

Application of Agile Development Methodology in the Design and Implementation of a Web-Based Online Examination System to Improve the Efficiency and Security of Learning Evaluation in Higher Education

Jajam Haerul Jaman¹, Alif Fadillah Ummar²

^{1,2}Department of Informatic, Universitas Singaperbangsa Karawang, Indonesia

Article Info

Article history:

Received 14/07/2026

Revised 16/09/2026

Accepted 16/09/2026

Keywords:

Online Examination System,
Agile Development,
Laravel,
Learning Evaluation,
User Acceptance Testing

ABSTRACT

Digital transformation in education demands more efficient, secure, and integrated evaluation systems. However, the implementation of online examinations at Universitas Singaperbangsa Karawang (UNSIKA) still relies on simple platforms such as Google Forms, which possess limitations in supervision, security, and data management. This study aimed to design and implement a web-based online examination system using the Laravel framework and a MySQL database, applying the Agile Development methodology to address these issues. The main contribution of this research is the provision of an adaptive and integrated examination system development model that specifically addresses institutional needs through features such as user management (administrators, lecturers, and students), question bank management, exam scheduling, timer and auto-save functionalities, as well as automated scoring and reporting. The Agile method was applied through iterative sprint cycles, starting from identifying user requirements for a product backlog, system design using UML and ERD, feature development, to testing and evaluation. Black-box testing results demonstrated that all system functions operated according to the functional requirements. Furthermore, user acceptance testing (UAT) involving 30 respondents (10 administrators, 10 lecturers, and 10 students) using a 5-point Likert scale yielded an average score of 4.5, indicating a very high level of user acceptance and usability. Thus, the developed system is proven to provide a more effective, efficient, and secure learning evaluation solution compared to the previously utilized platform.



This work is licensed under a [CC-BY-NC](https://creativecommons.org/licenses/by-nc/4.0/)

Corresponding Author:

Jajam Haerul Jaman,
Department of Informatic,
Universitas Singaperbangsa Karawang,
Jl. HS.Ronggo Waluyo, Telukjambe Timur, Karawang, Jawa Barat 41361.
Email: jajam.haeruljaman@staff.unsika.ac.id

1. INTRODUCTION

The development of information technology has driven significant transformations in various sectors, including learning evaluation systems in higher education. Examinations are one of the most commonly used forms of evaluation, serving as an objective tool to measure students' competency achievement [1]. Along with technological advancements, examinations that were previously conducted conventionally using paper-based methods have gradually shifted to online systems to improve time efficiency and resource utilization [2].

However, in practice, the implementation of examinations at Universitas Singaperbangsa Karawang (UNSIKA) still relies on platforms such as Google Forms. The use of this method is considered less optimal, as it still faces various challenges, such as the potential for cheating, limited supervision, and the lack of supporting features, such as integrated time management and adequate security systems. In addition, the process of managing examinations by lecturers and educational staff is also less efficient, particularly in terms of question arrangement, scoring, and the presentation of evaluation results that are not yet well-integrated [3].

Several previous studies have shown that the use of the Laravel framework in web-based application development can improve system efficiency and quality. A study by [4] indicated that Laravel can accelerate the development process by reducing the amount of code that needs to be written and facilitating integration with various third-party services. Applications built with Laravel also demonstrate good responsiveness and scalability. Similar findings were also stated by [5], which highlighted that Laravel provides various features that facilitate developers in building applications efficiently. A comparative performance study between PHP frameworks also revealed that Laravel offers a more modern structure and a richer ecosystem for large-scale web application development compared to CodeIgniter [6]. Furthermore, the application of user-centered design (UCD) in interface design has been proven to enhance the quality of user interaction with information systems, as demonstrated in the UI/UX design of a web-based batik sales application [7]. These good design principles are also aligned with the development of an employee information system using the waterfall method, where clear interface design is one of the success factors for system implementation [8]. For systems requiring real-time user identification, such as student attendance, the integration of Internet of Things (IoT) technology with RFID cards has also been implemented and shown effective results in improving the accuracy and efficiency of the process [9]. However, these studies focused more on the technical aspects of software development and did not specifically address the application of adaptive development methodologies to overcome specific problems in online examination systems, such as the need for strict supervision, data security, and integration of the evaluation process.

Based on the identified problems and research gap, this study offers a novelty by concretely integrating the Agile Development method to design and implement an online examination system that not only prioritizes technical aspects but also systematically solves problems of supervision, security, and efficiency of evaluation management. The main objective of this study is to develop a web-based online examination system that is user-friendly, easily accessible, and capable of improving the efficiency and accuracy of the learning evaluation process. Specifically, this system is designed to provide a platform for conducting online examinations, tests, and quizzes within UNSIKA, facilitate lecturers in creating, managing, and assessing examinations digitally, and assist students in completing exams with features such as time management (timer), auto-save, and security mechanisms to minimize cheating. The main contribution of this study is the provision of an integrated and adaptive online examination system model developed through an iterative Agile process that can address the specific needs of the institution and enhance academic integrity.

