Journal of Informatics, Computer Science, Data Science And Arificial Intelligence
Volume 3 No. 2 December 2025, ISSN : 3031-9145

IMPLEMENTATION OF FISHER-YATES ALGORITHM IN MOBILE-BASED ONLINE EXAM APPLICATION AT SMP NEGERI 1 KARIMUN

Nur Alya Afifah[1], Mhd Zulfansyuri Siambaton[2], Rachmat Aulia[3]
[1,2] Universitas Islam Sumatera Utara, [1] nralyaafifah@gmail.com, [2] zulfansyuri@ft.uisu.ac.id
[3] Universitas Harapan Medan, jackm4t@gmail.com

ARTICLE INFO

ARTICLE HISTORY:
Received : 10 November 2025
Revised : 15 November 2025
Accepted : 19 November 2025

Keywords: *Algorithm, Examination, Fisher-Yates, Mobile, Online, SMP Negeri 1 Karimun.*

ABSTRACT

The development of information technology has driven significant changes in educational evaluation systems, one of which is through the implementation of mobile-based examinations. However, one of the main challenges in conducting online examinations is maintaining question randomization so that distribution is fair and reduces the potential for cheating. The Fisher-Yates algorithm is an algorithm used to shuffle the order of elements in a list or array in a perfectly random (uniform random) manner, where every possible permutation of those elements has an equal probability of occurring.

The implementation results demonstrate that the Fisher-Yates algorithm is capable of producing evenly distributed and non-repetitive question randomization, thereby enhancing the fairness and reliability of the examination system. Thus, the application of this algorithm can support the effectiveness of the learning evaluation process and provide a positive contribution to the utilization of educational technology at the junior high school level.

INTRODUCTION

Learning is a complex process that involves multiple components, including planning, implementation, and evaluation, all of which aim to achieve optimal educational outcomes. Evaluation plays a crucial role in determining the effectiveness of instructional methods and the extent of students' understanding. Through proper and systematic evaluation, educators are able to identify strengths and weaknesses within

the learning process, enabling continuous improvement and innovation to enhance the overall quality of education (Suryani & Putra, 2021); (Yuliani, 2020). In general, evaluation is conducted through assessments such as daily tests, mid-term examinations, and final examinations.

The rapid development of digital technology has significantly influenced the education sector by providing easier access to information and learning tools. Digital learning platforms and mobile-based educational applications have become increasingly common in supporting teaching and assessment processes (Hidayah & Saputra, 2024); (Maulana & Rachmat, 2022). As part of this technological integration, the implementation of Computer-Based Tests (CBT) using mobile devices has gradually replaced traditional paper-based examinations with more practical and efficient digital alternatives.

At SMP Negeri 1 Karimun, online examinations have been conducted using the Google Form platform. This platform offers several advantages, such as enabling teachers to create questions easily, distribute exam links, and monitor students' results in real time. Students can take exams conveniently using their personal mobile devices. However, despite these benefits, several issues related to academic integrity and examination security have been identified. One major issue is the possibility of students submitting their responses multiple times due to Google Form's default settings, which allow resubmission if not configured properly. In several cases, students attempt multiple submissions to determine the correct answer combinations. Additionally, Google Form may display the correct answers after submission if the settings are not adjusted, enabling students to view the answer key and share it with others.

These issues compromise the objectivity of the assessment and increase the potential for cheating. Teachers also face difficulties ensuring that students complete examinations independently without exchanging answers. Therefore, a more secure examination system is required to minimize opportunities for cheating and enhance the integrity of the assessment process.

One technical solution is the implementation of the Fisher-Yates Shuffle algorithm, a method capable of producing a perfect randomization of question sequences without repetitive patterns. By applying the Fisher-Yates algorithm in a mobile-based examination application, each student receives a uniquely randomized set of questions. This makes it more difficult for students to share answers, as the question order differs from one device to another. The development and structuring of such an application can be effectively supported through UMLbased modeling techniques, which offer clear frameworks for system design, such as use case diagrams, sequence diagrams,

and class diagrams (Firmansyah & Rachman, 2022); (Dewi & Maulana, 2021); (Setiawan & Nurhayati, 2022).

