JURNAL ILMIAH SISTEM INFORMASI (JUSI)
VOL 4, No. 3, November 2025, pp. 724 - 740
p-ISSN : 2803-1507 (media cetak) | e-ISSN : 2803-1531 (media online)
Link Publikasi: https://ejurnal.provisi.ac.id/index.php/JUISI/issue/view/151

Development of the Andana Mobile Application: An Interactive Japanese Language Learning Platform Using Flutter and Firebase

Sovantri Putra Paskah Halawa*[1], Rizal Muslim Sinaga[2], Agata Putri Handayani Simbolon[3], Selfi Audy Priscilia[4], Adidtya Perdana[5]
[1,2,3,4,5] Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Medan
Jl. William Iskandar Ps. V, 20221, Medan, Sumatera Utara, Indonesia
e-mail: sovanhalawa06@mhs.unimed.ac.id*[1]; sinaga.rizal456@gmail.com[2];
agataputrisimbolon@gmail.com[3]; selfipriscilia05@gmail.com[4]; adidtya@unimed.ac.id[5]

ARTICLE INFO

Article History:

Received: 02-09-2025
Revised: 01-10-2025
Accepted: 23-10-2025
Published: 21-11-2025

ABSTRACT

The growing interest among Indonesian learners in the Japanese language has led to an increased demand for more interactive and flexible learning media. However, most existing applications remain limited to static material delivery without adaptive progress tracking. This study aims to develop Andana*, a mobile learning platform for Japanese language education built with Flutter and Firebase. The research adopts a Research and Development (R&D) method with a software engineering approach, encompassing requirement analysis, system design, implementation, and functional testing. Data were collected from functional trials and user-based performance evaluations.* Andana *supports two leading roles—user and admin—and integrates multimedia-based learning features (PDF, video, audio), flashcards, listening and speaking exercises, and personalized note-taking. The results indicate that all system functionalities operate effectively, with real-time data synchronization and a responsive user interface. The integration of Flutter and Firebase enables dynamic data management and seamless content updates. This study contributes to the development of adaptive, cloud-based learning applications that are easy to maintain and expand. The findings imply that* Andana *can serve as an efficient and engaging medium for Japanese language learning, with strong potential for adaptation to other foreign languages in the future.*

Keywords: *Flutter, Firebase, Japanese Language Learning, Mobile Application, Mobile Learning.*

Abstrak

Peningkatan minat masyarakat Indonesia terhadap bahasa Jepang mendorong kebutuhan akan media pembelajaran yang lebih interaktif dan fleksibel. Namun, sebagian besar aplikasi yang ada masih terbatas pada penyajian materi secara statis tanpa sistem pelacakan progres belajar yang adaptif. Penelitian ini bertujuan untuk mengembangkan *Andana*, sebuah aplikasi *mobile learning* bahasa Jepang interaktif berbasis Flutter dan Firebase. Metode penelitian yang digunakan adalah *Research and Development* (R&D) dengan pendekatan *software engineering*, yang mencakup analisis kebutuhan, perancangan sistem, implementasi, dan pengujian fungsional. Data penelitian diperoleh dari hasil uji coba fungsionalitas dan evaluasi performa

DOI: https://doi.org/10.51903/0m972c26

p-ISSN : 2803-1507 e-ISSN : 2803-1531

aplikasi oleh pengguna. Aplikasi *Andana* dirancang dengan dua peran utama, yaitu pengguna dan admin, serta dilengkapi fitur pembelajaran berbasis multimedia (PDF, video, dan audio), *flashcard*, latihan mendengarkan, berbicara, serta sistem catatan pribadi. Hasil penelitian menunjukkan bahwa seluruh fungsi aplikasi berjalan dengan baik, dengan sinkronisasi data secara *real-time* dan antarmuka pengguna yang responsif. Integrasi antara Flutter dan Firebase memungkinkan pengelolaan data pengguna serta pembaruan konten secara dinamis. Kontribusi penelitian ini terletak pada pengembangan model aplikasi pembelajaran berbasis *cloud* yang adaptif dan mudah dikembangkan. Implikasinya, *Andana* berpotensi menjadi media pembelajaran bahasa Jepang yang efisien, menarik, dan dapat diadaptasi untuk bahasa asing lainnya di masa mendatang.

Kata Kunci: *Flutter, Firebase, Pembelajaran Bahasa Jepang, Aplikasi Mobile, Mobile Learning*.

1. PENDAHULUAN

Pembelajaran bahasa asing semakin strategis di era global. Di Indonesia, minat terhadap bahasa Jepang meningkat di institusi pendidikan maupun pembelajaran mandiri melalui platform digital. Kenaikan ini didorong oleh faktor ekonomi–budaya, termasuk kolaborasi pendidikan dan industri Indonesia–Jepang, serta pengaruh budaya populer seperti anime, manga, dan musik. Di sisi lain, karakteristik bahasa Jepang dengan tiga sistem tulisan—Hiragana, Katakana, dan Kanji—menuntut latihan berulang dan dukungan media belajar yang adaptif.

Paradigma *mobile learning* (ML) menawarkan fleksibilitas waktu, tempat, dan akses, serta berkontribusi terhadap peningkatan keterlibatan belajar [1]. Dalam konteks bahasa Jepang, efektivitas ML sangat dipengaruhi desain yang interaktif dan berpusat pada pengguna [2]. Namun, banyak aplikasi yang ada masih menyajikan materi secara statis tanpa *progress tracking* real-time atau *adaptive learning* yang memandu lintasan belajar pengguna [3]. Kondisi ini menyulitkan pemantauan kemajuan dan berpotensi menurunkan motivasi.

Penelitian terdahulu telah mengeksplorasi *flashcard* dan *spaced repetition system* (SRS) serta pendekatan gamifikasi untuk meningkatkan retensi [4], dan menekankan pentingnya *ubiquitous learning environment* agar pembelajaran berlangsung lintas konteks [5]. Tinjauan lokal juga menunjukkan pertumbuhan aplikasi ML dalam pendidikan bahasa [6]. Meski demikian, banyak studi masih konseptual atau belum mengintegrasikan *cloud backend* modern (misalnya Firebase) untuk pengelolaan data yang aman, sinkron, dan dinamis. Integrasi teknologi *backend* yang canggih dapat memberikan dukungan terhadap analisis perilaku pengguna dan personalisasi materi secara otomatis.

Adopsi Flutter sebagai kerangka lintas platform untuk aplikasi pembelajaran bahasa juga relatif terbatas, padahal kombinasi Flutter–Firebase memungkinkan sinkronisasi real-time, autentikasi terpadu, dan manajemen konten dinamis pada arsitektur *cloud-based* [7]. Kesenjangan riset yang kami identifikasi adalah minimnya implementasi empiris yang menggabungkan Flutter–Firebase untuk pembelajaran bahasa Jepang dengan dukungan multimedia (audio, video, teks), evaluasi kemajuan adaptif berbasis SRS, serta *progress tracking* yang terikat profil pengguna. Pendekatan lintas platform seperti *Flutter* dapat mempercepat proses pengembangan karena kode yang sama dapat dijalankan di berbagai sistem operasi. Selain itu, pemanfaatan integrasi *Firebase* dapat memfasilitasi pembaruan data dan pemantauan performa pengguna secara efisien.

Tujuan studi ini adalah mengembangkan *Andana*, platform pembelajaran bahasa Jepang interaktif berbasis Flutter–Firebase. *Andana* menyediakan materi multimedia, *flashcard*, latihan mendengar dan berbicara, serta sistem catatan dan pelacakan progres real-time. Kontribusi kebaruan kami meliputi: (i) arsitektur *cloud-based* modular dengan *role-based access control* (RBAC), (ii) pembaruan konten *over-the-air* tanpa rilis aplikasi, dan (iii) integrasi prinsip SRS ke dalam kuis adaptif yang memanfaatkan histori performa pengguna. Studi ini menggunakan metode *Research and Development (R&D)* dengan pendekatan *software engineering* agar hasil dapat diimplementasikan secara praktis dan terukur.