2. METHODS

This study employs the Agile Development methodology, an approach to software development carried out in incremental and iterative stages. Its primary goal is to quickly adapt to changing user requirements. Each iteration in Agile, known as a sprint, typically lasts between one and four weeks, depending on the complexity of the project. The implementation of Agile Development allows for faster decision-making processes, maintains the quality of the developed system, and effectively accommodates changes, resulting in a system that is more flexible, responsive, and aligned with user needs as well as the dynamics of technological advancement [10].

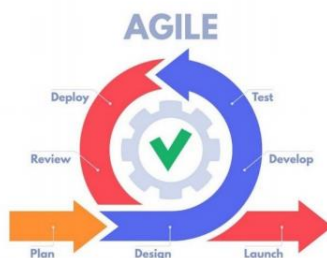
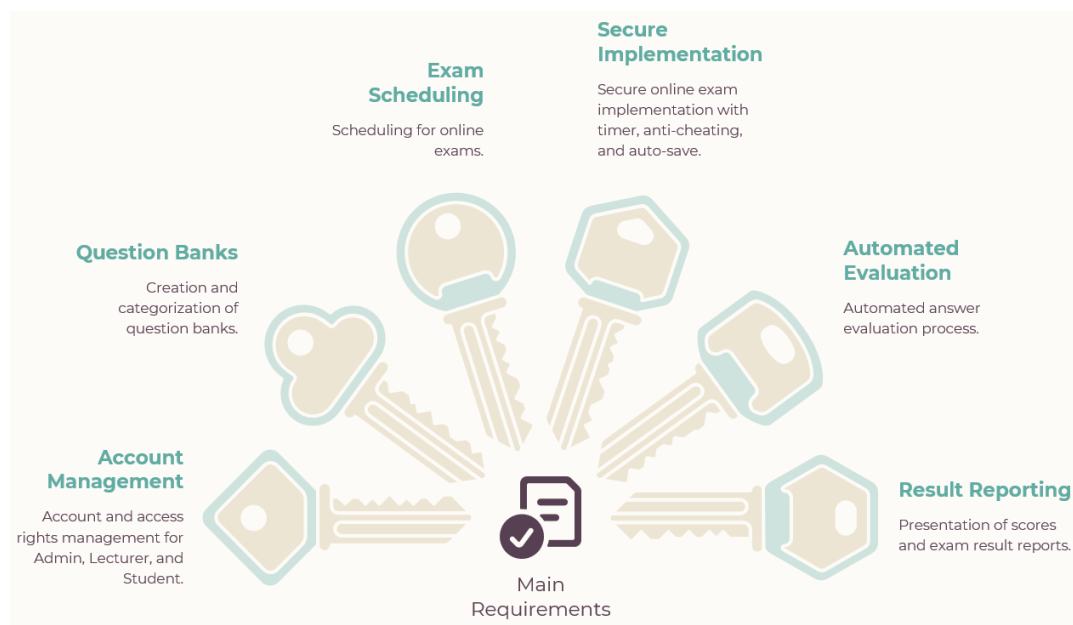


Figure 1 Stages of the Agile Development Model

The stages of Agile Development applied in this study are illustrated in Figure 1 and explained as follows. The Agile methodology was selected because it aligns with the iterative nature of educational system development, as demonstrated in similar research on e-learning platforms and competency models [11]. The Research and Development (R&D) approach by Borg and Gall was also considered, but Agile was preferred because of its flexibility in accommodating changing user requirements during the development process [12].

1) Plan (Planning)

At this stage, user requirements are comprehensively identified, involving Admins, Lecturers, and Students at Universitas Singaperbangsa Karawang, Indonesia. This process was conducted through interviews and observations to understand the existing problems with the conventional examination system.

**Figure 2 Main Requirements**

The main requirements gathered are presented in Figure 2.

- Account and access rights management (Admin, Lecturer, Student).
- Creation and categorization of question banks
- Exam scheduling.
- Secure online exam implementation (equipped with a timer, anti-cheating features, and autosave).
- Automated answer evaluation
- Presentation of scores and exam result reports
- All these requirements were then compiled into a product backlog that served as the basis for the sprint planning.

2) Design (System Design)

At this stage, various UML models were developed to represent the structure and workflow of the online examination system. The diagrams used include:

- A use case diagram was used to describe the relationship between actors (Admin, Lecturer, Student) and system functionalities such as managing questions, scheduling tests, taking tests, and viewing results.

- Activity Diagram to illustrate the main activity flows, such as test creation, the test-taking process by students, and the grading process.
- Sequence diagram to explain the sequence of interactions when users log in, manage questions and tests, take examinations, and view scores and reports.
- Entity Relationship Diagram (ERD) to design an integrated database structure.

