

PENGEMBANGAN APLIKASI *SAFE EXAM BROWSER* BERBASIS ANDROID TERINTEGRASI MOODLE MENGGUNAKAN METODE *RAPID APPLICATION DEVELOPMENT* DI SMAN 2 SUKAWATI

(Development of an Android-Based Safe Exam Browser Application Integrated with Moodle Using the Rapid Application Development Method at SMAN 2 Sukawati)

Putu Aris Surya Kusuma ^[1], A.A. Istri Ita Paramitha ^[1], Putri Anugrah Cahya Dewi ^[1,2]

^[1] Dept. Informatics Engineering, Mataram University

Jl. Majapahit 62, Mataram, Lombok NTB, INDONESIA

^[2] Dept. Information Systems, Primakara University, Denpasar, Indonesia

Email: putuaris2004@gmail.com, ita@primakara.ac.id, cahya@primakara.ac.id

Abstract

The integration of digital technology in education has increased the use of online assessment systems, particularly through Learning Management Systems. However, the implementation of digital exams at SMAN 2 Sukawati still faces challenges related to academic integrity, as students can switch applications, access external resources, and capture exam content during assessments. This study aims to develop an Android-based Safe Exam Browser application integrated with Moodle to create a secure and controlled exam environment. The application utilizes a device locking mechanism that restricts access to other applications and system functions during the exam. The development process applies the Rapid Application Development method, allowing iterative design and rapid adaptation to user requirements. The system includes features such as application locking, screenshot protection, external link blocking, violation detection, and automated penalty mechanisms. Scenario-based functional testing was conducted involving students and teachers to evaluate whether each feature performed as intended. The results indicate that the developed features—including device locking, violation detection, and the penalty mechanism—successfully functioned according to the defined test scenarios. These findings suggest that the proposed system has the potential to support a more secure and controlled digital exam environment, although further evaluation involving user acceptance and comparative testing is needed to confirm its effectiveness in reducing cheating behavior.

Keywords: Safe exam browser, android application, moodle integration, digital assessment, rapid application development

1. PENDAHULUAN

Perkembangan teknologi informasi yang pesat telah memberikan pengaruh signifikan di berbagai bidang, termasuk pendidikan, mendorong pembelajaran menjadi lebih efisien, fleksibel, dan inovatif [1], [2], [3]. Salah satu implementasi teknologi tersebut adalah *Learning Management System* (LMS), yaitu perangkat lunak yang digunakan untuk mengelola kegiatan pembelajaran secara *online*, mulai dari penyampaian materi hingga administrasi dan dokumentasi pembelajaran [4], [5].

SMAN 2 Sukawati merupakan salah satu institusi pendidikan yang telah mengimplementasikan LMS berbasis Moodle dalam pelaksanaan penilaian kompetensi. Namun, dalam praktiknya masih ditemukan kendala pada aspek keamanan, di mana

siswa berpotensi melakukan kecurangan seperti berpindah aplikasi, mengakses sumber eksternal, serta melakukan tangkapan layar selama ujian berlangsung. Hal ini menunjukkan bahwa sistem yang digunakan belum mampu sepenuhnya mengontrol aktivitas perangkat pengguna selama proses ujian.

Permasalahan tersebut berdampak pada menurunnya objektivitas dan integritas hasil penilaian. Oleh karena itu, diperlukan solusi berupa sistem tambahan yang mampu membatasi akses perangkat selama ujian berlangsung. Pemilihan *platform* Android sebagai basis pengembangan didasarkan pada tingginya tingkat kepemilikan dan penggunaan perangkat Android di kalangan siswa SMAN 2 Sukawati dibandingkan perangkat lain, serta tersedianya fitur *Lock Task Mode* pada sistem

operasi Android yang memungkinkan pembatasan akses aplikasi secara langsung pada tingkat perangkat, tanpa memerlukan instalasi perangkat tambahan. Salah satu solusi yang dapat diterapkan adalah pengembangan aplikasi *Safe Exam Browser* berbasis Android dengan memanfaatkan fitur *Lock Task Mode* untuk mengunci perangkat agar tetap berada pada aplikasi ujian [6].

Pengembangan sistem dalam penelitian ini menggunakan metode *Rapid Application Development* (RAD), yang berfokus pada pembangunan prototipe secara cepat serta pengujian iteratif untuk menyempurnakan sistem berdasarkan umpan balik pengguna [7]. Selain itu, pengujian sistem dilakukan menggunakan metode pengujian fungsional berbasis skenario (*scenario-based black-box testing*) untuk memastikan bahwa setiap fitur pada sistem yang dikembangkan berjalan sesuai dengan spesifikasi yang telah ditentukan [8]. Pendekatan ini memungkinkan pengembang untuk merespons perubahan dengan cepat serta meningkatkan kualitas sistem secara berkelanjutan [9]. Berdasarkan permasalahan tersebut, penelitian ini bertujuan untuk mengembangkan aplikasi *Safe Exam Browser* berbasis Android yang terintegrasi dengan LMS Moodle guna meningkatkan keamanan serta integritas pelaksanaan ujian digital di SMAN 2 Sukawati.

2. TINJAUAN PUSTAKA

Pemanfaatan *Learning Management System* (LMS) dalam pembelajaran digital telah banyak diteliti sebagai solusi dalam pengelolaan pembelajaran daring. LMS memungkinkan distribusi materi, pelaksanaan evaluasi, serta interaksi antara guru dan siswa secara lebih fleksibel dan terstruktur. Penelitian sebelumnya menunjukkan bahwa penggunaan LMS mampu meningkatkan efektivitas pembelajaran serta kemandirian siswa dalam mengakses materi dan mengerjakan tugas [10]. Studi lain yang mengintegrasikan aplikasi Android dengan platform LMS Moodle melalui REST API juga menunjukkan bahwa pendekatan *mobile learning* berbasis Android mampu meningkatkan fleksibilitas akses serta tingkat kepuasan pengguna terhadap sistem pembelajaran [11]. Namun, penelitian-penelitian tersebut umumnya masih berfokus pada aspek distribusi materi dan interaksi pembelajaran, dan belum membahas secara spesifik terkait keamanan pelaksanaan ujian digital.

Beberapa penelitian telah mengembangkan sistem ujian berbasis *exam browser* untuk meningkatkan keamanan ujian, dengan fitur seperti pembatasan akses aplikasi lain [3], penguncian layar penuh dan pemblokiran akses eksternal [6], serta pemantauan aktivitas perangkat peserta [13], seluruhnya terbukti efektif mengurangi peluang kecurangan. Secara lebih spesifik, penerapan *Lock Task Mode* pada *Safe Exam Browser* berbasis Android terbukti mampu memaksa aplikasi ujian tetap berada di latar depan meskipun pengguna berupaya memindahkannya ke latar belakang, sehingga integritas sesi ujian tetap terjaga [12]. Meskipun demikian, penelitian-penelitian tersebut umumnya masih berfokus pada sistem berbasis *desktop* atau belum mengoptimalkan integrasi dengan LMS serta pengamanan perangkat *mobile* secara spesifik.