2. TINJAUAN PUSTAKA

Penelitian mengenai penerapan *mobile learning* (ML) menunjukkan perkembangan pesat dalam dekade terakhir karena kemampuannya menghadirkan pembelajaran yang fleksibel, mandiri, dan kontekstual di berbagai bidang. Studi [8] mengembangkan *Mobile Academy*, sebuah platform *ubiquitous learning* yang memungkinkan akses belajar lintas waktu dan tempat, menunjukkan bahwa perangkat bergerak dapat menjadi sarana efektif dalam mendukung keterlibatan belajar berkelanjutan. Dalam konteks pembelajaran

p-ISSN : 2803-1507 e-ISSN : 2803-1531

bahasa, [9] meneliti penerapan *mobile assisted language learning* dan menemukan bahwa penggunaan aplikasi interaktif mampu meningkatkan retensi kosakata dan partisipasi siswa. [10] juga mengembangkan sistem pembelajaran Kanji adaptif berbasis *mobile phone* yang menyesuaikan kesulitan materi berdasarkan performa pengguna. Hasil-hasil tersebut memperkuat urgensi pengembangan sistem ML yang tidak hanya menyediakan konten, tetapi juga mendukung personalisasi dan adaptivitas bagi pembelajar.

Penelitian terkait pembelajaran bahasa Jepang juga memperlihatkan meningkatnya minat pengguna terhadap media digital yang interaktif dan kontekstual. [11] merancang antarmuka *mobile gamified* untuk pembelajaran bahasa Jepang melalui pendekatan *design thinking* dan menemukan bahwa penggunaan elemen interaktif mampu meningkatkan motivasi belajar. [12] meneliti perilaku mahasiswa di Chengdu, Tiongkok, yang menunjukkan kecenderungan tinggi terhadap penggunaan aplikasi bahasa Jepang karena kemudahan akses dan antarmuka menarik. Penelitian [13] juga mengembangkan sistem pembelajaran bahasa Jepang berbasis aplikasi dan menekankan pentingnya adaptasi konten terhadap karakteristik pengguna. Temuan-temuan ini konsisten dengan hasil studi [14] yang menyoroti peluang dan tantangan *mobile learning* di kelas EFL Indonesia, terutama dalam aspek kesiapan teknologi pengajar dan siswa.

Pendekatan *gamifikasi* menjadi salah satu inovasi penting dalam pengembangan media pembelajaran modern. [15] menunjukkan bahwa penerapan elemen permainan dalam aplikasi pembelajaran bahasa Jepang mampu meningkatkan konsistensi pengguna dalam berlatih kosakata. [16] mendemonstrasikan bahwa poin, lencana, dan level dapat memperkuat komitmen belajar dengan memberikan pengalaman menyenangkan dan interaktif. [17] menambahkan dimensi kognitif melalui penerapan *spaced repetition scheduling* berbasis kecerdasan buatan, yang terbukti meningkatkan daya ingat kosakata jangka panjang dibandingkan metode tradisional. Dalam konteks serupa, [18] menekankan perlunya lingkungan belajar yang *ubiquitous* dan adaptif untuk meningkatkan efektivitas *mobile learning*. Pendekatan kombinatif antara gamifikasi dan SRS ini terbukti menjadi landasan kuat bagi desain sistem pembelajaran berbasis performa pengguna seperti yang diusung dalam aplikasi Andana.

Dari sisi teknologi, sejumlah penelitian berfokus pada efisiensi dan stabilitas dalam pengembangan aplikasi pembelajaran berbasis *mobile*. [19] menekankan pentingnya efisiensi logika dan struktur kode dalam memastikan performa aplikasi. Flutter, yang menggunakan bahasa Dart, dinilai unggul karena kemampuannya menghasilkan performa tinggi, *hot reload*, serta efisiensi memori dalam pengembangan aplikasi lintas platform. Penelitian [20] menunjukkan bahwa kombinasi Flutter dan Firebase sangat ideal untuk pengembangan aplikasi dengan kebutuhan *real-time synchronization* karena memungkinkan pembaruan konten secara langsung tanpa pembaruan aplikasi. Sementara itu, penelitian [21] menyoroti pemanfaatan pembelajaran mesin untuk deteksi dini pencurian data pada perangkat *mobile*, menegaskan relevansi penggunaan teknologi *real-time* dalam meningkatkan keamanan sistem berbasis *mobile learning*.

Dari sisi *backend*, beberapa penelitian membahas penerapan Firebase sebagai *Backend-as-a-Service (BaaS)* yang efisien untuk mendukung arsitektur aplikasi modern. Google menjelaskan bahwa Firebase menyediakan autentikasi pengguna, penyimpanan data *cloud*, serta layanan basis data *real-time* yang dapat diintegrasikan dengan mudah oleh pengembang. [22] membuktikan bahwa integrasi layanan *commercial cloud* meningkatkan kecepatan sinkronisasi data dan keamanan sistem, sedangkan [23] menegaskan efisiensi Firebase untuk pengembangan aplikasi *mobile* dan *web* edukatif. Studi [24] menyoroti pentingnya keamanan privasi pengguna dalam ekosistem pendidikan digital, yang dapat ditunjang oleh arsitektur Firebase dengan perlindungan data berbasis *cloud*. Selanjutnya, penelitian [25] menjelaskan transisi arsitektur basis data dari *on-premise* menuju *cloud computing* yang lebih efisien dan skalabel, mendukung tren global pengembangan sistem pembelajaran berbasis daring. Berdasarkan temuan tersebut, integrasi Flutter dan Firebase dalam pengembangan aplikasi Andana dapat dipandang sebagai langkah strategis untuk menciptakan platform pembelajaran bahasa Jepang yang adaptif, aman, dan berorientasi pengguna.

3. METODOLOGI PENELITIAN

3.1. *Jenis Penelitian*

Penelitian ini merupakan penelitian *Research and Development (R&D)* yang berfokus pada proses pengembangan aplikasi pembelajaran bahasa Jepang berbasis *mobile* bernama Andana. Tujuan penelitian ini adalah menghasilkan aplikasi edukatif yang berfungsi dengan baik sebagai media pembelajaran interaktif tanpa menekankan pada efektivitas hasil belajar pengguna. Fokus utama penelitian ini adalah pada proses perancangan, implementasi, dan integrasi sistem aplikasi. Model *R&D* dipilih karena memungkinkan

p-ISSN : 2803-1507 e-ISSN : 2803-1531

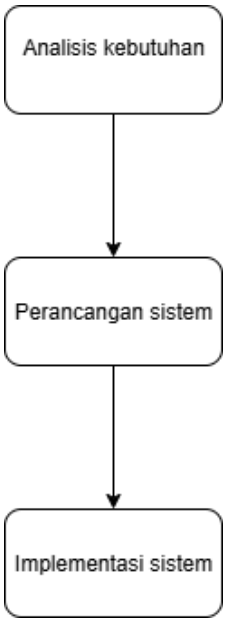
pengembangan produk secara sistematis melalui tahapan analisis kebutuhan, perancangan, implementasi, serta evaluasi awal terhadap hasil rancangan.

3.2. *Pendekatan Penelitian*

Pendekatan yang digunakan adalah pendekatan rekayasa perangkat lunak (*software engineering approach*). Metode ini menekankan proses pengembangan perangkat lunak yang terstruktur, efisien, serta dapat dikembangkan lebih lanjut. Aplikasi dibangun menggunakan *framework Flutter* dengan bahasa pemrograman Dart serta layanan *Firebase* sebagai *backend* untuk autentikasi dan penyimpanan data berbasis *cloud*. Seluruh pengembangan dilakukan secara iteratif dan fleksibel menyesuaikan kebutuhan fitur dan peningkatan pengalaman pengguna (*user experience*).