3) Development

Application development was carried out iteratively through several sprints using PHP with the Laravel framework, React TypeScript for the front end, and MySQL as the database.

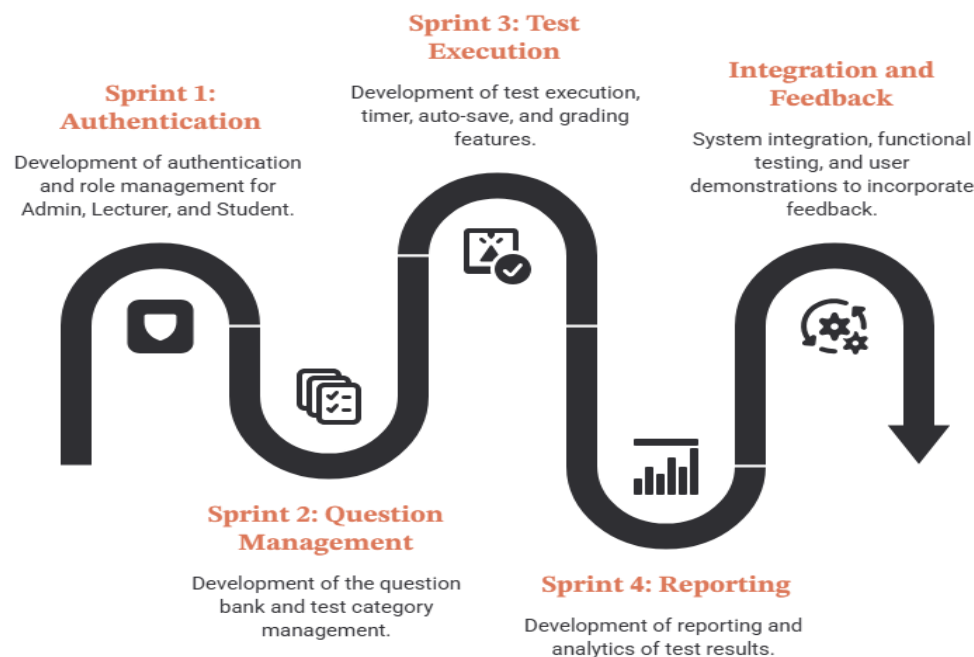


Figure 3 Agile Sprint Timeline

Each sprint focused on one or more core modules, as presented in Figure 3.

- Sprint 1: Authentication and role management (Admin, Lecturer, Student).
- Sprint 2: Question bank and test category management.
- Sprint 3: Test execution, timer, auto-save, and grading features were evaluated.
- Sprint 4: Reporting and analytics of the test results.
- At the end of each sprint, system integration and initial functional testing were conducted, followed by demonstrations to prospective users to obtain feedback. This feedback was then incorporated into the product backlog for improvement in subsequent iterations.

4) Testing

System functionality was tested using the Black Box Testing method to ensure that all features (login, question management, test execution, grading, and reporting) function according to requirements. In addition, usability testing was conducted using the System Usability Scale (SUS) method, as presented in Figure 4.

System Testing and Usability

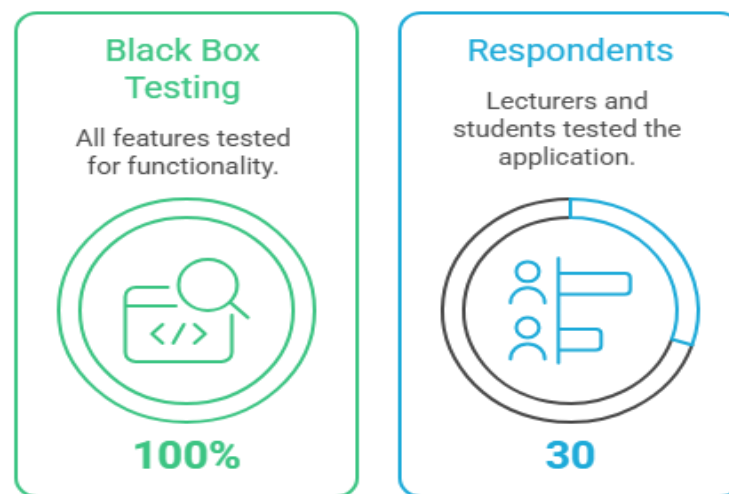


Figure 4 Black Box testing confirmed functionality, while SUS testing measure user perception

A total of 30 respondents, comprising lecturers and students, were asked to try the application and complete a SUS questionnaire using a 5-point Likert scale. The SUS score was then calculated according to standard procedures to determine the perceived usability level of the developed online examination system.