In this study, a mobile-based examination application was designed and developed for junior high school students using the Dart Flutter programming language, incorporating the Fisher-Yates Shuffle algorithm as the method for question randomization. The system was developed specifically for Class 8.1 students at SMP Negeri 1 Karimun, consisting of 33 students and covering 11 subjects, each comprising 25 examination questions. This application is expected to improve examination integrity and security while providing teachers with an efficient tool to manage examinations and obtain student scores automatically.

RESEARCH METHODS

This research was conducted through several systematic stages to develop a mobile based online exam application with the implementation of the Fisher-Yates Shuffling algorithm. The research methods used include system analysis, UML modeling, algorithm implementation case studies, system design, database design, and user interface design.

System Analysis

The initial stage of the research was conducted through system analysis to understand user needs, application workflow, and data structure required in developing the mobile-based online exam system. The analysis began with identifying problems in exam implementation that previously used Google Form, such as:

1. Students can submit answers more than once.
2. Google Form displays answer keys after submission if settings are not properly configured.
3. The same question order for all students when randomization settings are not used.

This identification was obtained through literature studies, observations, and discussions with SMP Negeri 1 Karimun school officials. The results of this analysis became the basis for designing a new system that integrates the Fisher-Yates Shuffling algorithm.

System Modeling (UML)

System modeling was carried out using Unified Modeling Language (UML) as a tool to visually and structurally describe the system design. The UML diagrams used include:

1. Use Case Diagram to describe interactions between users and the system.

2. Activity Diagram to illustrate the flow of activities within the system.
3. Class Diagram to map the data structure and relationships between classes.

This modeling serves as a guide in implementing a secure and structured Android-based online exam system.

Problem Analysis

Problem analysis aims to understand the constraints that emerged in the previous online exam system, namely:

1. Opportunities for cheating due to identical question order.
2. Students can access answers directly after submission.
3. Lack of consistent and evenly distributed question randomization.

These problems became the background for selecting the Fisher-Yates algorithm as a solution to randomize question order fairly and efficiently.

Fisher-Yates Algorithm Implementation Case Study

Case Description

A teacher creates a question bank consisting of 10 multiple-choice questions for the Indonesian Language subject. The questions are stored in a database table with fixed IDs (1-10). In a traditional system, all students would receive the same order, which could potentially lead to cheating.

Tabel 1 Soal ujian dalam *Database*

Question ID	Short Question	Correct Answer
1	Question 1	B
2	Question 2	A
3	Question 3	C
4	Question 4	D
5	Question 5	B
6	Question 6	C
7	Question 7	A
8	Question 8	D
9	Question 9	A
10	Question 10	B

To address this issue, the developed system implements the Fisher-Yates Shuffling algorithm so that each student receives a unique question order.

Implementation Steps

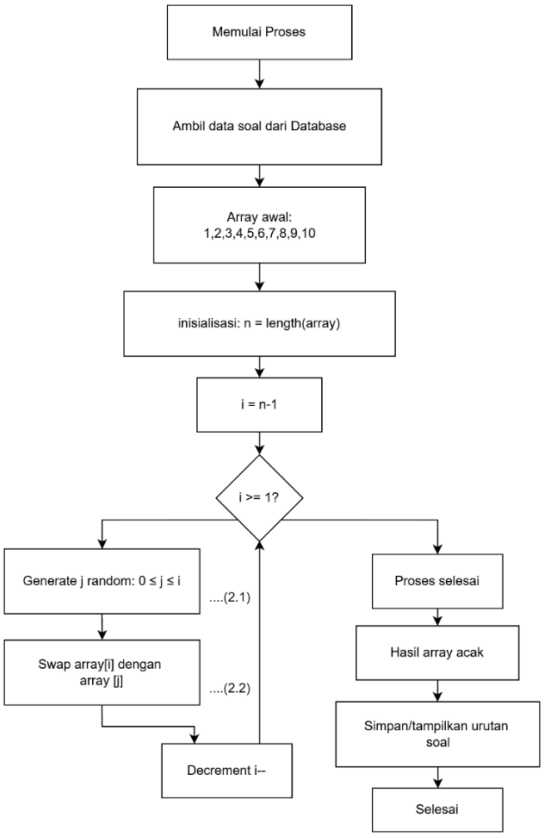

Figure 1. Case Resolution Process Flow

The following are the detailed stages of applying the Fisher-Yates Shuffling algorithm to the case above