3.3. *Tahapan Penelitian*

Tahapan penelitian ini dilakukan untuk memastikan proses pengembangan sistem berjalan secara terstruktur dan efisien. Setiap tahapan dirancang agar hasil penelitian dapat terukur serta mudah dievaluasi sesuai dengan tujuan yang telah ditetapkan. Alur penelitian ini menggambarkan urutan kegiatan mulai dari analisis kebutuhan hingga tahap pengujian sistem. Gambaran umum mengenai alur tahapan penelitian secara sistematis dapat dilihat pada Gambar 1.



Gambar 1. Alur Diagram Penelitian yang digunakan

Alur tahapan penelitian secara sistematis ditunjukkan pada Gambar 1. Tahapan tersebut meliputi analisis kebutuhan, perancangan sistem, implementasi, dan pengujian. Analisis kebutuhan dilakukan untuk mengidentifikasi kebutuhan fungsional dan non-fungsional sistem. Tahap perancangan mencakup desain alur sistem, struktur *database*, dan antarmuka aplikasi. Implementasi merupakan proses pengkodean berdasarkan desain menggunakan *Flutter* dan *Firebase*. Proses pengujian kemudian dilakukan untuk memastikan seluruh komponen sistem bekerja sesuai fungsi yang diharapkan.

3.3.1. *Analisis Kebutuhan*

Tahap ini bertujuan untuk mengidentifikasi kebutuhan pengguna serta menentukan fitur utama aplikasi. Analisis kebutuhan dilakukan agar fungsi aplikasi sesuai dengan karakteristik pengguna yang menjadi target utama. Informasi mengenai aktor diperoleh dari studi awal terhadap perilaku dan peran pengguna dalam konteks pembelajaran bahasa Jepang. Aplikasi Andana memiliki dua jenis aktor utama:

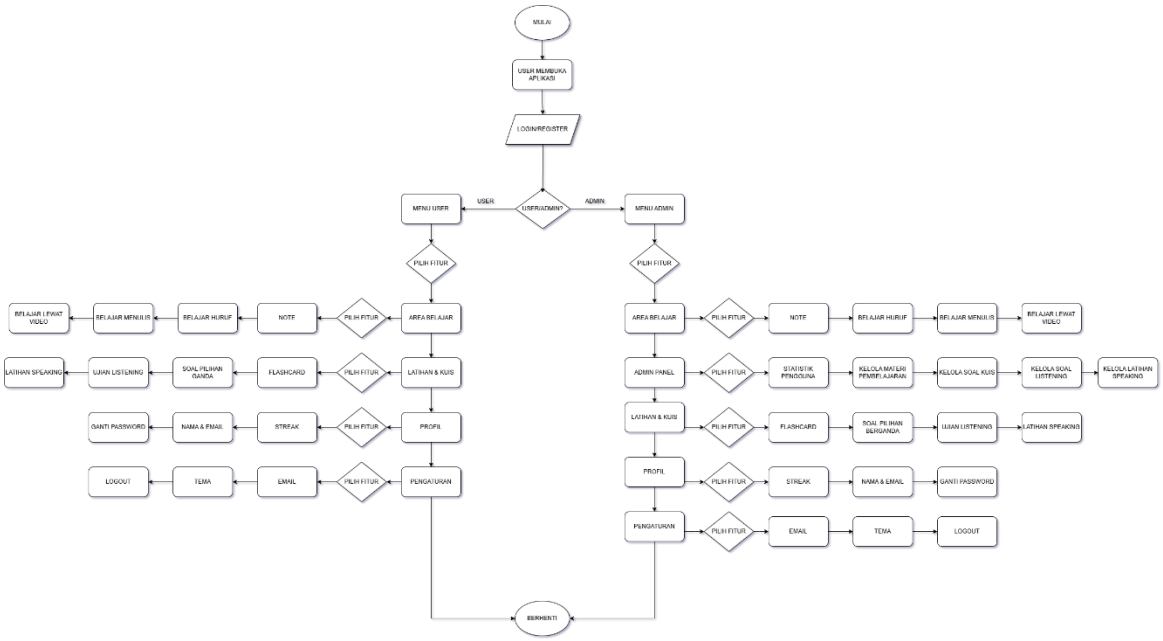
1. Pengguna (pembelajar): belajar bahasa Jepang secara mandiri melalui materi, latihan, dan kuis.
2. Admin: mengelola konten pembelajaran, soal latihan, dan memantau statistik pengguna.

p-ISSN : 2803-1507 e-ISSN : 2803-1531

Fitur utama yang dikembangkan meliputi autentikasi pengguna, pengaturan tema, pembelajaran berbasis media (PDF, video, audio), catatan pribadi, latihan interaktif (*listening*, *speaking*, dan *quiz*), serta panel admin untuk pengelolaan konten secara dinamis. Struktur data dan konten dirancang agar dapat diperluas hingga level JLPT N1, meskipun dalam penelitian ini hanya digunakan *demo content* untuk uji coba fungsionalitas sistem. Setiap fitur tersebut dirancang agar saling terintegrasi dan mendukung proses belajar secara mandiri. Implementasi fitur juga mempertimbangkan kemudahan navigasi serta efisiensi penggunaan sumber daya sistem agar aplikasi tetap ringan dijalankan.

3.3.2. *Perancangan Sistem*

Tahap perancangan sistem dilakukan setelah proses analisis kebutuhan selesai dilaksanakan. Pada tahap ini, peneliti menyusun rancangan alur interaksi antara pengguna dan sistem untuk menggambarkan fungsi utama yang akan diimplementasikan. Perancangan ini bertujuan agar setiap fitur dalam aplikasi dapat terintegrasi dengan baik sesuai peran masing-masing pengguna. Diagram rancangan interaksi sistem dan pengguna digambarkan dalam bentuk *use case diagram* seperti ditunjukkan pada Gambar 2.



Gambar 2. Use Case Diagram

Diagram *use case* aplikasi ditunjukkan pada Gambar 2. Alur aplikasi diawali dengan pengguna membuka aplikasi dan melakukan *login* atau *registrasi*. Tahapan ini menggambarkan mekanisme awal sistem dalam membedakan jenis pengguna sebelum mengarahkan mereka ke antarmuka yang sesuai. Sistem kemudian memeriksa peran pengguna (*user* atau *admin*). Adapun tahapan alur aplikasi tersebut adalah sebagai berikut:

1. Jika pengguna adalah *user*, sistem menampilkan menu utama yang berisi fitur belajar huruf, menulis, video pembelajaran, catatan, *flashcard*, kuis, *listening*, *speaking*, profil, dan pengaturan.
2. Jika pengguna adalah *admin*, sistem akan menampilkan akses ke *admin panel* dengan fitur tambahan untuk pengelolaan materi pembelajaran, soal, latihan *speaking*, ujian *listening*, dan statistik pengguna.

Pengembangan Andana menggunakan pendekatan *user-centered design* agar mudah digunakan oleh pengguna umum dan efisien bagi admin. Pendekatan ini menekankan keterlibatan pengguna dalam setiap tahapan desain agar produk akhir sesuai kebutuhan aktual. Pembagian peran membantu peneliti menentukan batas tanggung jawab dan fitur yang relevan bagi masing-masing pengguna. *Use case* aplikasi dibagi menjadi dua aktor berikut:

Aktor 1: Pengguna (Pembelajar)

1. Autentikasi dan pengaturan akun.
2. Mengakses materi pembelajaran dalam bentuk PDF dan video.

p-ISSN : 2803-1507 e-ISSN : 2803-1531

3. Membuat dan mengelola catatan pribadi (*NotesScreen*) yang dapat digunakan kembali sebagai *flashcard*.
4. Mengerjakan kuis, latihan *listening*, dan latihan *speaking* berbasis *speech-to-text*.

Aktor 2: Admin

1. Mengakses *AdminPanelScreen* untuk mengelola konten.
2. Menambah, mengubah, dan menghapus materi (PDF/video) dan soal latihan.
3. Melihat statistik jumlah pengguna melalui *UserStatsScreen*.