Dari sisi metode pengembangan, penerapan *Rapid Application Development* (RAD) pada aplikasi berbasis Android dinilai sesuai untuk proyek dengan ruang lingkup terbatas dan kebutuhan pengembangan yang cepat, di mana keterlibatan pengguna dalam proses perancangan menjadi faktor penting keberhasilan sistem [14]. Sementara itu, dari sisi pengujian, pendekatan pengujian fungsional berbasis skenario (*black-box testing*) pada aplikasi Android terbukti efektif digunakan untuk memastikan bahwa setiap fitur pada sistem berjalan sesuai dengan spesifikasi yang telah ditentukan, tanpa memerlukan instrumen penerimaan pengguna yang kompleks [8].

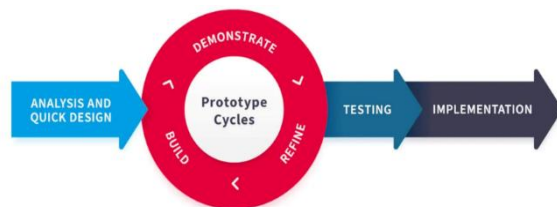
Berdasarkan keterbatasan tersebut, penelitian ini mengusulkan pengembangan aplikasi *Safe Exam Browser* berbasis Android yang terintegrasi dengan LMS Moodle dengan pendekatan keamanan yang lebih komprehensif. Sistem yang dikembangkan tidak hanya membatasi akses aplikasi melalui *Lock Task Mode*, tetapi juga dilengkapi dengan fitur deteksi pelanggaran, sistem penalti bertahap, serta manajemen konfigurasi ujian berbasis *file JSON*. Dengan demikian, penelitian ini memberikan kontribusi dalam pengembangan sistem ujian digital yang tidak hanya terintegrasi dengan LMS, tetapi juga mampu meningkatkan keamanan dan integritas pelaksanaan ujian secara lebih optimal pada perangkat *mobile*.

3. METODE PENELITIAN

Penelitian ini menggunakan metode *Rapid Application Development* (RAD) sebagai pendekatan pengembangan aplikasi *Safe Exam Browser* berbasis

Android, dipilih karena menekankan kecepatan pembangunan sistem serta fleksibilitas penyesuaian kebutuhan pengguna melalui siklus iteratif [15]. Metode ini memungkinkan pengembang membangun prototipe secara cepat, memperoleh umpan balik pengguna (guru dan staf IT) sejak tahap awal, serta melakukan perbaikan berkelanjutan agar sistem sesuai dengan kondisi pelaksanaan ujian digital di SMAN 2 Sukawati.

Tahapan dalam metode RAD yang digunakan pada penelitian ini meliputi *Analysis and Quick Design*, *Prototype Cycles (Build, Demonstrate, Refine)*, *Testing*, dan *Implementation*. Alur tahapan metode RAD yang digunakan dalam penelitian ini ditunjukkan pada Gambar 1.



Gambar 1. Ilustrasi Metode RAD

3.1. Analysis and Quick Design

Tahap ini diawali dengan analisis kebutuhan melalui observasi dan wawancara bersama staf IT serta guru, untuk mengidentifikasi potensi kecurangan yang terjadi dalam pelaksanaan ujian berbasis LMS Moodle di SMAN 2 Sukawati, seperti perpindahan aplikasi dan akses sumber eksternal. Hasil analisis menjadi dasar penentuan kebutuhan dan batasan sistem, khususnya penerapan *Lock Task Mode*, sesuai prinsip RAD yang menekankan identifikasi kebutuhan pengguna pada tahap awal [16].

Selanjutnya dilakukan tahap *quick design* berupa perancangan awal antarmuka dan alur sistem menggunakan Figma, mencakup navigasi aplikasi, akses ujian Moodle, serta sistem penalti.

3.1.1. Analisis Kebutuhan

Analisis kebutuhan dilakukan melalui wawancara dan diskusi dengan pihak sekolah, khususnya staf IT dan guru di SMAN 2 Sukawati. Hasil analisis menunjukkan bahwa meskipun LMS Moodle telah digunakan sebagai media ujian digital, masih terdapat potensi kecurangan, seperti membuka mesin pencari dan berpindah aplikasi selama ujian berlangsung. Oleh karena itu, diperlukan aplikasi pendukung yang mampu meningkatkan keamanan pelaksanaan ujian.

Kebutuhan sistem yang dihasilkan dibagi menjadi kebutuhan fungsional dan non-fungsional. Kebutuhan fungsional sistem, yang menjelaskan fitur-fitur utama yang harus disediakan oleh aplikasi, dirangkum pada Tabel 1 berikut.

Tabel 1. Kebutuhan Fungsional

ID	Requirement Statement
FR-01	Sistem menyediakan fitur pembuatan dan pengelolaan konfigurasi (membuat dan mengedit konfigurasi) yang berisi URL ujian, token, dan durasi ujian.
FR-02	Sistem menghasilkan file konfigurasi ujian yang dapat didistribusikan kepada siswa.
FR-03	Sistem memungkinkan siswa membuka konfigurasi ujian dan mengakses halaman ujian Moodle melalui <i>WebView</i> .
FR-04	Sistem menerapkan <i>Lock Task Mode</i> untuk membatasi akses pengguna agar tetap berada dalam aplikasi selama ujian berlangsung.
FR-05	Sistem mendeteksi pelanggaran seperti perpindahan aplikasi dan memberikan penalti secara otomatis.
FR-06	Sistem menampilkan informasi perangkat (waktu, baterai, dan jaringan) selama ujian berlangsung.

Selain kebutuhan fungsional, penelitian ini juga mengidentifikasi kebutuhan non-fungsional sistem yang mencakup aspek performa, keamanan, usabilitas, portabilitas, dan reliabilitas. Kebutuhan non-fungsional tersebut dirangkum pada Tabel 2 berikut.