Step 1: Retrieving Question Data

The system executes a query such as:

```
SELECT * FROM soal_ujian WHERE id_ujian = 'UJ001';
```

Result in the form of question ID array:

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

Step 2: Randomization Initialization

The system calls the shuffle() function with the Fisher-Yates algorithm.

Step 3: Iteration and Swapping

Example:

- Iteration 1: i=9, j=3 → swap

- Iteration 2: i=8, j=1 → swap

- and so on until i=1

Step 4: Final Result

Example of randomized array:

[8, 9, 3, 6, 5, 10, 1, 7, 2, 4]

Step 5: Randomization Testing

Testing was conducted on 5 students and the results showed no identical question orders.

Case Study Conclusion

The implementation of the Fisher-Yates algorithm proved to:

- Generate unique question orders for each student.
- Prevent cheating among participants.
- Maintain exam integrity without affecting question difficulty level.
- Be computationally efficient with O(n) complexity.

System Design

System design includes the following elements:

Use Case Diagram

The diagram in Figure 2 illustrates the interaction between users and the system. The main actors in the mobile-based online examination system are the Admin, Teacher, and Student. The main use cases in the system include: login, manage questions, take exam, and view exam results.

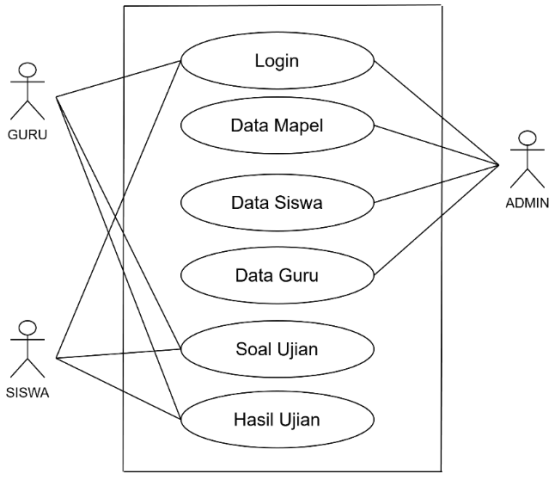

Figure 2. Application Use Case Diagram

Activity Diagram

Activity diagrams are created for each actor:

- Admin
- Teacher
- Student

These diagrams illustrate the workflow from login to exam result storage.

Class Diagram

Class diagrams explain the structure of classes such as:

- User
- Student
- Teacher
- Exam
- Question
- Exam Result

This diagram serves as the basis for composing database tables on Supabase.

Database Design

The database uses PostgreSQL through the Supabase platform which supports:

- Real-time synchronization
- Row-level security
- Auto-generated REST API
- Flexible integration with Flutter

Database tables include:

- Login
- Student
- Teacher
- Exam
- Exam Question
- Exam Result
- Subject

Each table is designed with appropriate data types to ensure optimal data storage and processing.

User Interface (UI) Design

The interface is designed to be user-friendly for students, teachers, and admins. Designs include:

- Login Page
- Admin Dashboard
- Data Management Page
- Teacher Main Page
- Question Management Page
- Question Creation Page
- Exam Schedule Selection Page
- Exam Execution Page (including question randomization)
- Exam Result Analysis Page
- Profile Page

RESULTS AND DISCUSSION

The results of this research consist of a mobile-based online exam application that implements the Fisher-Yates Shuffling algorithm as a question randomization method. This application has three types of users: Admin, Teacher, and Student, where each role has different access rights and features according to their functions.

The following is a description of the application results along with descriptions of each displayed output.