3.3.3. *Implementasi Sistem*

Implementasi dilakukan berdasarkan rancangan yang telah dibuat. Proses implementasi ini melibatkan penerapan komponen perangkat lunak yang telah dirancang pada tahap sebelumnya. Setiap teknologi dipilih untuk mendukung fungsi aplikasi agar dapat berjalan secara optimal di berbagai perangkat. Teknologi yang digunakan adalah sebagai berikut:

1. Flutter dan Dart digunakan untuk membangun antarmuka dan logika aplikasi secara modular.
2. Firebase (firebase_core, firebase_auth, cloud_firestore) digunakan untuk autentikasi pengguna, penyimpanan data dinamis, dan sinkronisasi *real-time*.
3. Google Drive digunakan sebagai penyimpanan file media (PDF, video, audio) yang diakses menggunakan tautan (URL).
4. google_fonts (Poppins, Noto Sans JP) digunakan agar tampilan teks dan karakter Jepang konsisten dan mudah dibaca.
5. flip_card digunakan untuk fitur *flashcard* (tampilan dua sisi, kata ↔ arti).
6. flutter_pdfview, video_player, chewie, audioplayers digunakan untuk menampilkan dan memutar media pembelajaran.
7. webview_flutter digunakan untuk fitur *speaking practice* berbasis Web Speech API.
8. shared_preferences digunakan untuk menyimpan preferensi tema aplikasi secara lokal.
9. permission_handler digunakan untuk mengelola izin perangkat seperti mikrofon.
10. Role-Based Access Control (RBAC) digunakan untuk membedakan hak akses antara admin dan pengguna berdasarkan data di Firestore.

Struktur arsitektur aplikasi bersifat modular dengan pendekatan *feature-first* yang memisahkan antarmuka, logika bisnis, dan pengelolaan data sehingga memudahkan pengembangan lanjutan dan pemeliharaan sistem. Pendekatan ini memungkinkan setiap fitur dikembangkan secara independen tanpa memengaruhi fitur lain. Pemisahan struktur kode juga membantu pengembang dalam melakukan pembaruan dan perbaikan secara efisien. Dengan struktur seperti ini, proses pengujian dan integrasi dapat dilakukan secara lebih terarah.

4. HASIL DAN PEMBAHASAN

4.1. *Hasil Pengembangan Aplikasi Andana*

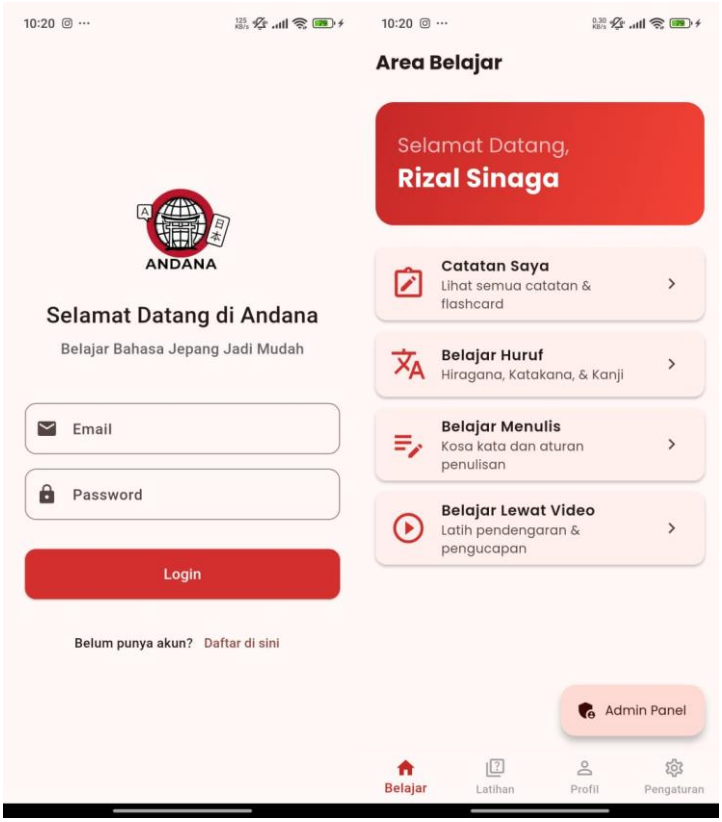
Penelitian ini berhasil mewujudkan sebuah produk perangkat lunak berupa aplikasi mobile bernama Andana, yang berfungsi sebagai media pembelajaran mandiri bahasa Jepang. Aplikasi ini dikembangkan menggunakan *framework* Flutter dengan dukungan penuh dari layanan *backend* Firebase, yang dirancang untuk menjadi platform yang interaktif, fleksibel, dan dinamis. Sifat interaktifnya diwujudkan melalui implementasi *state management* pada sisi klien untuk memberikan umpan balik *real-time* saat pengguna mengerjakan berbagai jenis latihan, mulai dari kuis pilihan ganda, ujian *listening* berbasis audio, hingga latihan *speaking* dengan validasi suara. Antarmuka aplikasi ini dirancang dengan pendekatan *user-centered design* agar setiap elemen tampilan dapat mendukung proses belajar yang intuitif dan efisien.

Fleksibilitas sistem tercermin dari arsitektur yang berpusat pada pengguna, di mana alur belajar tidak bersifat linear. Pengguna dapat memilih materi secara acak sesuai kebutuhan dan membangun korpus pengetahuan mereka sendiri melalui fitur catatan yang terikat pada *UserID* unik mereka. Aspek dinamis menjadi keunggulan arsitektur utama, di mana pemisahan antara lapisan presentasi (*frontend*) dan lapisan data (*backend*) memungkinkan pembaruan konten secara *over-the-air*, artinya materi dan soal dapat diperbarui oleh admin kapan saja tanpa mengharuskan pengguna mengunduh versi baru aplikasi. Pendekatan ini memastikan pengalaman belajar tetap relevan dan terbarui tanpa mengganggu aktivitas pengguna yang sedang berlangsung.

p-ISSN : 2803-1507 e-ISSN : 2803-1531

Sistem Andana dirancang untuk mengakomodasi dua jenis pengguna (aktor) dengan peran yang berbeda: Pengguna (pembelajar) sebagai konsumen utama konten yang berfokus pada kegiatan belajar, dan Admin sebagai manajer konten yang bertugas mengelola siklus hidup data pembelajaran, mulai dari pembuatan materi, pengelolaan soal, hingga pemantauan statistik dasar pengguna. Perbedaan peran ini memungkinkan sistem berjalan secara seimbang antara sisi konsumsi dan produksi konten. Admin memiliki akses ke dasbor analitik yang menampilkan data performa pengguna untuk membantu dalam pengembangan materi lanjutan. Sementara itu, pengguna dapat berinteraksi dengan sistem melalui berbagai mode latihan yang disesuaikan dengan tingkat kemampuan mereka.

Aplikasi Andana telah berhasil dikembangkan sebagai platform *mobile* pembelajaran bahasa Jepang yang interaktif dan mudah diakses oleh pengguna. Hasil pengembangan menunjukkan bahwa antarmuka aplikasi telah berfungsi dengan baik dan menampilkan desain yang sederhana namun informatif. Tampilan utama aplikasi terdiri dari halaman *login* serta *dashboard* yang berperan sebagai pusat navigasi pengguna. Hasil rancangan tampilan antarmuka awal aplikasi ditunjukkan pada Gambar 3.