5) Deploy, Review, and Launch

After several main sprints were completed and the system was deemed to meet the functional and usability requirements, the application was deployed in a test server environment at Universitas Singaperbangsa Karawang. The system was then tested by actual users (Admins, Lecturers, and Students) to ensure its suitability with the examination process at UNSIKA. The results of testing and field evaluation were used for final refinements before the system was officially launched as an operational online examination system, and served as a basis for maintenance and further feature development in subsequent iterations.

3. RESULTS AND DISCUSSION

This section presents the results of each Agile stage, including interface implementation, functional testing results, and system usability evaluation.

3.1 System Design Results

The system design began with the creation of UML diagrams to model functional requirements. Figure 5, 6, and 7 shows the Use Case Diagram, Activity Diagram and Sequence Diagram illustrating the interaction between Admin, Lecturer, and Student with the system. The Admin has full access to manage users, schedules, and test categories. Lecturers can manage the question bank and view test results. Students can take online tests and view their scores and history.

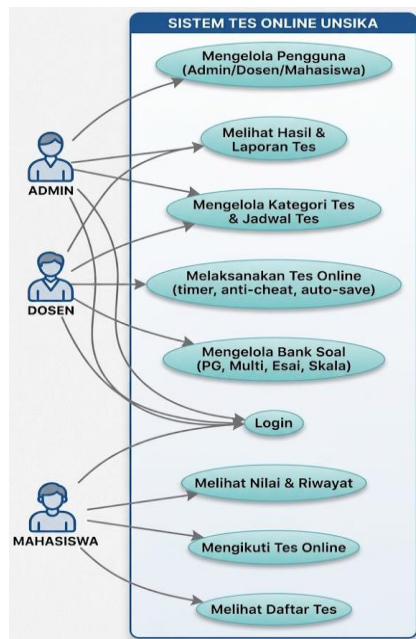


Figure 5 Use Case Diagram

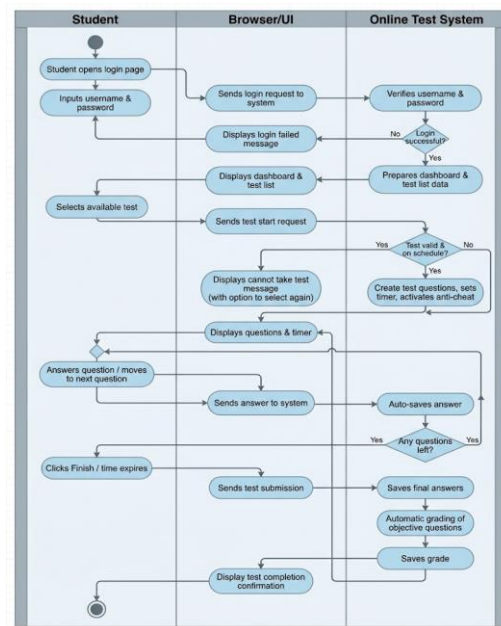


Figure 6 Activity Diagram

Furthermore, the Entity Relationship Diagram (ERD) in Figure 8 illustrates the database structure supporting the entire examination process. The main entities include users (storing admin, lecturer, and student data), roles, courses, classes, enrolments (many-to-many relationship between students and classes), tests, questions, question options, test sessions, and answers. The relationships between these entities are designed to support the system flow from user management, test creation, test administration by students, to the final assessment process.