Tabel 2. Kebutuhan Non-Fungsional

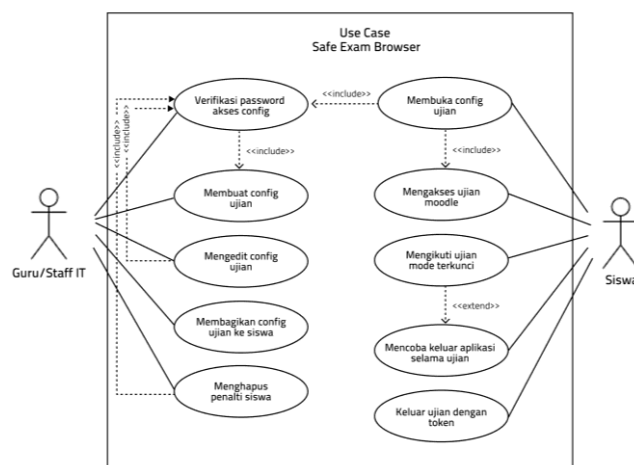
ID	Requirement Statement	Quality Attribute
NFR-01	Sistem menampilkan halaman konfigurasi dan ujian tanpa jeda yang signifikan berdasarkan pengamatan langsung, dengan kecepatan dipengaruhi oleh kondisi jaringan pengguna.	Performance
NFR-02	Sistem menerapkan mekanisme keamanan untuk mencegah akses aplikasi lain dan aktivitas <i>screenshot</i> .	Security
NFR-03	Sistem dapat digunakan oleh siswa tanpa kendala berarti berdasarkan pengamatan langsung selama pengujian, tanpa memerlukan pelatihan khusus.	Usability
NFR-04	Sistem dapat berjalan dengan baik pada berbagai merek perangkat Android milik siswa, diamati pada sekitar 8 merek perangkat berbeda selama pengujian.	Portability

NFR-05	Sistem tidak mengalami <i>crash</i> atau <i>error</i> selama sesi pengujian berlangsung, berdasarkan pengamatan langsung terhadap seluruh partisipan.	<i>Reliability</i>
--------	---	--------------------

3.1.2. Perancangan Sistem

Pemodelan sistem dilakukan untuk menggambarkan interaksi antara pengguna dengan aplikasi yang dikembangkan. Representasi sistem disajikan menggunakan *use case diagram* yang menunjukkan aktor serta fungsi-fungsi utama yang dapat diakses dalam sistem. Diagram ini digunakan untuk memberikan gambaran umum mengenai peran pengguna, yaitu guru/staf IT dan siswa, dalam menggunakan aplikasi selama pelaksanaan ujian.

a. Use Case Diagram



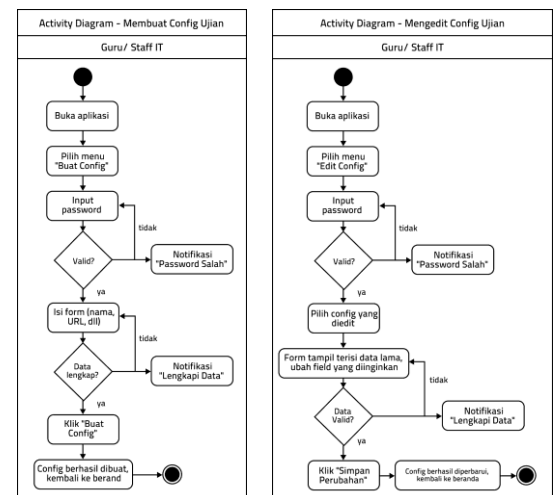
Gambar 2. Use Case Diagram

Gambar 2 menunjukkan interaksi antara dua aktor utama, yaitu Guru/Staf IT dan Siswa, dengan sistem *Safe Exam Browser*. Guru/Staf IT memiliki empat *use case*: membuat, mengedit, dan membagikan *config* ujian, serta menghapus penalti siswa, di mana ketiga *use case* pertama (kecuali membagikan) mensyaratkan relasi *include* terhadap "Verifikasi Password Akses Config" sebagai mekanisme keamanan dasar.

b. Activity Diagram

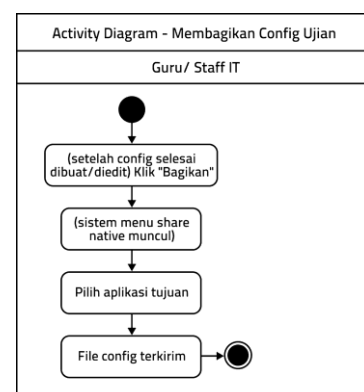
Activity diagram digunakan untuk menggambarkan alur proses yang terjadi dalam sistem selama pelaksanaan ujian. Mengacu pada prinsip satu *use case* direpresentasikan oleh satu *activity diagram*, penelitian ini menghasilkan tujuh *activity diagram* yang masing-masing menggambarkan proses pada level *use case*. Setiap diagram hanya menampilkan satu jalur aktivitas

sesuai aktor yang menjalankannya (Guru/Staf IT atau Siswa), tanpa jalur tersendiri untuk sistem, respon sistem ditampilkan sebagai kelanjutan langsung dari aktivitas pengguna dalam jalur yang sama. Dengan adanya *activity diagram*, alur kerja sistem dapat divisualisasikan secara lebih jelas dan terstruktur.



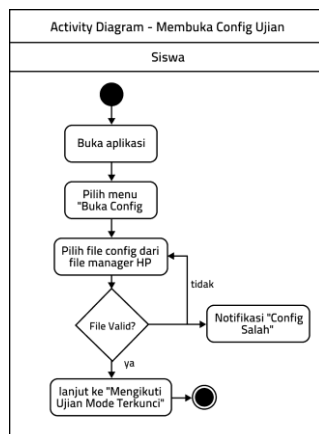
Gambar 3. Activity Diagram Membuat Config Ujian dan Gambar 4. Mengedit Config Ujian

Gambar 3 dan Gambar 4 menunjukkan alur proses pembuatan dan pengeditan konfigurasi ujian oleh Guru/Staf IT. Kedua proses memiliki struktur yang serupa, yaitu diawali dengan verifikasi *password* sebagai mekanisme keamanan, sesuai relasi *include* pada *use case diagram*. Pada Gambar 3, pengguna mengisi data konfigurasi baru (nama ujian, URL, token, dan durasi) hingga sistem menghasilkan *file* konfigurasi. Pada Gambar 4, pengguna memilih konfigurasi yang telah ada, kemudian melakukan perubahan pada data yang sudah tersimpan, sebelum sistem menyimpan pembaruan tersebut.



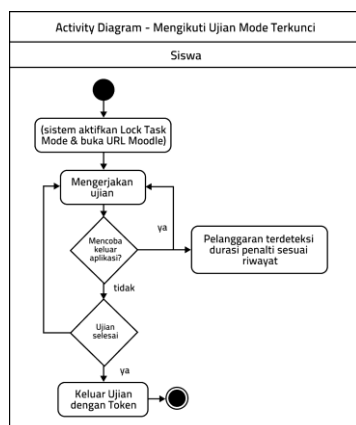
Gambar 5. Activity Diagram Membagikan Config Ujian

Gambar 5 menggambarkan alur proses pembagian konfigurasi ujian oleh Guru/Staf IT kepada siswa. Setelah konfigurasi selesai dibuat atau diperbarui, pengguna dapat membagikan *file* konfigurasi tersebut melalui fitur berbagi bawaan sistem operasi Android kepada media distribusi yang dipilih, seperti aplikasi perpesanan atau penyimpanan awan.