Gambar 3. Tampilan Login dan Dashboard

Tampilan antarmuka pengguna (*UI*) halaman login dan dashboard ditunjukkan pada Gambar 3, yang menggambarkan rancangan Andana sebagai *platform mobile* untuk pembelajaran bahasa Jepang interaktif. Tampilan ini dirancang dengan mempertimbangkan kemudahan akses bagi pengguna pemula maupun yang sudah terbiasa menggunakan aplikasi pembelajaran. Elemen visual pada halaman login menonjolkan kesederhanaan dan keterbacaan agar proses autentikasi berlangsung tanpa kendala. Selain itu, tata letak dashboard disusun agar navigasi ke fitur utama dapat dilakukan secara intuitif. Adapun penjelasan mengenai fungsi masing-masing bagian antarmuka adalah sebagai berikut:

1. Layar kiri (halaman *login*): merupakan gerbang utama aplikasi untuk autentikasi. Proses login dan registrasi dikelola oleh *Firebase Authentication* yang menangani akun pengguna berbasis email dan kata sandi.
2. Layar kanan (area belajar/*HomeScreen*): merupakan halaman utama setelah pengguna berhasil login dan berfungsi sebagai pusat navigasi. Halaman ini menampilkan fitur-fitur utama yang dapat diakses pengguna, seperti:

p-ISSN : 2803-1507 e-ISSN : 2803-1531

3. Catatan Saya: membuat dan mengelola catatan pribadi yang juga dapat digunakan sebagai *flashcard* untuk pengulangan materi.
4. Belajar Huruf, Menulis, dan Video: modul-modul pembelajaran inti yang menyediakan materi untuk belajar huruf (Hiragana, Katakana, Kanji), latihan menulis, serta latihan mendengarkan dan pengucapan melalui media video.
5. *Admin Panel*: hanya dapat diakses oleh pengguna dengan peran admin dan merupakan wujud implementasi *Role-Based Access Control (RBAC)* untuk mengelola materi pembelajaran, soal kuis, serta memantau statistik pengguna.

Sebagai penegasan kebaruan, Andana mengimplementasikan arsitektur cloud-based modular dengan pemisahan lapisan presentasi, logika, dan data yang memungkinkan pembaruan konten over the air tanpa rilis aplikasi. Skema role-based access control memastikan diferensiasi hak akses antara pengguna dan admin pada tingkat data. Prinsip spaced repetition diintegrasikan ke dalam kuis adaptif yang memanfaatkan riwayat performa untuk memprioritaskan materi yang belum dikuasai. Kombinasi ketiga aspek ini menghadirkan platform belajar bahasa yang dinamis, terukur, dan berpusat pada pengguna.

4.2. *Desain dan Implementasi Sistem*

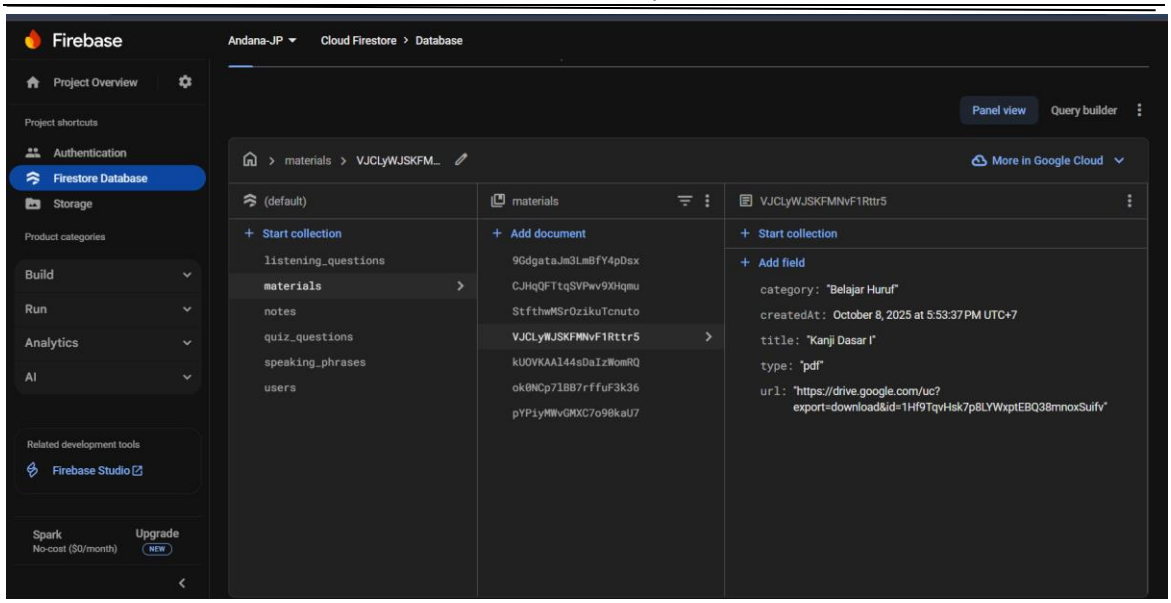
Aplikasi *Andana* mengadopsi arsitektur *client–server* modern. Sisi klien, yang bertanggung jawab atas *rendering* antarmuka (*UI*) dan manajemen *state*, dibangun sepenuhnya menggunakan *Flutter* dan bahasa Dart. Pendekatan ini memungkinkan kompilasi ke kode mesin asli (*native*) untuk platform Android dan iOS dari satu basis kode. Sisi server memanfaatkan platform *Backend-as-a-Service (BaaS)* dari Google *Firebase*, yang menyediakan lapisan persistensi data melalui *Cloud Firestore*, lapisan autentikasi melalui *Firebase Authentication*, dan layanan pendukung lainnya.

Struktur *database* pada *Cloud Firestore* dirancang secara modular dengan pendekatan *NoSQL* untuk mendukung skalabilitas dan fleksibilitas. Desain modular ini bertujuan agar struktur penyimpanan dapat menyesuaikan kebutuhan aplikasi di masa mendatang. Setiap koleksi dirancang agar mampu menyimpan data dengan format yang konsisten sekaligus fleksibel terhadap perubahan skema. Pendekatan ini membantu menjaga efisiensi dalam proses pengambilan maupun pembaruan data di *Cloud Firestore.* Koleksi utama yang diimplementasikan meliputi:

1. *users*: menjadi fondasi personalisasi aplikasi. Koleksi ini tidak hanya menyimpan data autentikasi tetapi juga *role* (user atau admin) untuk implementasi *RBAC*, serta data progres seperti *streak* belajar harian.
2. *materials*: dirancang dengan denormalisasi parsial, di mana metadata penting seperti *category* dan *type* disimpan langsung di setiap dokumen. Pendekatan ini memungkinkan *query* pemfilteran pada sisi klien berjalan efisien tanpa memerlukan operasi *join* yang kompleks.
3. *notes*: menyimpan catatan pribadi pengguna, di mana setiap dokumen terikat pada *userId* pemiliknya. Struktur ini menjamin privasi data dan memungkinkan *query* yang cepat untuk mengambil catatan spesifik pengguna.
4. *quiz_questions, listening_questions, speaking_phrases*: konten ujian dipisahkan ke dalam koleksi masing-masing untuk mempermudah manajemen dan pemfilteran soal berdasarkan kategori, level, atau topik.

Hasil implementasi sistem pada aplikasi Andana menunjukkan bahwa komponen penyimpanan data telah berfungsi dengan baik menggunakan layanan *Cloud Firestore*. Sistem ini dirancang agar dapat mengelola data pengguna, materi, dan hasil belajar secara terintegrasi. Struktur penyimpanan dibuat agar proses pembacaan dan penulisan data berlangsung cepat serta aman digunakan di lingkungan *cloud*. Gambaran hasil implementasi struktur *database* dapat dilihat pada Gambar 4.