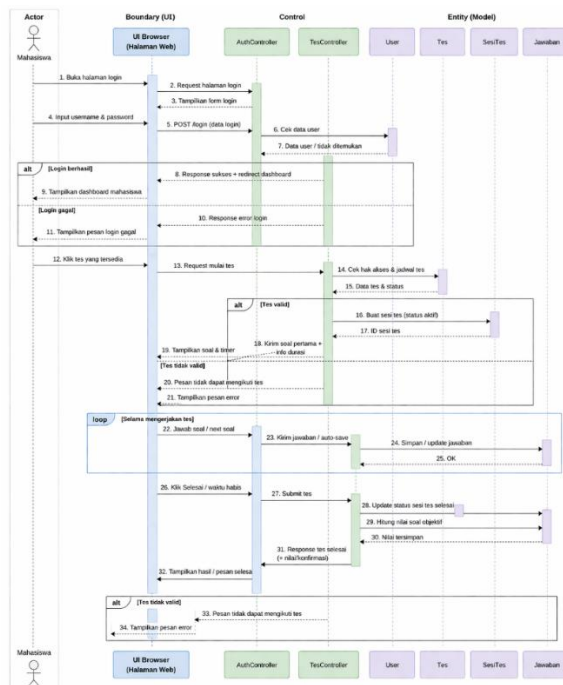


Figure 7 Sequence Diagram

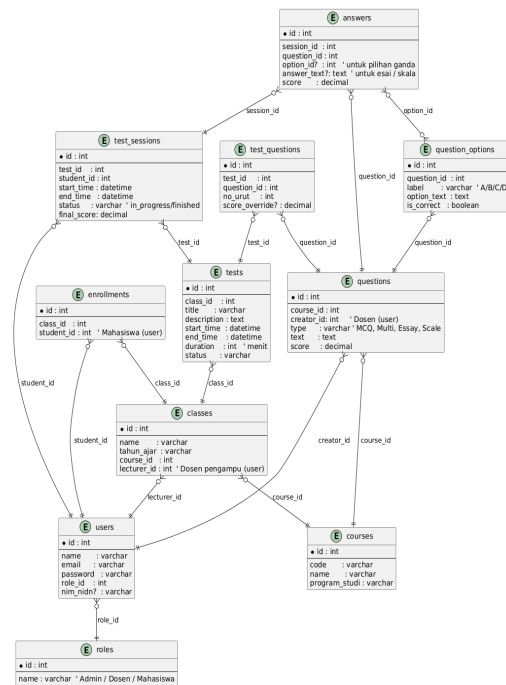


Figure 8 Entity Relationship Diagram

3.2 System Design Results

The system interface was designed to facilitate user interaction. The following are the main pages that have been successfully implemented:

1. Dashboard

The dashboard page is the main display after a user successfully logs in. On this page, admins can view summary information, such as the total number of participants, online participants, and active tests, as well as recent activity and daily statistics. The dashboard interface is designed to be simple, responsive, and user-friendly to facilitate monitoring of system conditions, as shown in Figure 9.

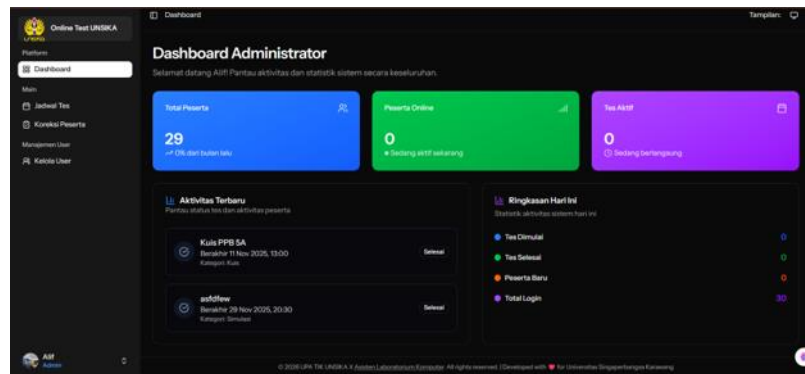


Figure 9 Administrator Dashboard

2. Test Schedule

The test schedule page is used to manage and display a list of available exam schedules. Admins can view information such as the test name, date, test status, access mode, and number of participants. Search, filter, and schedule addition features are also available to facilitate more efficient data management, as presented in Figure 10.

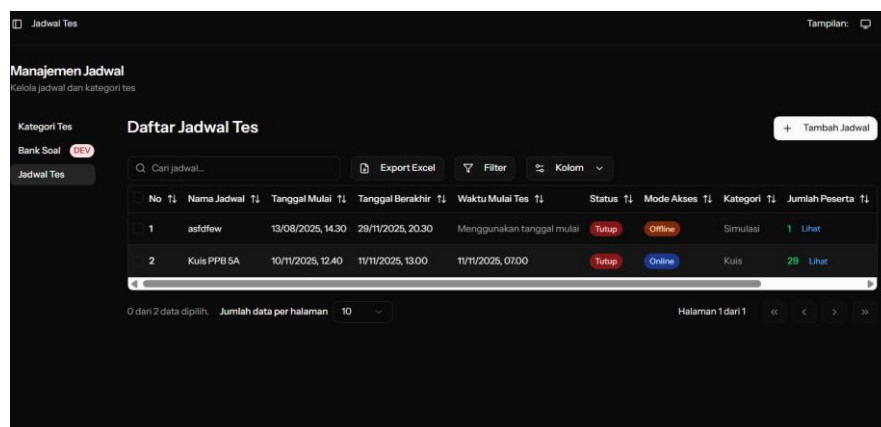


Figure 10 Test Schedule Management Page

3. Question Bank

The question bank page is used to manage the collection of questions to be used in exams. Admins can view a list of questions along with information such as difficulty level, score, and usage status. Search, filter, question addition, and data import features are available to support efficient question management, as shown in Figure 11.

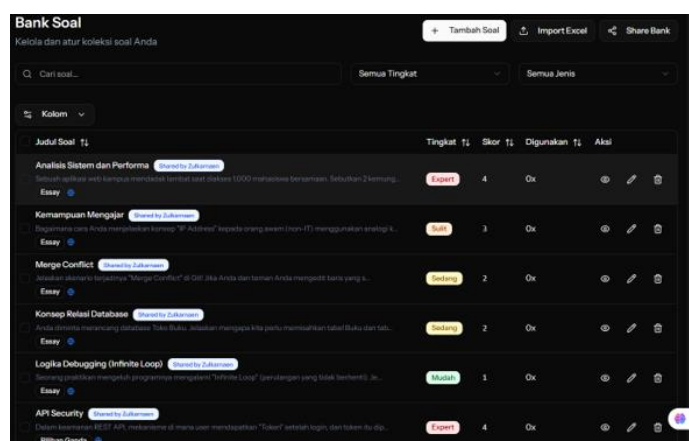


Figure 11 Question Bank Page

4. Test Categories

The test categories page is used to manage exam types or categories within the system. Admins can view a list of categories and the number of associated test schedules, as shown in Figure 12.