Gambar 6. Activity Diagram Membuka Config Ujian

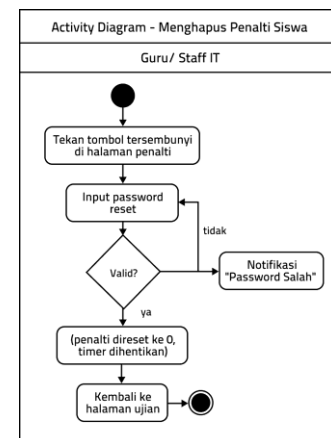
Gambar 6 menggambarkan alur proses siswa dalam membuka konfigurasi ujian yang telah diterima dari guru. Siswa memilih *file* konfigurasi melalui pengelola berkas pada perangkat, kemudian sistem memuat dan memvalidasi isi berkas tersebut. Apabila berkas valid, proses dilanjutkan ke tahap mengikuti ujian dalam mode terkunci sebagaimana dijelaskan pada Gambar 7.



Gambar 7. Activity Diagram Mengikuti Ujian Mode Terkunci

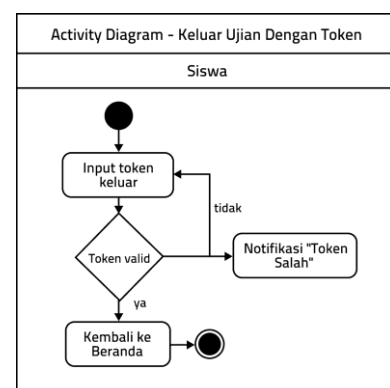
Gambar 7 menggambarkan alur proses siswa selama mengikuti ujian dalam mode terkunci, merupakan kelanjutan dari proses pada Gambar 6. Setelah *Lock Task Mode* aktif dan halaman ujian Moodle terbuka, siswa mengerjakan ujian dalam kondisi perangkat terkunci pada aplikasi. Apabila

siswa berupaya keluar dari aplikasi selama ujian berlangsung, sistem akan mendeteksi hal tersebut sebagai pelanggaran dan menerapkan penalti sesuai riwayat pelanggaran siswa, sebagaimana dijelaskan lebih lanjut pada Gambar 8. Proses mengerjakan ujian akan terus berlanjut hingga waktu ujian selesai, sebelum dilanjutkan ke proses keluar ujian pada Gambar 9.



Gambar 8. Activity Diagram Menghapus Penalti Siswa

Gambar 8 menggambarkan alur proses penghapusan penalti siswa oleh Guru/Staf IT. Mekanisme ini disediakan sebagai bentuk kendali manual bagi guru terhadap penalti yang diterima siswa, khususnya apabila pelanggaran terjadi bukan akibat kecurangan yang disengaja, melainkan kendala teknis seperti panggilan telepon masuk atau gangguan sistem pada perangkat. Guru mengakses fitur ini melalui tombol tersembunyi pada halaman penalti, kemudian memasukkan *password* verifikasi sebelum sistem mereset penalti siswa ke kondisi awal.



Gambar 9. Activity Diagram Keluar Ujian Dengan Token

Gambar 9 menggambarkan alur proses siswa saat mengakhiri ujian, merupakan kelanjutan dari proses pada Gambar 7. Setelah waktu ujian selesai,

siswa memasukkan token keluar yang telah ditentukan pada konfigurasi ujian. Apabila token valid, sistem akan menonaktifkan *Lock Task Mode* dan mengembalikan perangkat ke kondisi normal.

3.2. Prototype Cycles

Tahap *Prototype Cycles* merupakan inti dari metode *Rapid Application Development* (RAD) yang dilakukan secara iteratif. Pada tahap ini dilakukan proses pembangunan prototipe sistem, demonstrasi kepada pengguna, serta penyempurnaan sistem berdasarkan umpan balik yang diperoleh. Pendekatan ini memungkinkan pengembangan aplikasi dilakukan secara cepat dan adaptif terhadap kebutuhan pengguna. Tahapan pada *Prototype Cycles* terdiri dari *Build*, *Demonstrate*, dan *Refine*.

3.2.1. Build

Penerapan *Lock Task Mode* menggunakan mekanisme screen pinning bawaan Android (*startLockTask*), bukan *Device Owner/Device Policy Controller*, karena tidak memerlukan *provisioning* khusus sehingga lebih mudah diterapkan pada perangkat siswa yang beragam (target minimum Android 8.0/API 26). Selama mode terkunci aktif, sistem secara otomatis memblokir:

- a. *Notification panel*, sepenuhnya tidak dapat diakses;
- b. Skenario *multi-window* (*split-screen*, *picture-in-picture*, *floating overlay*), tidak dapat diaktifkan tanpa penanganan tambahan.

Pada kondisi *restart* perangkat selama ujian, sistem mendeteksinya sebagai pelanggaran saat aplikasi dibuka kembali sebelum mengembalikan siswa ke sesi ujian. Sebaliknya, panggilan telepon masuk tidak memicu pelanggaran, karena sistem hanya menampilkan getaran tanpa antarmuka panggilan yang dapat mengalihkan fokus siswa.

Meskipun *screen pinning* memberikan keamanan yang memadai, terdapat keterbatasan yang perlu diakui: pengguna dapat melepas status terkunci dengan menahan tombol *Back* dan *Recent Apps* bersamaan (mode navigasi tombol) atau gestur *swipe* bawah-atas (mode navigasi gestur). Namun, tindakan tersebut langsung terdeteksi sebagai pelanggaran dan memicu penalti, sehingga tetap terpantau sistem. Sebagai jalur keluar darurat bagi pengawas, tersedia tombol tersembunyi pada halaman penalti yang hanya diketahui guru/staf IT,

dapat diakses khusus saat siswa berada dalam kondisi penalti.

Sistem penalti diterapkan secara bertingkat. Dimulai dari 3 menit untuk pelanggaran pertama dan 10 menit untuk pelanggaran berikutnya berdasarkan kesepakatan dengan pihak SMAN 2 Sukawati agar memberikan efek jera yang lebih kuat dibanding durasi tunggal. Selama masa penalti, siswa dialihkan ke halaman penalti dan tidak dapat kembali ke sesi ujian hingga waktu berakhir. Untuk mengantisipasi pelanggaran akibat kendala teknis (bukan kecurangan disengaja), guru/staf IT dapat menghapus penalti melalui tombol tersembunyi disertai verifikasi *password*, sebagaimana digambarkan pada Gambar 8.