Login Page

The login page is the first gateway for users to access the system. This page functions to perform authentication based on *username* and *password* stored in the database.

After successfully logging in, the system will detect the user's role:

- Teacher → directed to Teacher Main Page
- Student → directed to Student Main Page
- Admin → directed to Admin Dashboard

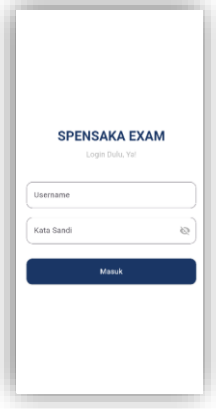

Figure 2 Description

This display shows an authentication form consisting of username and password fields, as well as a login button. The page design is made simple and focused to ensure users can log in quickly and securely.

Admin Dashboard Page

The admin dashboard page is the main control center. On this page, the admin can manage all core data, namely:

1. Manage Student Data
2. Manage Teacher Data
3. Manage Subject Data

The dashboard image displays three main menus in the form of large buttons/icons so that admins can easily navigate administrative tasks. Each button leads to the relevant management page.

Figure 3 Admin Dashboard Page

Data Management Page (Admin)

This page allows admins to perform CRUD (Create, Read, Update, Delete) processes on data:

- Teachers
- Students
- Subjects

Admins can add, modify, and delete data quickly.

The image shows a complete list of teachers with "Edit" and "Delete" buttons. There is also an "Add Data" button at the top to add new teachers. The layout is arranged in

table form for easy reading.

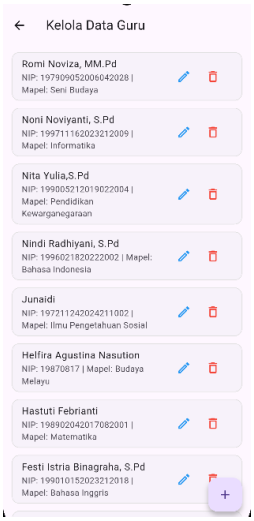

Figure 4 Teacher Data Management Page

Data Input Form Page (Admin)

The form page is used to add new data to the system, in the form of:

- Teacher Accounts
- Student Accounts
- Subjects

Admins fill in all columns according to the required attributes.

The form display contains input fields such as name, username, password, and other attributes according to data type. The form is designed to be simple and structured so admins can avoid input errors.

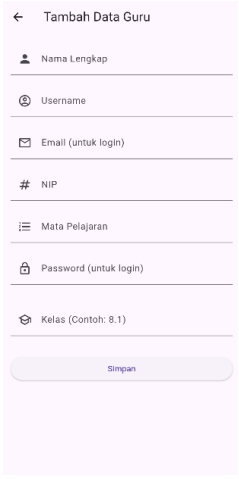

Figure 5 Teacher Data Input Form Page

Teacher Main Page

The teacher main page serves as a navigation point for:

- Creating questions
- Managing questions
- Viewing exam results
- Accessing profile

The image displays four main menus in the form of large cards/icons representing important functions for teachers. The design is kept clean for easy operation.

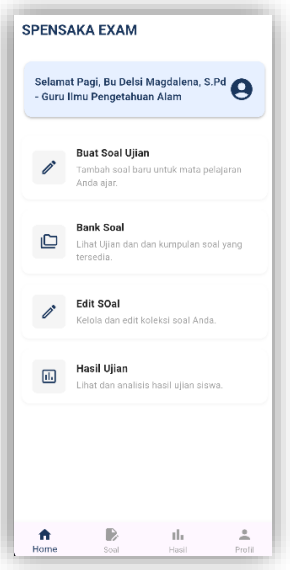

Figure 6 Teacher Main Page

Student Main Page

The student main page displays:

- List of exams
- Grade history
- Student profile

The display consists of a clear and organized main menu. Students can directly see active exams and access the list of grades obtained.