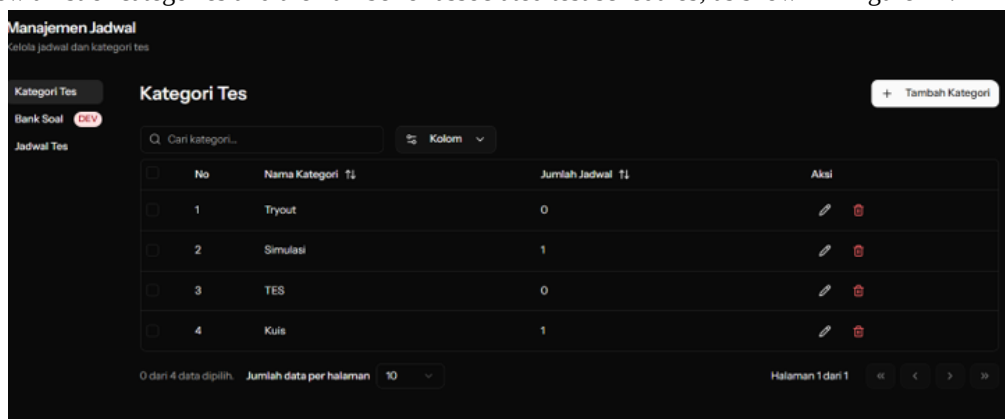


Figure 12 Test Categories Page

5. Participant Correction

The participant correction page is used to manage and monitor the assessment process for participant exam results. Admins or lecturers can view the test schedule, including the number of participants, correction status, and assessment progress, as presented in Figure 13.

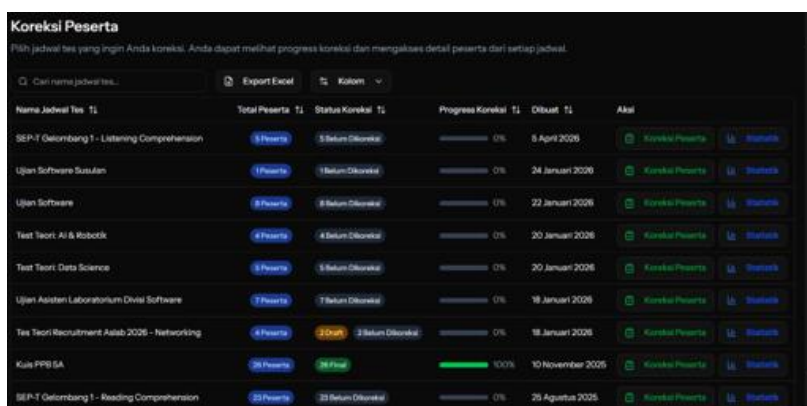


Figure 13 Participant Correction Page

6. Manage Users

The user management page is used to manage user data in the system, such as admins, lecturers, and students. Admins can view user information such as username, full name, email address, and role, as presented in Figure 14.

Username	Nama Lengkap	Email	Role	No. HP	Dibuat	Aksi
2210631250071	Vendamme Muhammad Rahardjo	2210631250071@student.unsika.ac.id	Peserta	-	6/4/2026	[Edit] [Delete]
ArQ	Arqi Putra Cassanova	2210631140059@student.unsika.ac.id	Peserta	-	6/4/2026	[Edit] [Delete]
2210631250102	Vernanda Ayu Prastika	2210631250102@student.unsika.ac.id	Peserta	-	6/4/2026	[Edit] [Delete]
Fatimahzahrati	SITI FATIMAH AZZAHRA	2210631020173@student.unsika.ac.id	Peserta	-	6/4/2026	[Edit] [Delete]
2210631050018	Laila Cantika	2210631050018@student.unsika.ac.id	Peserta	-	6/4/2026	[Edit] [Delete]
2210631020041	Nadia Sadia	2210631020041@student.unsika.ac.id	Peserta	-	6/4/2026	[Edit] [Delete]
2210631160057	Muhammad Rizal Hakim	2210631160057@student.unsika.ac.id	Peserta	-	6/4/2026	[Edit] [Delete]
2210631170096	Putri Fisichella Chasanovve	2210631170096@student.unsika.ac.id	Peserta	-	6/4/2026	[Edit] [Delete]
2210631260039	Murnitaz Najih Fauzan	2210631260039@student.unsika.ac.id	Peserta	-	6/4/2026	[Edit] [Delete]
2210631020033	Khensia Agila Haryanto	2210631020033@student.unsika.ac.id	Peserta	-	6/4/2026	[Edit] [Delete]

Figure 14 Manage Users Page

3.3 Functional Testing Results (Black Box Testing)

Black box testing was performed to ensure that all system functionalities operated properly without considering the internal structure of the program. Table 1 summarizes the testing results for several key features.