3.2.2. Demonstrate

Tahap *Demonstrate* dilakukan dengan menampilkan prototipe aplikasi kepada pengguna, yaitu guru dan staf IT di SMAN 2 Sukawati. Pada tahap ini pengguna dapat mencoba fungsi utama aplikasi, seperti pembuatan konfigurasi ujian dan mekanisme penguncian perangkat selama ujian berlangsung. Tujuan dari tahap ini adalah untuk memperoleh umpan balik terkait tampilan antarmuka dan alur penggunaan aplikasi, sehingga sistem yang dikembangkan dapat disesuaikan dengan kebutuhan pengguna di lapangan.

3.2.3. Refine

Tahap *Refine* merupakan proses penyempurnaan sistem berdasarkan hasil evaluasi dari tahap demonstrasi. Perbaikan dilakukan pada aspek fungsionalitas maupun antarmuka pengguna untuk meningkatkan kinerja dan kemudahan penggunaan aplikasi. Selain itu, dilakukan pengujian awal dengan menjalankan aplikasi pada perangkat Android untuk memastikan fitur utama, seperti penguncian aplikasi, pemblokiran tangkapan layar, dan mekanisme penalti, dapat berjalan dengan baik sebelum memasuki tahap pengujian sistem secara keseluruhan.

3.3. Testing

Tahap pengujian dilakukan untuk memastikan bahwa aplikasi *Safe Exam Browser* dapat berfungsi sesuai dengan kebutuhan pengguna di SMAN 2 Sukawati. Pengujian sistem menggunakan metode pengujian fungsional berbasis skenario (*scenario-based black-box testing*) [8], dengan melibatkan pengguna secara langsung, yaitu guru dan siswa,

untuk menjalankan setiap skenario pengujian sesuai fitur yang telah dikembangkan.

Pada tahap ini, aplikasi dibangun dalam bentuk *file* APK dan diinstal pada perangkat Android untuk diuji dalam kondisi penggunaan nyata. Pengujian dilakukan terhadap seluruh fitur utama sistem, mencakup pembuatan dan pengeditan konfigurasi ujian, pembagian konfigurasi kepada siswa, pembukaan konfigurasi dan akses ujian Moodle, pelaksanaan ujian dalam mode terkunci, mekanisme deteksi pelanggaran dan penalti, penghapusan penalti oleh guru, serta proses keluar ujian menggunakan token. Ringkasan skenario pengujian beserta hasilnya disajikan pada Tabel 3 dan Tabel 4 di Bab 4.

Selain itu, pengujian juga mencakup skenario normal maupun skenario kesalahan atau kondisi ekstrem (seperti input *file* konfigurasi tidak valid, *field* yang kosong, atau kata sandi yang salah) untuk memastikan sistem mampu menangani berbagai kemungkinan yang terjadi selama ujian berlangsung. Hasil pengujian ini digunakan untuk mengevaluasi kesesuaian sistem dengan kebutuhan pengguna sebelum memasuki tahap implementasi.

3.4. Implementation

Tahap implementasi merupakan proses penerapan aplikasi *Safe Exam Browser* pada lingkungan nyata di SMAN 2 Sukawati. Aplikasi yang telah melalui tahap pengujian kemudian didistribusikan kepada pengguna dalam bentuk *file* APK untuk diinstal pada perangkat Android siswa.

Dalam penerapannya, guru atau staf IT terlebih dahulu membuat dan mendistribusikan konfigurasi ujian kepada siswa. Selanjutnya, siswa menggunakan aplikasi untuk membuka konfigurasi tersebut dan mengakses halaman ujian Moodle melalui sistem yang telah dilengkapi dengan mekanisme pengamanan, seperti pembatasan akses aplikasi dan sistem penalti.

Hasil implementasi menunjukkan bahwa aplikasi dapat digunakan dalam mendukung pelaksanaan ujian digital serta membantu membatasi aktivitas siswa di luar aplikasi selama ujian berlangsung.

4. HASIL DAN PEMBAHASAN

Pengembangan sistem dilakukan dengan mengacu pada metode *Rapid Application Development* (RAD) yang menekankan proses iteratif dan kecepatan dalam menghasilkan solusi yang sesuai dengan kebutuhan pengguna. Pendekatan ini memungkinkan adanya penyesuaian

secara berkelanjutan melalui keterlibatan pengguna pada setiap tahapan pengembangan. Dengan demikian, sistem yang dihasilkan tidak hanya berfungsi secara teknis, tetapi juga relevan dengan kondisi dan permasalahan yang dihadapi di lapangan.

Pelaksanaan pengembangan mengikuti tahapan dalam metode RAD yang meliputi *Analysis and Quick Design*, *Prototype Cycles*, *Testing*, dan *Implementation*. Setiap tahapan memiliki peran penting dalam memastikan sistem dapat dikembangkan secara terstruktur, mulai dari proses identifikasi kebutuhan hingga penerapan sistem secara nyata. Hasil dari setiap tahapan tersebut kemudian menjadi dasar dalam merealisasikan fitur-fitur utama pada aplikasi *Safe Exam Browser* sesuai dengan kebutuhan pengguna.

4.1 Hasil Metode RAD

Pelaksanaan pengembangan sistem sudah berjalan mengikuti timeline yang telah disepakati dengan menggunakan metode *Rapid Application Development* (RAD), sebagaimana telah dijelaskan pada Gambar 1 di Bab 3. Alur setiap tahapan pada proyek ini berdasarkan metode RAD, yakni: (1) *Analysis and Quick Design*, (2) *Prototype Cycles*, (3) *Testing*, dan (4) *Implementation*.

4.1.1 Analysis and Quick Design

Tahap *Analysis and Quick Design* dilaksanakan dengan melakukan observasi dan diskusi bersama staf IT serta guru pada bulan November di SMAN 2 Sukawati. Pada tahap ini diperoleh informasi mengenai permasalahan yang terjadi pada pelaksanaan ujian berbasis LMS Moodle. Selanjutnya dilakukan perancangan tampilan antarmuka menggunakan Figma sebagai acuan dalam proses pengembangan aplikasi *Safe Exam Browser*.

4.1.2 Prototype Cycles

Tahap *Prototype Cycles* dilakukan setelah proses analisis dan perancangan sistem selesai. Pada tahap ini dilakukan proses pengembangan aplikasi *Safe Exam Browser* secara bertahap yang meliputi kegiatan *build*, *demonstrate*, dan *refine*.

a. Build

Pada tahap ini dilakukan proses pengembangan aplikasi *Safe Exam Browser* menggunakan Kotlin dan XML pada Android Studio. Fitur utama yang dibangun meliputi pembuatan konfigurasi ujian, pembukaan

konfigurasi oleh siswa, integrasi *WebView* untuk akses Moodle, serta penerapan *Lock Task Mode* dan sistem penalti.

b. Demonstrate

Tahap ini dilakukan dengan menguji coba aplikasi secara langsung bersama guru dan staf IT di SMAN 2 Sukawati. Pengguna mencoba fitur utama aplikasi dan memberikan masukan terkait kemudahan penggunaan serta kesesuaian sistem dengan kebutuhan ujian di sekolah.

c. Refine

Tahap ini dilakukan dengan memperbaiki sistem berdasarkan hasil uji coba. Perbaikan difokuskan pada peningkatan fungsi aplikasi, terutama pada mekanisme penalti dan stabilitas sistem agar dapat digunakan secara optimal.

