a valid value.

2.3. Comparison testing

Test each software version with the same data to ensure all versions produce the same output [11].

2.4. Sample testing

Involves several values selected from an equivalent class. Integrate values into test cases. Selected values may be selected in a specific order or at certain intervals.

2.5. Robustness testing

Input data is selected outside of defined specifications. The purpose of this test is to prove that there is no error if the input is invalid.

2.6. Behavior testing

The test results cannot be evaluated if you only test once but can be evaluated if the test is carried out several times, for example, testing the stack data structure.

2.7. Performance testing

Evaluate the program's ability to operate with correct in terms of reference needs, for example, data flow, memory usage size, execution speed. To find out the workload or program configuration conditions. Performance specifications are defined at the specification or design stage. It can be used to test the limits of the application program environment.

2.8. Requirement testing

Requirements specification associated with software (input / output / function / performance) is identified at the requirements specification and design stage. Requirement testing involves creating test cases for each requirement specification associated with the program. Each requirement specification can be traced to a test case using a traceability matrix.

2.9. Endurance testing

Endurance Testing involves a repeated number of test cases to evaluate whether the program conforms to the specification requirements. The test requirements specification is defined at the requirements or design specification stage.

2.10. Cause –effect relationship testing

Software test design techniques cover a wide variety of areas, including identifying input and output cases [12]. The steps taken include determining the input conditions (cause) and the output conditions (effect) and making examples of cases for testing and creating test tables.

3. E-Learning madrasah

E-learning Madrasah is a free application for Madrasah products that are intended to support the learning process in Madrasas from Raudatul Athfal (RA), Madrasah Ibtidayah (MI), Madrasah Tsanawiyah (MTs) and Madrasah Aliyah (MA), to make it more structured, attractive and interactive. Madrasah e-learning has six access roles, including Madrasah Operators (Administrators), Subject Teachers, Counseling Guidance Teachers, Classroom Teachers, Students, and Supervisors (Madrasah Principals and staff). The learning tools used by the teacher are the 2013 Curriculum with a Learning Implementation Plan (RPP) and an assessment rubric that refers to the syllabus. Learning resources use textbooks with material taken from Graduate Competency Standards following Content Competencies and Basic Competencies.

The function of e-learning for learning activities in the classroom is as a supplement, which is optional or optional, complementary, or substitute (substitute). It is said to function as a complementary if the electronic learning material is programmed to complement the learning material received by students in the classroom. As a complement, it means that electronic learning material is programmed to become reinforcement or remedial material for students in following conventional learning activities.

4. Research methodology

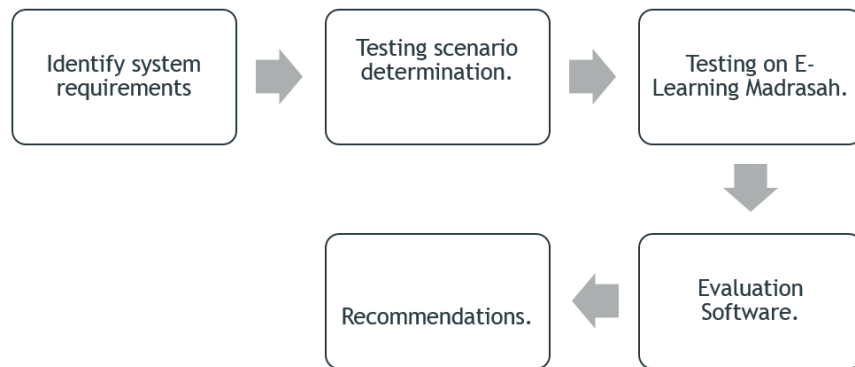The following research includes several stages, including the following Figure 1.

Figure 1. Research steps.

4.1. Identify system requirements

The system requirements phase is the process of understanding the proposed problem and ensuring that it supports the system requirements. Other things that are done include building a solid foundation for system development, system analysis activities, system modeling requirements, and development strategies.

4.2. Testing scenario determination

Testing the madrasah e-learning application based on the data that has been obtained from the application. The testing scenarios include login form testing, class setting testing, computer-based test testing, student learning evaluation testing. A scenario is a story or narrative that is easy to understand to make the application come alive. Stories that are easily accessible help the target of the story to understand the reasons why the problem is being solved by the system to be built. A scenario is not a technical specification description of a software product. A scenario is a narrative or story.