p-ISSN : 2803-1507 e-ISSN : 2803-1531



Gambar 4. Struktur Database Cloud Firestore Aplikasi Andana

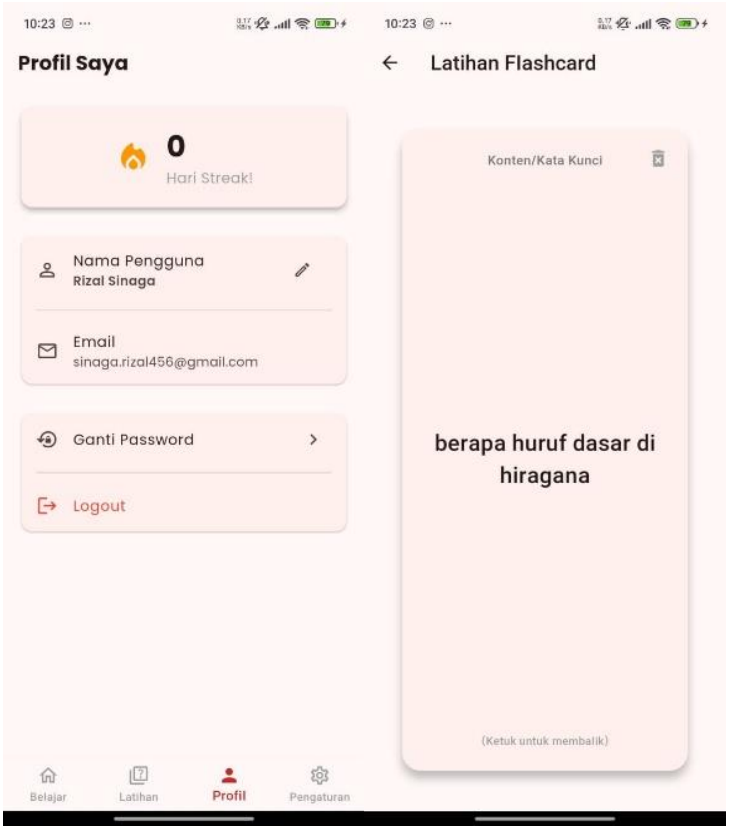
Struktur *backend* aplikasi Andana menggunakan *Cloud Firestore* ditunjukkan pada Gambar 4. Struktur *database* ini dirancang secara modular untuk mendukung fleksibilitas dan skalabilitas aplikasi. Desain ini memungkinkan hubungan antar-koleksi tetap efisien meskipun jumlah data bertambah seiring waktu. Pendekatan tersebut juga memudahkan proses integrasi dengan layanan lain di ekosistem *Firebase*. Koleksi yang diimplementasikan meliputi:

1. *users*: menyimpan data autentikasi dan profil pengguna, serta informasi peran (user atau admin) untuk menerapkan *RBAC*.
2. *materials*: berisi metadata materi pembelajaran. Setiap dokumen materi memiliki *field* seperti *category*, *title*, dan *type* (misalnya PDF). File media disimpan di Google Drive dan diakses melalui tautan (*URL*) yang tersimpan pada dokumen.
3. *notes*: menyimpan catatan pribadi yang dibuat oleh setiap pengguna dan terikat pada *userId* pemiliknya untuk menjaga privasi data.
4. *quiz_questions*, listening_questions, speaking_phrases: konten berbagai jenis latihan dan ujian dipisahkan dalam koleksi masing-masing agar admin mudah mengelola soal.

Pendekatan ini memungkinkan admin memperbarui materi dan soal latihan secara dinamis langsung dari *Firebase,* dan perubahan tersebut akan langsung tampil di aplikasi pengguna tanpa perlu melakukan pembaruan aplikasi. Mekanisme ini memanfaatkan kemampuan sinkronisasi *real-time* yang dimiliki oleh *Firebase Cloud Firestore.* Setiap perubahan pada koleksi akan segera terdistribusi ke seluruh pengguna aktif tanpa jeda yang berarti. Hal ini membantu memastikan konten pembelajaran selalu terkini dan relevan bagi seluruh pengguna.

Hasil implementasi aplikasi Andana juga mencakup pengembangan *fitur* pembelajaran yang dirancang untuk meningkatkan interaksi serta kenyamanan pengguna. Fitur-fitur ini memungkinkan pengguna menyesuaikan pengalaman belajar sesuai kebutuhan dan minatnya. Elemen personalisasi dikombinasikan dengan sistem pembelajaran berbasis aktivitas agar pengguna dapat belajar secara mandiri maupun terarah. Hasil tampilan *fitur* personalisasi dan pembelajaran *interaktif* ditunjukkan pada Gambar 5.

p-ISSN : 2803-1507 e-ISSN : 2803-1531



Gambar 5. Fitur Personalisasi dan Pembelajaran Interaktif Aplikasi Andana

Gambar 5 menampilkan berbagai *fitur* yang berfokus pada personalisasi pengguna dan alat pembelajaran *interaktif* untuk menciptakan pengalaman belajar yang utuh. Pada tampilan ini, pengguna dapat menyesuaikan preferensi materi, tingkat kesulitan, serta gaya belajar sesuai kebutuhannya. Sistem juga menyediakan elemen visual dan kuis interaktif untuk mendukung keterlibatan aktif dalam proses pembelajaran. Implementasi *fitur* ini menunjukkan bahwa aplikasi mampu memberikan pengalaman belajar yang adaptif dan responsif terhadap karakteristik pengguna.

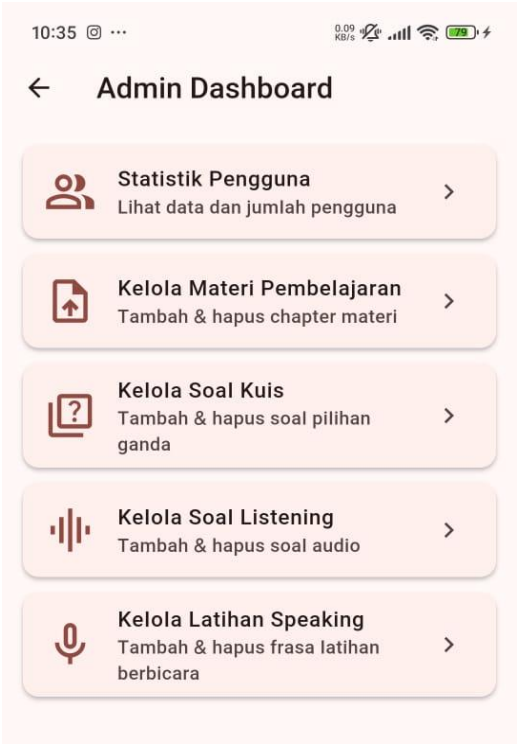
4.3. *Manajemen Akun dan Personalisasi*

Manajemen akun dan personalisasi pada aplikasi ini berfungsi untuk memberikan pengalaman belajar yang lebih personal dan interaktif bagi pengguna. Melalui fitur ini, pengguna dapat mengatur identitas pribadi, menyesuaikan tampilan aplikasi sesuai preferensi, serta mendukung proses belajar mandiri secara berkelanjutan. Selain itu, sistem juga dilengkapi elemen *gamifikasi* guna meningkatkan motivasi pengguna dalam mempertahankan konsistensi belajar setiap hari. Secara umum, manajemen akun dan personalisasi dalam aplikasi ini mencakup beberapa fitur utama, yaitu sebagai berikut:

1. Pendaftaran dan Profil: pengguna dapat membuat akun baru melalui halaman registrasi. Setelah masuk, halaman profil menampilkan data personal seperti nama dan email, serta elemen *gamification* berupa *streak* harian untuk memotivasi konsistensi belajar.
2. Pengaturan Aplikasi: halaman pengaturan memungkinkan pengguna menyesuaikan preferensi aplikasi, seperti mengubah tema terang atau gelap. Pilihan tema disimpan secara lokal menggunakan shared_preferences agar konsisten antarsesi.
3. Siklus Belajar Mandiri: Catatan dan *Flashcard*
4. Catatan Pribadi: pengguna membuat catatan pribadi saat mempelajari materi. Fitur ini membantu membangun basis pengetahuan personal.
5. Latihan *Flashcard*: catatan yang telah dibuat dapat dimanfaatkan kembali sebagai *flashcard* menggunakan paket flip_card untuk animasi kartu dua sisi (kata ↔ arti). Siklus ini mendukung alur belajar mandiri yang efektif: Mempelajari → Mencatat → Mengulang.

p-ISSN : 2803-1507 e-ISSN : 2803-1531

Hasil implementasi sistem menunjukkan bahwa aplikasi Andana tidak hanya berfokus pada pengguna akhir, tetapi juga menyediakan fasilitas khusus bagi pengelola sistem. Salah satu komponen penting adalah *Admin Panel*, yang dirancang untuk memudahkan pengelolaan data dan konten pembelajaran. Melalui fitur ini, admin dapat menambahkan, mengubah, atau menghapus data tanpa perlu melakukan pembaruan langsung pada kode sumber aplikasi. Tampilan antarmuka hasil implementasi *Admin Panel* dapat dilihat pada Gambar 6.