4.1.3 Testing

Pengujian sistem dilaksanakan pada tanggal 1 April 2026 di SMAN 2 Sukawati dengan melibatkan 20 siswa dalam satu kelas sebagai pengguna utama, serta 5 guru sebagai penguji dari sisi pengelolaan sistem. Metode pengujian yang digunakan adalah pengujian fungsional berbasis skenario (*scenario-based black-box testing*) [8], yaitu pengujian yang dilakukan secara langsung oleh pengguna berdasarkan skenario penggunaan nyata aplikasi.

Pengujian dilakukan dengan pendekatan berbasis skenario yang mencakup seluruh fitur utama sistem, baik dari sisi siswa maupun guru. Setiap skenario diuji secara bertahap, dimulai dari skenario normal hingga skenario kesalahan atau kondisi ekstrem, seperti kesalahan *input*, pelanggaran aturan sistem, serta percobaan akses yang tidak sesuai. Adapun detail langkah pengujian, data uji, dan hasil pengujian secara lengkap disajikan pada lampiran. Adapun hasil ringkasan skenario yang telah dilakukan:

Tabel 3. Ringkasan Skenario Pengujian Sistem Siswa

Skenario Pengujian	Expected Result	Actual Result	Status
Membuka <i>file</i> konfigurasi (JSON)	Sistem berhasil memuat dan membaca isi <i>file</i> konfigurasi JSON	<i>File</i> JSON berhasil dimuat dan diproses sistem	Berhasil
Validasi <i>file</i> (menolak <i>file</i> selain JSON)	Sistem menolak <i>file</i> dengan format selain JSON dan	Sistem menolak <i>file</i> non-JSON dan	Berhasil

Skenario Pengujian	Expected Result	Actual Result	Status
	menampilkan notifikasi kesalahan	menampilkan notifikasi	
Menjalankan ujian Moodle	Halaman ujian Moodle tampil dan dapat diakses dalam <i>WebView</i> aplikasi	Halaman ujian Moodle tampil sesuai dan dapat digunakan	Berhasil
Pengujian <i>Lock Task Mode</i> (blokir tombol navigasi)	Tombol <i>Home</i> , <i>Back</i> , dan <i>Recent Apps</i> tidak berfungsi selama sesi ujian berlangsung	Seluruh tombol navigasi terblokir selama ujian	Berhasil
Proteksi <i>screenshot</i> selama ujian	Upaya <i>screenshot</i> ditolak dengan pemberitahuan sistem	<i>Screenshot</i> berhasil diblokir oleh sistem	Berhasil
Pemblokiran akses <i>link</i> eksternal	Tautan menuju halaman di luar domain ujian tidak dapat diakses	Akses ke <i>link</i> eksternal berhasil diblokir	Berhasil
Sistem penalti saat keluar aplikasi	Sistem mendeteksi upaya keluar aplikasi dan menjalankan mekanisme penalti sesuai durasi yang ditentukan	Pelanggaran terdeteksi dan penalti diterapkan sesuai skema	Berhasil
Validasi token keluar ujian	Sistem menerima token yang valid dan menonaktifkan <i>Lock Task Mode</i> ; menolak token yang salah	Token valid berhasil menonaktifkan <i>Lock Task Mode</i> , token salah ditolak	Berhasil
<i>Monitoring</i> indikator (baterai, jaringan, waktu)	Indikator baterai, status jaringan, dan sisa waktu ujian ditampilkan secara <i>real-time</i>	Ketiga indikator tampil dan terupdate secara <i>real-time</i>	Berhasil

Tabel 4. Ringkasan Skenario Pengujian Sistem Guru

Skenario Pengujian	Expected Result	Actual Result	Status
Akses fitur membuat	Sistem meminta password	Sistem menampilkan	Berhasil

Skenario Pengujian	Expected Result	Actual Result	Status
konfigurasi dengan <i>password</i>	sebelum mengizinkan akses ke fitur pembuatan konfigurasi	<i>form password</i> sebelum lanjut ke fitur	
Validasi <i>password</i> (benar & salah)	<i>Password</i> benar mengizinkan akses; <i>password</i> salah ditolak dengan notifikasi	<i>Password</i> benar diterima, <i>password</i> salah ditolak dan menampilkan notifikasi	Berhasil
Mengisi dan validasi data <i>config</i>	Sistem menerima input data <i>config</i> yang lengkap dan valid	Data <i>config</i> berhasil diinput dan divalidasi sistem	Berhasil
Membuat <i>file</i> konfigurasi JSON	Sistem menghasilkan <i>file</i> konfigurasi JSON sesuai data yang diinput	<i>File</i> JSON berhasil dibuat sesuai input	Berhasil
Mengedit konfigurasi yang sudah dibuat	Sistem menampilkan data <i>config</i> lama dan menyimpan perubahan yang dilakukan	Data lama tampil dan perubahan berhasil disimpan	Berhasil
Preview konfigurasi	Sistem menampilkan ringkasan data <i>config</i> sebelum disimpan/dibagikan	Ringkasan <i>config</i> tampil sesuai data yang diinput	Berhasil
Validasi input (<i>field</i> kosong)	Sistem menolak proses jika ada <i>field</i> wajib yang kosong, disertai notifikasi	Sistem menampilkan notifikasi dan menolak proses saat <i>field</i> kosong	Berhasil
Reset penalti siswa	Sistem menerima verifikasi <i>password</i> guru dan mereset penalti siswa ke nol	Penalti berhasil mereset setelah verifikasi <i>password</i>	Berhasil

Secara keseluruhan, hasil pengujian menunjukkan bahwa seluruh skenario yang diuji berhasil dijalankan sesuai dengan hasil yang diharapkan (*expected result*), baik pada sisi siswa maupun guru. Hal ini mengindikasikan bahwa fitur-fitur utama

sistem telah berfungsi sesuai dengan spesifikasi yang dirancang.