Table 1 Black Box Testing

No	Feature	Scenario Testing	Input	Output	Result
1	Login	Login with valid data	username and password are correct	Successfully logged in to the dashboard	Valid
2	Login	Login with invalid data	Incorrect username/password	An error message appears	valid
3	Dashboard	Display statistical data	-	Participant data, active test appears	valid
4	Test schedule	Add test schedule	Input schedule data	saved data	Valid
5	Test schedule	edit test schedule	Change schedule data	Data updated successfully	Valid
6	Test schedule	Delete test schedule	Click delete	Data deleted from the system	Valid
7	Test schedule	Schedule Search	Enter keywords	Data as shown	valid
8	Question bank	Add questions	Input new questions	Saved questions	Valid
9	Question bank	Edit questions	Change the question	Question data changed	Valid
10	Question bank	Delete the question	Click delete	Question deleted	Valid
11	Question bank	Import questions	Excel File	Question data enters the system	Valid
12	Test categories	Add Category	Input Category	Saved Categories	valid
13	Test categories	Edit Category	Data changed	Data changed	Valid
14	Test categories	Delete Category	click delete	Data deleted	Valid
15	Test	Start the test	Click start	Regarding appearance	Valid
16	Test	Answering questions	Select an answer	Saved answers (auto-save)	Valid
17	Test	Submit test	Click finish	Test completed	valid
18	Correct	Open the correction page	Click correction	Open page	Valid
19	Correct	Save Value	Input value	Status changed	valid
20	User	Add user	Input user data	Saved users	valid
21	User	Edit user	Change data	Data changed	valid
22	User	Delete user	Click delete	User deleted	valid

3.4 Usability Evaluation Results (User Acceptance Testing)

This testing involved 30 respondents (10 admin, 10 lecturers, and 10 students) to assess the developed system using a Likert scale (1 = strongly disagree, 5 = strongly agree). Table 2 presents a summary of the UAT results, presented in Table 2.

Table 2 User Acceptance Testing Results

No	Feature	Statement	Skor (1-5)
1	Login	Easy to use login system	5
2	Dashboard	The information on the dashboard was easy to understand.	4
3	Test schedule	Managing test schedules is easy	5
4	Question bank	Easy to use question bank feature	4
5	Test categories	Test category management is clear and easy	4
6	Test	The exam process went smoothly	5
7	Test	Auto-save filter helps users	5
8	correct	The correction process is easy to do	4
9	User	User management is easy to understand	4
10	System	The system was running well without any errors.	5

Based on the test results, an average score of 4.5 was obtained, indicating that the developed online examination system ran well and was acceptable to users.

3.5 Discussion

The results of this study indicate that the concrete application of the Agile Development method through planned sprint cycles successfully addressed the problems of the conventional examination system at UNSIKA. Unlike simple platforms such as Google Forms, the developed system provides integrated features that directly address the needs for supervision, security, and efficiency in evaluation management.

The advantages of this system include its ability to:

1. Improved Security and Academic Integrity: The timer, autosave, and anti-cheating mechanisms designed during the test development sprint significantly reduced cheating opportunities compared to Google Forms-based exams.
2. Increase Examination Management Efficiency: Automated processes for question banking, scheduling, and grading reduce the administrative burden on lecturers and staff, allowing them to focus more on teaching activities.
3. Provide a Better User Experience: The average UAT score of 4.5 proves that the system is easy to use and well accepted by admins, lecturers, and students. This demonstrates that the agile approach, which involved user feedback in each sprint, successfully created a user-centric system.

These results align with the findings of [4], [5], which state that Laravel and Agile methodologies can improve the quality and efficiency of software development. However, this study contributes further by practically demonstrating how the integration of both can solve specific problems in the domain of online examination systems.

4. CONCLUSION

This study successfully designed and implemented a web-based online examination system using the Agile Development approach. The concrete application of this method through iterative sprint cycles can address the challenges in the learning evaluation system in higher education. Based on the results of the tests conducted, it can be concluded that:

1. Research Objectives Achieved: The developed online examination system successfully met the functional requirements identified during the planning stage. This is evidenced by the Black Box Testing results, where all 16 functional test scenarios were declared valid.
2. High Level of User Acceptance: The User Acceptance Testing (UAT) results involving 30 respondents (10 admins, 10 lecturers, and 10 students) yielded an average score of 4.5 out of 5. This

score indicates that the system is rated very good, easy to use, and accepted by users, effectively solving the problems of efficiency and ease of exam management.