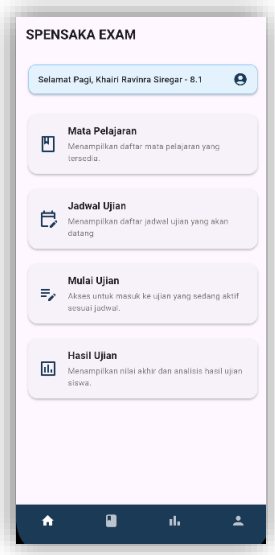

Figure 7 Student Main Page

Question Management Page (Teacher)

On this page, teachers can:

- Display question list
- Edit questions
- Delete questions

The image displays a list of questions in list form, complete with edit and delete buttons on each row. This facilitates teachers in question management.

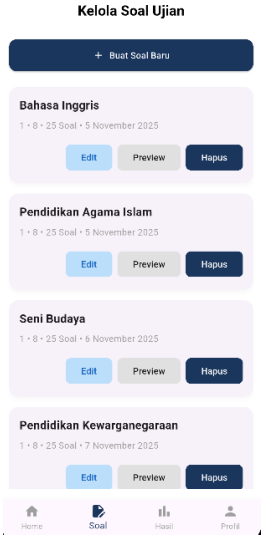

Figure 8 Teacher Question Management Page

Question Creation Form

Teachers can create new questions by filling in:

- Questions
- Answer options (A, B, C, D)
- Supporting images
- Answer key

The image shows the question creation form display. Text fields are available for questions and answer choices as well as an image upload button. After saving, questions enter the database.

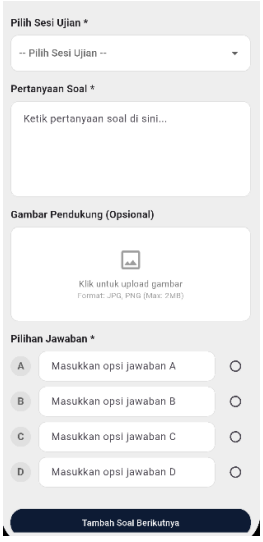

Figure 9 Question Creation Form Page

Exam Selection Page (Student)

Students can select available exams and check their schedules. The system provides time verification.

If the time has not arrived → "Take Exam" button is inactive.

The image displays a list of exams with subject names, exam dates, and action buttons. This display helps students understand the schedule clearly.

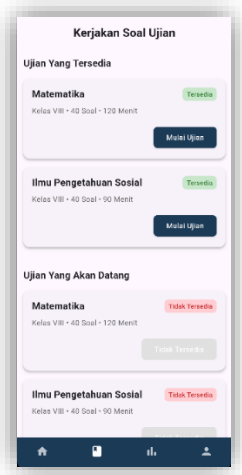

Figure 10 Student Exam Selection Page

Exam Execution Page (Student)

This page is the core of the exam system. At this stage:

- Questions are displayed in random order using the Fisher-Yates algorithm.
- Each student receives a different order.
- Questions can be in the form of text and images.

After completion, students press the Finish button, then grades are automatically processed.

The display contains questions at the top, answer choices, question navigation buttons, and exam time indicator. Question orders vary based on the shuffle algorithm.

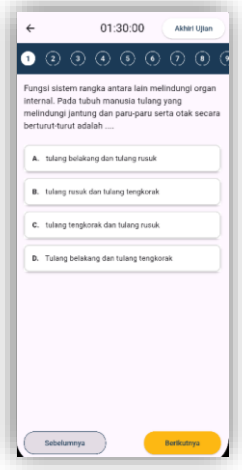

Figure 11 Student Exam Execution Page

Exam Result Analysis Page

After students complete the exam, the system displays results in the form of:

- Final grade
- Number of correct answers
- Number of incorrect answers

Grading criteria:

- Correct answer: 4 points
- Incorrect or blank answer: 0 points

The image displays a grade summary in card format. Students can see their performance directly without waiting for the teacher.