Selain pengujian terhadap kebutuhan fungsional, dilakukan pula pengamatan terhadap pemenuhan kebutuhan non-fungsional selama proses pengujian berlangsung. Dari sisi performa, proses pembukaan konfigurasi dan halaman ujian berjalan tanpa jeda yang signifikan, dengan kecepatan yang dipengaruhi oleh kondisi jaringan masing-masing pengguna. Dari sisi reliabilitas, tidak ditemukan *crash* maupun *error* pada aplikasi selama sesi pengujian berlangsung, baik pada sisi siswa maupun guru. Dari sisi portabilitas, aplikasi berhasil dijalankan pada perangkat Android milik siswa yang beragam, mencakup sekitar delapan merek perangkat berbeda, tanpa ditemukan kendala kompatibilitas. Dari sisi *usability*, seluruh partisipan dapat menggunakan aplikasi tanpa memerlukan pelatihan khusus maupun mengalami kebingungan yang berarti selama proses pengujian.

4.1.4 Implementation

Tahap implementasi dilakukan dengan menerapkan aplikasi *Safe Exam Browser* pada lingkungan SMAN 2 Sukawati. Pada tahap ini aplikasi yang telah dikembangkan dibangun menjadi *file* APK dan didistribusikan melalui Google Drive kepada siswa untuk diunduh pada perangkat Android masing-masing. Selain itu, dilakukan juga koordinasi dengan guru dan staf IT dalam penggunaan aplikasi untuk mendukung pelaksanaan ujian berbasis digital.

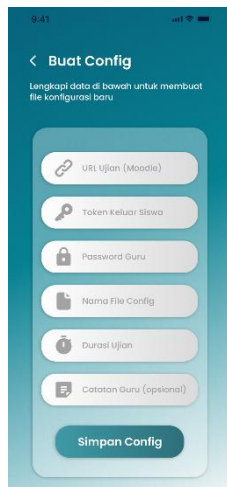
4.2 Implementasi Fitur Sistem

Pada tahap ini dijelaskan hasil implementasi fitur pada aplikasi *Safe Exam Browser* yang telah dikembangkan. Implementasi dilakukan pada perangkat Android dan menampilkan fungsi sistem secara nyata sesuai dengan kebutuhan yang telah dirancang sebelumnya.

4.2.1 Halaman Buat Konfigurasi

Halaman buat *config* digunakan untuk menghasilkan konfigurasi ujian, dengan input berupa URL ujian, token keluar, nama *file config*, waktu keluar, dan catatan guru. Sistem memvalidasi seluruh *input*, menampilkan notifikasi jika ada data kosong sebelum menghasilkan *file* konfigurasi yang

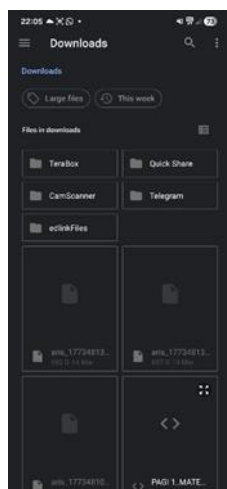
dapat dibagikan kepada siswa sebagai akses masuk ke ujian.



Gambar 6. Halaman Buat Config

4.2.2 Halaman Buka Konfigurasi

Siswa menggunakan fitur membuka konfigurasi melalui *file* manager perangkat. Sistem kemudian membaca isi konfigurasi dan otomatis mengarahkan siswa ke halaman ujian Moodle melalui *WebView*, tanpa proses input manual sehingga mengurangi potensi kesalahan pengguna.

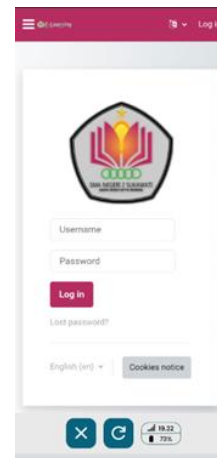


Gambar 7. Halaman Buka Config

4.2.3 Halaman Ujian dan Lock Task Mode

Halaman ujian ditampilkan melalui *WebView* terintegrasi LMS Moodle, yang otomatis mengaktifkan *Lock Task Mode* sehingga siswa tidak dapat keluar aplikasi, membuka aplikasi lain, melakukan *screenshot*, maupun menerima gangguan dari luar. Indikator waktu, status jaringan, dan persentase baterai ditampilkan secara *real-time*

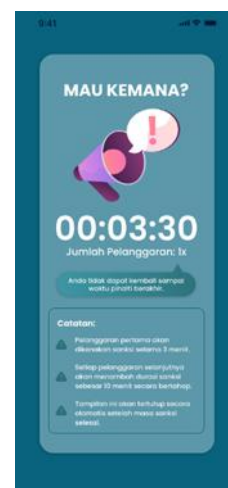
untuk membantu pengguna dan pengawas memantau kondisi perangkat selama ujian.



Gambar 8. Halaman Ujian dan Lock Task Mode

4.2.4 Halaman Penalti

Halaman penalti ditampilkan saat sistem mendeteksi pelanggaran, seperti berpindah atau keluar aplikasi saat ujian. Durasi penalti dihitung otomatis dan bertingkat. Dimulai dari 3 menit untuk pelanggaran pertama, 10 menit untuk pelanggaran berikutnya, dan selama itu siswa tidak dapat kembali ke halaman ujian. Sebagai kendali manual, guru/staf IT dapat menghapus penalti melalui mekanisme yang telah dijelaskan pada sub-bab 3.2.1, khususnya untuk pelanggaran akibat kendala teknis, bukan kecurangan disengaja.



Gambar 9. Halaman Penalti

5. KESIMPULAN DAN SARAN

Berdasarkan hasil perancangan, pengembangan, dan implementasi yang telah dilakukan, aplikasi *Safe Exam Browser* berhasil dikembangkan sebagai

solusi untuk meningkatkan keamanan pelaksanaan ujian digital di SMAN 2 Sukawati. Aplikasi ini dirancang untuk mengurangi potensi kecurangan siswa melalui mekanisme penguncian perangkat (*Lock Task Mode*), pembatasan akses aplikasi lain, konfigurasi ujian berbasis *file* yang memudahkan distribusi oleh guru, serta integrasi langsung dengan LMS Moodle melalui *WebView* dalam lingkungan yang telah diamankan.

Hasil pengujian menunjukkan bahwa seluruh fitur, termasuk deteksi pelanggaran dan sistem penalti otomatis, berjalan sesuai fungsinya. Dengan demikian, aplikasi yang dikembangkan berpotensi mendukung terciptanya lingkungan ujian yang lebih aman dan terkontrol, meskipun evaluasi lebih lanjut terkait penerimaan pengguna dan perbandingan dengan kondisi tanpa aplikasi masih diperlukan untuk memastikan efektivitasnya secara menyeluruh dalam mengurangi kecurangan.