Gambar 6. Panel Admin untuk Manajemen Konten Dinamis

Gambar 6 memperlihatkan antarmuka *Admin Panel*, fitur eksklusif untuk pengguna dengan peran admin. Panel ini berfungsi sebagai pusat kendali untuk pengelolaan konten pembelajaran secara dinamis tanpa mengubah kode aplikasi. Setiap komponen dalam *Admin Panel* dirancang agar saling terintegrasi, sehingga seluruh aktivitas pengelolaan dapat dilakukan secara efisien dan konsisten. Fungsionalitas utama meliputi:

1. *Dashboard* utama: navigasi ke semua fitur manajemen, termasuk pengelolaan materi, soal kuis, soal *listening*, latihan *speaking*, dan pemantauan statistik pengguna.
2. Manajemen materi dan soal (CRUD): admin dapat menambah materi baru (PDF/video) beserta judul dan tautan Google Drive, membuat soal *listening* baru dengan *link* audio, level, topik, dan pilihan jawaban, menambahkan frasa untuk latihan *speaking* beserta level, romaji, dan terjemahan, serta menyusun soal kuis pilihan ganda dengan pertanyaan dan beberapa opsi jawaban.
3. Pembaruan konten *real-time*: setiap perubahan yang dilakukan admin tersimpan di *Cloud Firestore* dan otomatis tampil pada aplikasi pengguna tanpa pembaruan aplikasi.

4.4. *Pembaruan dan Dinamika Materi Pembelajaran*

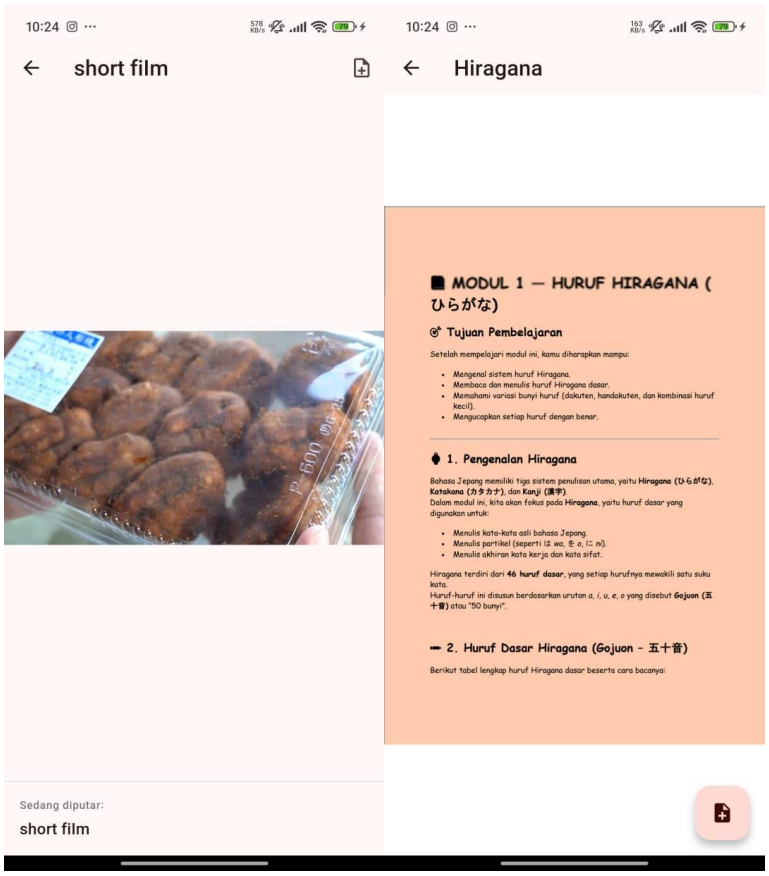
Keunggulan arsitektur Andana terletak pada pemisahan antara logika aplikasi dan konten, sehingga mendukung pembaruan materi secara *real-time*. Admin dapat memperbarui konten kapan saja melalui *Cloud Firestore*, dan perubahan tersebut langsung tersinkronisasi tanpa memerlukan pembaruan aplikasi di sisi pengguna. Pendekatan ini menjadikan Andana solusi *cloud-based* yang tangkas, mudah dipelihara, dan hemat biaya distribusi. Arsitektur ini juga memungkinkan pengembang melakukan pemisahan tanggung jawab (*separation of concern*) yang lebih jelas antara pengelolaan data dan tampilan antarmuka pengguna.

Materi dalam bentuk PDF ditampilkan menggunakan *flutter_pdfview*. Untuk meningkatkan efisiensi, aplikasi memanfaatkan mekanisme *caching* sederhana menggunakan *path_provider* dan *http* sehingga dokumen yang pernah diunduh dapat diakses ulang lebih cepat dan tetap tersedia saat koneksi internet terbatas. Implementasi

p-ISSN : 2803-1507 e-ISSN : 2803-1531

pembaca PDF yang menyatu dengan aplikasi meminimalkan interupsi alur belajar karena pengguna tidak perlu keluar ke aplikasi pihak ketiga. Selain itu, integrasi ini memperkecil risiko ketidaksesuaian format dokumen karena seluruh proses *rendering* dilakukan secara internal oleh sistem aplikasi.

Hasil implementasi aplikasi Andana menunjukkan bahwa sistem pembelajaran telah dilengkapi dengan berbagai media interaktif untuk mendukung pengalaman belajar yang lebih dinamis. Salah satu pengembangan penting adalah penyematan materi berbasis audiovisual agar pengguna dapat memahami pelafalan dan percakapan dengan lebih alami. Integrasi media ini dikembangkan agar dapat berjalan langsung di dalam aplikasi tanpa perlu membuka layanan eksternal. Hasil implementasi *fitur* pemutar video dan pembaca *PDF* ditunjukkan pada Gambar 7.

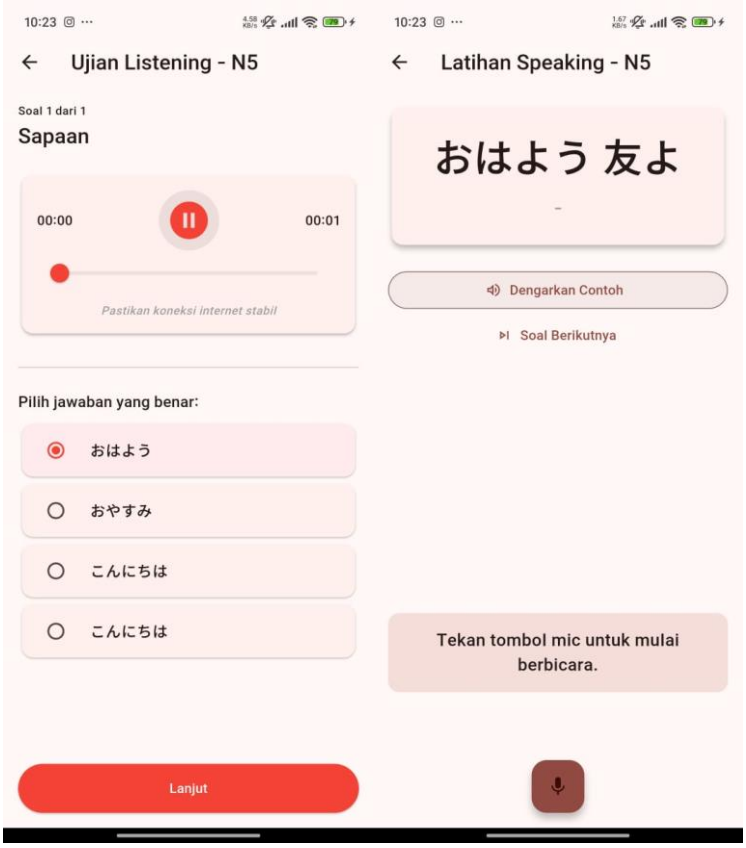


Gambar 7. Implementasi Pemutar Video Dan pembaca PDF dalam Aplikasi

Konten video pembelajaran diimplementasikan menggunakan kombinasi *video_player* sebagai mesin pemutar dasar dan *library chewie* untuk menyediakan antarmuka pengguna yang lebih kaya *fitur*, termasuk kontrol putar, jeda, *seek*, dan tampilan layar penuh. Integrasi ini menghasilkan pengalaman belajar audiovisual yang lebih mulus, terutama untuk materi pengucapan dan percakapan. Selain itu, pembaca *PDF* dihadirkan untuk mempermudah akses materi teks seperti kosakata dan tata bahasa secara langsung di dalam aplikasi. Implementasi kedua media ini menunjukkan peningkatan kenyamanan dan fleksibilitas dalam proses belajar pengguna.