3. **Research Contribution:** The main contribution of this research is the provision of an integrated and adaptive online examination system development model. This model not only focuses on technical aspects (Laravel, MySQL) but also methodologically demonstrates how Agile Development can be applied to create solutions that specifically address institutional problems, such as improving security, supervision, and integration of evaluation data. This system offers a superior alternative to generic platforms such as Google Forms.

Research Limitations:

This study has several limitations. First, the scope of implementation and testing was limited to the Universitas Singaperbangsa Karawang environment. Second, the number of respondents for UAT (30 people) may not fully represent the entire user population within the university. Third, the system security evaluation is still limited to functional testing and does not include penetration testing or a more in-depth vulnerability analysis.

Recommendations for Future Research:

For future development, the following recommendations are suggested:

1. The scope of implementation and testing should be expanded to more educational institutions to test the scalability and adaptability of the system.
2. Conduct more comprehensive security testing, including penetration testing, to identify and address potential vulnerabilities.
3. Integrate the system with the existing Academic Information System (SIKAD) at the university to create a more centralized data ecosystem.
4. Develop more advanced analytical features to provide deeper insights into student performance and the quality of the exam questions.

References

- [1] A. Siswopranoto, G. Saputri, F. I. Komputer, and U. Pamulang, "Penerapan Aplikasi E-Perkara di Pengadilan Agama Jakarta Barat dengan Model Agile Development," vol. 18, no. 4, pp. 305–314, 2025.
- [2] A. Kurniawan, M. Ratnasari, S. Ilafiah, and S. Dwi Nugroho, "Perancangan Aplikasi CBT Exam Browser Dengan Metode Waterfall," *JRIIN J. Ris. Inform. dan Inov.*, vol. 2, no. 2 SE-, pp. 287–291, 2024, [Online]. Available: <https://jurnalmahasiswa.com/index.php/jriin/article/view/1399>
- [3] A. Soderi and K. Diantoro, "Perancangan Sistem Aplikasi Ujian Online Berbasis Web pada LPIA Jatimurni Menggunakan Laravel," vol. 4, no. 3, pp. 5186–5193, 2025.
- [4] F. Sinlae, E. Irwanda, Z. Maulana, and V. E. Syahputra, "Penggunaan Framework Laravel dalam Membangun Aplikasi Website Berbasis PHP," *J. Siber Multi Disiplin*, vol. 2, no. 2, pp. 119–132, 2024.
- [5] R. P. T. K. M. M. and J. Dominique, "An Analysis Of The Impact On Modern Web Application Development : Basis For Php Framework Efficiency Model," no. February, 2024.
- [6] J. Informasi, F. Praseyto, E. Putra, K. Mufidah, S. Anggreni, and A. Sugi, "Comparative Literature Study of CodeIgniter and Laravel Framework Performance in Website Creation," vol. 7, no. 1, pp. 4–5, 2025, doi: 10.60083/jidt.vi0.617.
- [7] W. Silviana and H. Kurniawan, "Perancangan Desain UI / UX Aplikasi Penjualan Batik Berbasis Website Pada Toko Batik Jambi Ariny Menggunakan Metode User Centered Design (UCD)," vol. 4, no. 2, pp. 687–698, 2025.
- [8] S. Anggrian and B. Y. Geni, "Perancangan Dan Pengembangan Sistem Informasi Kepegawaian Menggunakan Metode Waterfall (Studi Kasus: Pt. Dola Usaha Indonesia)," *J. Mhs. Tek. Inform.*, vol. 8, no. 1, pp. 1029–1035, 2024.
- [9] D. R. Hakim, P. D. Arnesia, and A. M. Yusup, "Perancangan Sistem Informasi Absensi Siswa Menggunakan Kartu Rfid Berbasis Internet Of Things (Iot) Di Smk Negeri 1 Cileungsi," *Semin. Nas. Teknol. Inf. dan Komun. (SeNTIK STI&K) STMIK Jakarta STI&K*, vol. 12, p. 2024, 2024.
- [10] R. Djatalov, M. Irham, A. D. Putra, A. Eliatiawan, and R. F. Syah, "Agile Methodology in Web and Mobile Based App of Tracer Study and Career Center," vol. 4, no. 1, pp. 453–462, 2024.
- [11] S. Gustiani and P. N. Sriwijaya, "Research And Development (R&D) Method As A Model Design In Educational Research And Its Alternatives," *HOLISTICS J.*, vol. 11, no. 2, 2019.
- [12] Torang Siregar, "Stages of Research and Development Model Research and Development (R&D)," *DIROSAT J.*

Educ. Soc. Sci. Humanit., vol. 1, no. 4, pp. 142–158, Oct. 2023, doi: 10.58355/dirosat.v1i4.48.