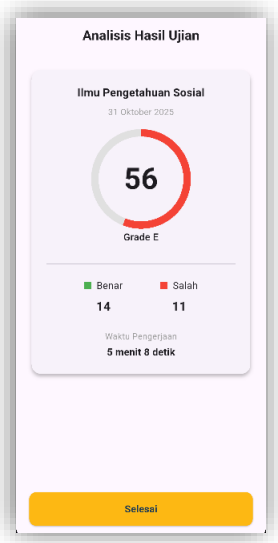

Figure 12 Student Exam Result Analysis Page

Student Exam Results Page

This page displays a list of grades for all exams taken by students. Teachers can also view:

- Student names
- Grades
- Exam dates
- Class average grade

The display is in the form of a complete grade table, so teachers can comprehensively monitor student progress.

Figure 13 Student Exam Results Page

Profile Page

The profile display shows user data:

- Teacher: name, employee ID number, subjects taught, classes taught
- Student: name, student ID number, national student ID number, class

The image displays complete identity details in profile card format. Information is neatly arranged for easy reading.

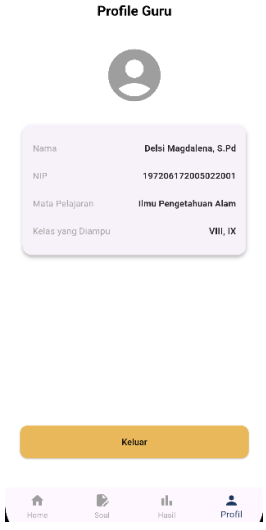

Figure 14 Teacher Profile Page

Discussion

This mobile-based exam application successfully addresses various weaknesses of previous platforms such as Google Form, especially regarding:

- Exam security
- Result integrity
- Cheating prevention

By implementing the Fisher-Yates Shuffling algorithm, the application successfully provides:

- Different question orders for each student
- Fair and evenly distributed randomization
- O(n) computational efficiency
- Improved exam security

The system also has a clear interface architecture for each role, so that the process of data management, exam execution, and grade analysis becomes more structured and easy to use.

CONCLUSION

Based on the findings and the implementation of the mobile-based online examination application utilizing the Fisher-Yates Shuffling Algorithm, it can be concluded that the algorithm is effective in enhancing the security, fairness, and integrity of the assessment process by generating unique question orders for each participant without causing duplication or data loss. The developed application has successfully met the needs of stakeholders—administrators, teachers, and students—through features that support data management, question creation, exam administration, and real-time result presentation. The automated scoring system provides fast, objective, and accurate evaluation results based on the total number of correct answers, thereby improving the efficiency of the assessment process. Furthermore, the use of a UML-based development methodology ensures that the design and analysis stages are conducted systematically and structurally, resulting in an application that aligns with user needs and specifications.

REFERENCES

Astuti, N., & Wibowo, D. (2021). *Analisis Pengembangan Aplikasi Berbasis Android Menggunakan Android Studio*. Jurnal Informatika, 8(2), 145–152.

Aulia, D., & Kurniawan, H. (2021). *Pemodelan Activity Diagram untuk Analisis Proses Bisnis Aplikasi Mobile*. Jurnal Rekayasa Sistem Informasi, 7(3), 112–121.

Dewi, R., & Maulana, F. (2021). *Analisis Sequence Diagram pada Sistem Informasi Pendidikan Berbasis Web*. Jurnal Teknologi dan Komputer, 10(2), 87–94.

Firmansyah, A., & Rachman, I. (2022). *Perancangan Use Case Diagram untuk Sistem Ujian Berbasis Android*. Jurnal Teknologi Informasi Terapan,

12(1), 44–53.

Gunawan, B., & Lestari, A. (2020). *Pemodelan Class Diagram dalam Pengembangan Sistem Informasi Akademik*. Jurnal Sistem dan Teknologi Informasi, 8(4), 205–213.

Hartono, Y., & Yusuf, A. (2024). *Analisis Implementasi Activity Diagram pada Aplikasi Android Berbasis Kotlin*. Jurnal Informatika dan Teknologi, 14(1), 93–102.

Hidayah, F., & Saputra, A. (2024). *Pemanfaatan Android dalam Pengembangan Aplikasi Ujian Online Sekolah Menengah*. Jurnal Ilmiah Informatika Pendidikan, 12(1), 56–64.