4.3. Testing on e-learning madrasah

Software Testing of e-learning Madrasah based on a scenario that has been made. Testing includes test methods applicable to all levels of software testing: unit, integration, functional, systems, and acceptance. This usually consists of software testing but can also dominate unit testing.

4.4. Evaluation software

Program evaluation is an effort to determine the extent to which the level of success of software which later results from this evaluation can be used by policymakers (Decision maker in making decisions whether the program is appropriate and still appropriate to continue or is still lacking, so it needs revision or even stopped.

4.5. Recommendations

Recommendations are resulting from the results obtained on software testing with the BlackBox Testing approach.

5. Results and discussion

The test scenario includes several things, including the following.

5.1. Login form scenarios

In the login form scenario, it is done by entering the username and password in the application program that has been run. Figure 2 is a display on the login form. The login form is used to find out which system users have obtained access rights from the application program. The test scenario on the login form knows the level of user access, both users by teachers, students, and administrators who have been appointed by the school.
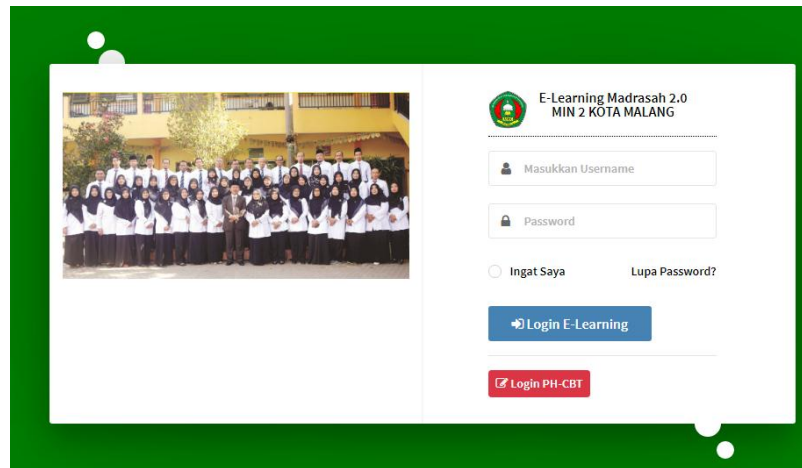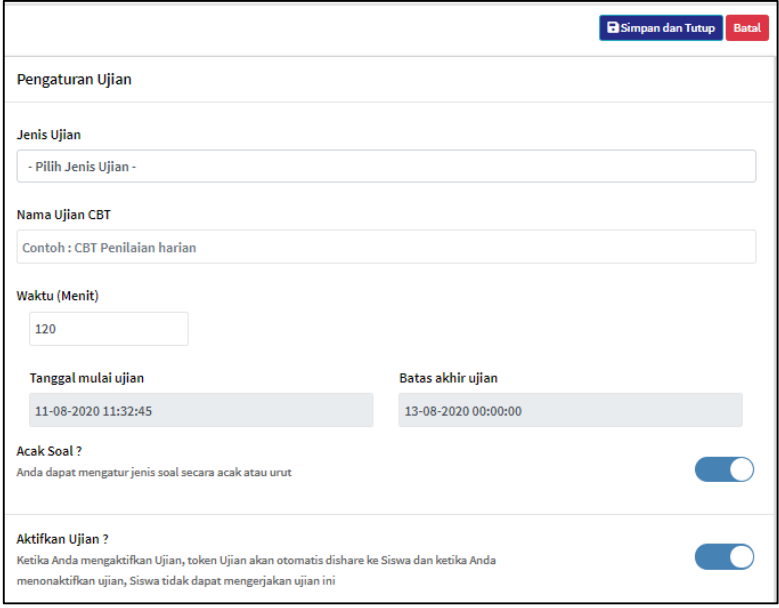

Figure 2. Login form.

5.2. Classroom scenario

Figure 3 shows the form to create a new class that is used for the class management process. From the results of the scenario, input parameters and agendas will be carried out at each meeting conducted based on the class that the teacher is teaching. Scenarios in the classroom setting test are needed to determine the level of validity results of the application function with the BlackBox testing approach.

Figure 3. Form create a new class.

5.3. Computer based test scenarios

The test on the Computer Based Test scenario is shown in Figure 4 From this scenario will measure the functionality of the online exam arrangements carried out by teacher users to students who have been connected in their respective classes. The parameters that were tested included the type of test, randomization of questions, time of the test, and the results obtained when students finished working on the questions that had been given by the teacher. In addition, there will also be a token generation test before the test starts.