Berdasarkan hasil pelaksanaan dan pengembangan sistem, terdapat beberapa saran yang dapat diberikan untuk pengembangan aplikasi *Safe Exam Browser* ke depannya, yaitu sebagai berikut:

- a. Pengembangan Sistem *Monitoring*
Disarankan untuk menambahkan fitur *monitoring* secara *real-time* yang dapat digunakan oleh guru atau pengawas untuk melihat aktivitas siswa selama ujian berlangsung, seperti status perangkat, koneksi jaringan, dan tingkat pelanggaran.
- b. Peningkatan Keamanan Sistem
Pengembangan lebih lanjut dapat dilakukan dengan menambahkan mekanisme keamanan tambahan, seperti enkripsi data konfigurasi ujian serta sistem autentikasi yang lebih kuat untuk mencegah penyalahgunaan aplikasi.
- c. Pengembangan Fitur Pelaporan
Disarankan untuk menambahkan fitur pelaporan hasil ujian dan aktivitas siswa yang dapat diakses oleh guru, sehingga memudahkan dalam melakukan evaluasi terhadap pelaksanaan ujian.
- d. Pengembangan *Multi-Platform*
Aplikasi dapat dikembangkan lebih lanjut agar dapat berjalan pada *platform* lain, seperti *iOS* atau *desktop*, sehingga dapat digunakan oleh lebih banyak pengguna dengan berbagai jenis perangkat.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada SMAN 2 Sukawati yang telah memberikan kesempatan dan dukungan sebagai tempat pelaksanaan penelitian ini. Ucapan terima kasih juga penulis sampaikan kepada teman-teman seperjuangan yang selalu memberikan semangat, dukungan, dan kebersamaan selama proses penelitian berlangsung. Dukungan tersebut menjadi kekuatan bagi penulis untuk menyelesaikan penelitian ini dengan penuh rasa syukur.

DAFTAR PUSTAKA

- [1] B. Nia Rahmawati, I. Wayan Agus Arimbawa, and A. Zafrullah Mardiansyah, "Rancang Bangun Sistem Informasi Layanan Elektronik Program Studi Teknik Informatika Fakultas Teknik Universitas Mataram," *JTIKA*, [Online]. Available: <http://jtika.if.unram.ac.id/index.php/JTIKA/>
- [2] N. Luh, M. Murniati, I. Putu, A. Swastika, P. Anugrah, and C. Dewi, "Perancangan Sistem Informasi Sekolah SD Negeri 4 Kerta Payangan Menggunakan Metode Design Thinking," 2024. [Online]. Available: <https://journal.stmiki.ac.id>
- [3] A. Maulidya, "Implementasi Penggunaan Exam Browser di SMA Negeri 1 Seputih Banyak Lampung," *Wacana Publik*, vol. 19, no. 1, pp. 34–37, Jun. 2025, doi: 10.37295/wp.v19i1.69.
- [4] A. A. Istri et al., "Usability Testing *Learning Management System* (LMS) Berbasis Moodle di SMA Negeri 1 Sukawati," 2022.
- [5] N. Taj Athazaina, A. A. Istri, I. Paramitha, and N. Purnama, "Evaluasi *Learning Management System* Timedoor Academy Pro dengan Metode Usability Testing," *Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*, Dec. 2024.
- [6] A. Righo and S. B. Saragih, "Analisis Efektifitas CBT Exam Browser Berbasis Keamanan Digital (Anti AI) untuk Mencegah Kecurangan Ujian," *Journal of Innovative and Creativity*, vol. 5, no. 2, pp. 11362–11371, Aug. 2025."
- [7] M. W. P. Putra, A. A. I. I. Paramitha, and I. Gst. A. P. D. Putri, "Pengembangan Sistem Informasi Manajemen Biopori Berbasis Web Menggunakan Metodologi RAD untuk Meningkatkan Efisiensi Pengelolaan Sampah Organik," *Jurnal Algoritma*, vol. 22, no. 1, pp. 267–277, May 2025, doi: 10.33364/algoritma/v.22-1.2282.
- [8] I. S. Handayanto and I. Nuryasin, "Pengujian Blackbox Decision Table pada Sistem Aplikasi Mobile Sharing Story App," *Smart Comp: Jurnalnya Orang Pintar Komputer*, vol. 13, no. 2, Apr. 2024, doi: 10.30591/smartcomp.v13i2.6572.
- [9] I. Putu, F. Ananda, A. A. Istri, I. Paramitha, P. Anugrah, and C. Dewi, "Rancang Bangun Website Sistem Manajemen Peminjaman Sarana dan Prasarana pada Primakara University Menggunakan Metode Extreme Programming," 2025.

- [10] I. Gusti, N. Wiragunawan, S. Negeri, and K. Selatan, "Pemanfaatan *Learning Management System* (LMS) dalam Pengelolaan Pembelajaran Daring pada Satuan Pendidikan," *EDUTECH: Jurnal Inovasi Pendidikan Berbantuan Teknologi*, vol. 2, no. 1, 2022.
- [11] A. Firdaus and P. Theo Lonando, "Analisis dan Penerapan E-Learning Berbasis Android dengan Menggunakan Pendekatan Metode Prototype di IKesT Muhammadiyah Palembang," *Jurnal Digital Teknologi Informasi*, vol. 8, no. 2, p. 64, Sep. 2025, doi: 10.32502/digital.v8i2.10330.
- [12] K. Suryani, J. Bagindo Aziz Chan, A. Pacah, and P. -, "Safe Exam Browser untuk Klien Android pada Ujian Berbasis Web," *Edukasi: Jurnal Pendidikan*, Dec. 2019.
- [13] H. M. Mohammed and Q. I. Ali, "Cheating Prevention in E-proctoring Systems Using Secure Exam Browsers: A Case Study," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, vol. 8, no. 4, p. 634, Dec. 2022, doi: 10.26555/jiteki.v8i4.25094.
- [14] M. S. Vivaldy and O. Sihombing, "Aplikasi Pembelajaran Fisika dan Matematika untuk Tingkat SMP," *Jurnal Sistem Informasi Ilmu Komputer Prima (JUSIKOM PRIMA)*, vol. 2, no. 2, 2019.
- [15] A. Octaviano, S. Sofiana, and M. Rizky Arrahman, "Kohesi: Jurnal Multidisiplin Saintek <https://ejournal.cahayailmubangsa.institute/index.php/kohesi> Analisis Metode *Rapid Application Development* untuk Aplikasi Absensi di Android," 2025, doi: 10.8734/Kohesi.v1i2.365.
- [16] I. Riadi, A. Yudhana, and A. Elvina, "Analysis Impact of *Rapid Application Development* Method on Development Cycle and User Satisfaction: A Case Study on Web-Based Registration Service," *Scientific Journal of Informatics*, vol. 11, no. 1, pp. 81–94, Feb. 2024, doi: 10.15294/sji.v11i1.49590.