Lestari, M., & Ahmad, S. (2023). *Desain Class Diagram sebagai Model Database Aplikasi Mobile*. Jurnal Rekayasa Perangkat Lunak, 13(1), 55–64.

Maulana, D., & Rachmat, A. (2022). *Pemanfaatan Flutter dalam Pengembangan Aplikasi Pendidikan Berbasis Mobile*. Jurnal Teknologi Informasi Pendidikan, 10(3), 211–220.

Nugroho, A., & Rahmawati, I. (2023). *Pemanfaatan Activity Diagram dalam Desain Sistem Aplikasi Pendidikan*. Jurnal Teknologi Komputer dan Sistem Informasi, 15(2), 145–153.

Prasetya, R., Nugroho, D., & Lestari, A. (2020). *Arsitektur Sistem Operasi Android dalam Pengembangan Aplikasi Pendidikan*. Jurnal Teknologi Pendidikan, 9(4), 312–319.

Pratama, D., & Fadilah, M. (2023). *Pemanfaatan UML untuk Desain Aplikasi Ujian Online Berbasis Android*. Jurnal Informatika Nusantara, 17(1), 33–42.

Putra, A., & Santoso, D. (2022). *Optimalisasi Gradle Build pada Android Studio untuk Efisiensi Pengembangan Aplikasi Mobile.* Jurnal Rekayasa Perangkat Lunak, 6(1), 55–64.

Ramdani, T., & Wulandari, F. (2020). *Perancangan Use Case Diagram untuk Sistem Informasi Sekolah*. Jurnal Teknologi Pendidikan, 9(3), 214–222.

Sari, M., & Prasetyo, B. (2021). *Pengenalan Android Studio sebagai IDE untuk Pengembangan Aplikasi Mobile*. Jurnal Sistem Informasi, 10(1), 23–29.

Setiawan, A., & Nurhayati, L. (2022). *Unified Modeling Language (UML) sebagai Alat Bantu Perancangan Sistem Informasi*. Jurnal Sistem Informasi dan Komputer, 12(2), 101–110.

Suharto, E., Nurcahyo, B., & Ramadhan, T. (2022). *Manajemen API Level pada Aplikasi Android Berbasis Pendidikan*. Jurnal Informatika dan Rekayasa Perangkat Lunak, 14(2), 211–220.

Suryani, E., & Putra, H. (2021). *Penerapan UML dalam Pengembangan Aplikasi Mobile Berbasis Android*. Jurnal Ilmiah Teknologi dan Sistem Informasi, 9(2), 87–96.

Utami, S., Wicaksono, H., & Lestari, I. (2022). *Pemanfaatan Kotlin pada Android Studio untuk Meningkatkan Efisiensi Kode Program*. Jurnal Teknologi dan Informasi, 8(1), 93–101.

Wibisono, H., & Hidayah, F. (2024). *Integrasi Flutter dan Supabase untuk Aplikasi Edukasi Real-Time Berbasis Cloud*. Jurnal Rekayasa Perangkat Lunak, 13(2), 156–165

Wibowo, D., & Santoso, T. (2022). *Analisis Class Diagram untuk Pengembangan Sistem Ujian Online Berbasis Database Relasional*. Jurnal Teknologi dan Rekayasa Perangkat Lunak, 10(4), 198–207.

Wijaya, A., & Sari, P. (2021). *Pemanfaatan Use Case Diagram untuk Identifikasi Kebutuhan Sistem Aplikasi Mobile Pendidikan*. Jurnal Informatika Pendidikan, 13(1), 72–80.

Wulandari, R., & Pratama, Y. (2024). *Perbandingan Firebase dan Supabase sebagai Backend untuk Aplikasi Mobile Pendidikan*. Jurnal Teknologi Informasi Terapan, 16(1), 67–78.

Yuliani, R. (2020). *Pengembangan Aplikasi Mobile Learning Berbasis Android untuk Meningkatkan Efektivitas Pembelajaran*. Jurnal Teknologi Informasi dan Pendidikan, 13(1), 25–33.