Figure 4. CBT form.

5.4. Student learning evaluation scenarios

The results of the evaluation of student learning make it easier for teachers to measure the level of student achievement in understanding the lessons that have been given. From these results, several things related to the assessment will be obtained. The test scenario in the evaluation of student learning is shown in Figure 5.

Figure 5. Student learning evaluation form.

From the results of the scenario, the level of validity of the test was generated, as shown in Table 1. In the scenario that was tested, there were results with different percentage variations.

Table 1. Evaluation result

Scenario	*Valid Test Case (%)*
1	82
2	95
3	85
4	93
5	95

The graph shown in Figure 6 shows the evaluation of software testing with the BlackBox Testing approach in madrasah e-learning. These results follow the requirements of the system applied to a particular school. These results may change according to the level of system requirements that have been identified.
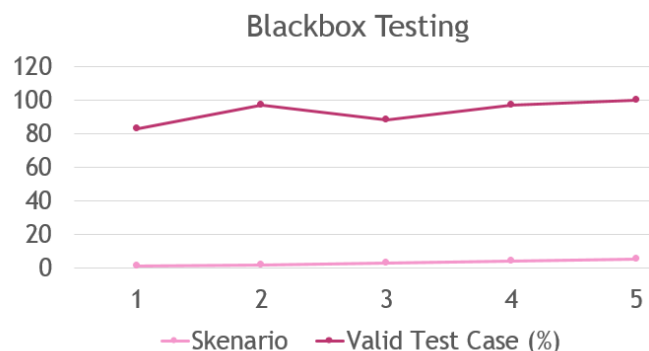

Figure 6. Blacbox testing form.

6. Conclusion

In the application testing scenario that has been tested, there are several things that need to be considered, including the results of the accuracy level of the BlackBox testing examiner that is in accordance with the existing results. The test results in Blackbox testing showed the level of functionality in e-learning Madrasah software shows the results reaching 90%. The recommendation results in the Madrasah e-learning test show are quite optimal results. Blackbox testing is perfect for testing Blackbox testing.

Acknowledgments

References

[1] Puspita Sari A and Setiawan A 2018 The Development of Internet-Based Economic Learning Media using Moodle Approach *Int. J. Act. Learn.* **3** 100–9

[2] Sutiah and Supriyono 2020 Improvement of e-learning based on hybrid learning methods at the university in the era of industrial revolution 4.0 *Int. J. Adv. Sci. Technol.* **29** 2137–42

[3] Sutiah S and Supriyono S 2020 Fuzzy topsis optimization on expert systems for core competency detection and Islamic religious education student learning achievement at Perguruan Tinggi Keagamaan Islam Negeri (PTKIN) *IOP Conf. Ser. Mater. Sci. Eng.* **732**

[4] Sutiah S and Supriyono S 2020 Software Testing on The Learning of Islamic Education Media Based on Information Communication Technology Using Blackbox Testing *IJISTECH (International J. Inf. Syst. Technol.* **3** 254–60

[5] Wang Y H 2020 Design-based research on integrating learning technology tools into higher education classes to achieve active learning *Comput. Educ.* **156** 103935

[6] Boukhris S, Andrews A, Alhaddad A and Dewri R 2017 A case study of black box fail-safe testing in web applications *J. Syst. Softw.* **131** 1339–51

[7] Lonetti F and Marchetti E 2018 *Emerging Software Testing Technologies* vol 108 (Elsevier Inc.)

[8] Siddiqui J H, Rauf A and Ghafoor M A 2018 *Advances in Software Model Checking* vol 108 (Elsevier Inc.)

[9] Irawan Y, Muzid S, Susanti N and Setiawan R 2019 System Testing using Black Box Testing Equivalence Partitioning (Case Study at Garbage Bank Management Information System on Karya Sentosa) 1–7

[10] Prayudha I P A, Hartati R S and Divayana Y 2019 on Learning Management System Applications **4** 25–8

[11] R. FARAH DINI QOYYIMAH, Erfan Rohadi, ST., M. Eng., Ph.D and Rizky Ardiansyah, S.Kom, MT 2020 Pengembangan Sistem Informasi Pemetaan Infrastruktur Sistem Informasi Di Kota Probolinggo *J. Inform. Polinema* **6** 65–70

[12] Babbar H 2017 Software Testing : Techniques and Test Cases *Int. J. Res. Comput. Appl. Robot.* **5** 44–53