Hasil pengembangan berikutnya berfokus pada peningkatan interaktivitas melalui latihan berbasis audio dan pengenalan suara. Fitur ini dirancang agar pengguna dapat berlatih kemampuan mendengarkan (*listening*) dan berbicara (*speaking*) secara mandiri di dalam aplikasi. Setiap latihan diatur agar berjalan ringan, cepat, dan memberikan umpan balik langsung terhadap performa pengguna. Hasil implementasi *fitur* latihan *listening* dan *speaking* ditunjukkan pada Gambar 8.

p-ISSN : 2803-1507 e-ISSN : 2803-1531



Gambar 8. Fitur Latihan Interaktif: Listening dan Speaking

Fitur *Listening Test* memanfaatkan paket *audioplayers* agar pemutaran audio ringan dan minim *buffering*. Penggunaan *preload* dan *caching* pada audio yang sering diakses membantu menurunkan latensi saat berpindah soal, sehingga ritme latihan tetap terjaga dan pengguna dapat fokus pada pemahaman *listening*. Fitur *Speaking Practice* menggunakan *webview_flutter* untuk menjalankan layanan *speech-to-text* berbasis *Web Speech API*. Komunikasi antara *Flutter* dan *JavaScript* dikelola melalui *JavascriptChannel* agar transkripsi suara dapat diterima dan diverifikasi secara langsung oleh aplikasi. Dengan pendekatan ini, validasi pelafalan pengguna dapat ditampilkan secara *real-time* dalam bentuk umpan balik (*feedback*) kualitatif yang membantu koreksi spontan.

4.5. *Pembahasan Pengembangan dan Kendala yang Dihadapi*

Selama proses pengembangan aplikasi, tim pengembang menghadapi beberapa kendala teknis yang memberikan pembelajaran penting dalam aspek efisiensi sistem, pengelolaan sumber daya, serta pengalaman pengguna. Setiap kendala yang muncul menjadi bahan evaluasi untuk meningkatkan stabilitas dan performa aplikasi secara keseluruhan. Berbagai permasalahan tersebut berhasil diatasi melalui analisis penyebab dan penerapan solusi teknis yang tepat guna. Adapun pembahasan terkait kendala yang dihadapi beserta solusi yang diterapkan dijelaskan sebagai berikut:

1. Kendala kinerja audio: pada awalnya integrasi fitur *listening* mengalami *buffering* yang lambat akibat penggunaan *controller* yang tidak dioptimalkan untuk audio murni. Tambahan: Permasalahan ini teridentifikasi saat pengujian dilakukan pada beberapa perangkat dengan kecepatan prosesor yang berbeda. Penggunaan *package audioplayers* juga memberikan fleksibilitas dalam pengaturan *bitrate* dan format file audio yang sebelumnya belum tersedia pada implementasi awal. Solusi yang diterapkan adalah migrasi ke paket *audioplayers* yang lebih ringan dan dirancang khusus untuk audio, serta menyediakan kontrol *caching* yang lebih baik sehingga mengurangi latensi pada pemutaran berikutnya.
2. Kendala izin perangkat: latihan *speaking* memerlukan akses mikrofon yang sering kali gagal jika izin belum diberikan oleh pengguna. Masalah ini mengganggu alur pengguna karena aplikasi tampak

p-ISSN : 2803-1507 e-ISSN : 2803-1531

tidak merespons. Tambahan: Kondisi tersebut terjadi terutama pada perangkat dengan versi *Android* yang memiliki sistem keamanan tambahan terhadap akses sensor. Solusi yang diambil adalah mengimplementasikan paket *permission_handler* dengan memeriksa status izin pada *initState* dari halaman latihan. Jika izin belum diberikan, alur pengguna dialihkan untuk menampilkan dialog permintaan izin sistem sebelum *widget* utama dirender.

3. Kendala persistensi data lokal: preferensi tema (terang/gelap) tidak tersimpan setelah aplikasi ditutup. Tambahan: Permasalahan ini muncul saat pengguna melaporkan bahwa tema aplikasi selalu kembali ke mode *default* setiap kali aplikasi dibuka ulang. Pengujian menunjukkan bahwa variabel status tema belum tersimpan secara permanen di memori lokal, sehingga diperlukan mekanisme penyimpanan yang lebih konsisten. Solusi yang dilakukan adalah menggunakan *shared_preferences* untuk menyimpan pilihan tema pengguna dalam bentuk *key–value* sederhana di penyimpanan lokal perangkat.
4. Kendala efisiensi *query*: seiring bertambahnya data, *query* untuk mengambil catatan pengguna menjadi lambat karena memindai seluruh koleksi *notes*. Tambahan: Hasil pengujian menunjukkan bahwa waktu respons meningkat secara signifikan ketika jumlah data pengguna melebihi seratus entri, terutama pada koneksi internet dengan latensi tinggi. Solusi yang diterapkan adalah merestrukturisasi skema data dengan memanfaatkan subkoleksi *users/{uid}/notes*. Pendekatan ini memungkinkan *query* yang lebih spesifik dan terindeks otomatis oleh *Firestore*, sehingga mengurangi jumlah *document reads* secara drastis.

Selain kendala teknis tersebut, pemilihan tumpukan teknologi menjadi aspek penting. *Flutter* dipilih karena mendukung pengembangan lintas platform dengan konsistensi *widget*, performa tinggi melalui *Ahead-of-Time compilation*, dan siklus iterasi cepat berkat *hot reload*. Pada sisi *backend*, *Firebase* menyatukan autentikasi, basis data sinkron *real time*, serta penyimpanan media dalam satu ekosistem, sehingga integrasi fitur menjadi sederhana dan beban *devops* menurun. Pendekatan ini memberikan kestabilan dalam proses pengembangan serta memungkinkan tim untuk fokus pada optimalisasi fitur pembelajaran daripada penanganan infrastruktur teknis.

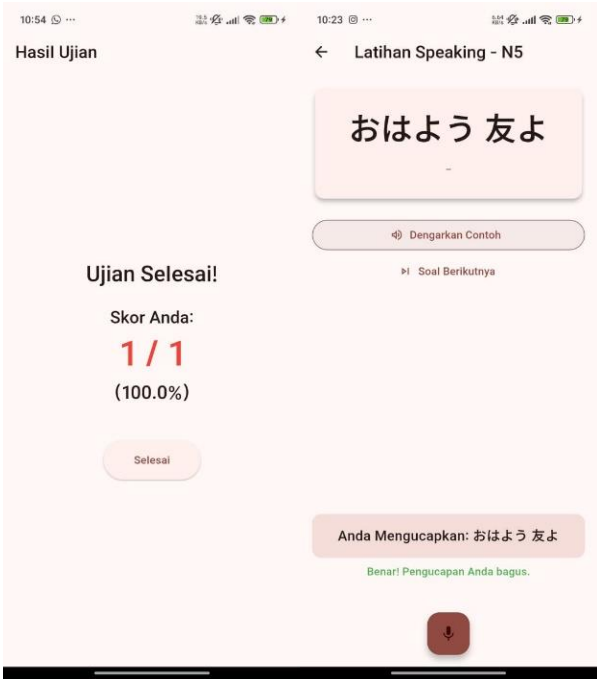
Secara komparatif, untuk kebutuhan aplikasi edukasi yang menuntut umpan balik langsung, pembaruan konten yang sering, dan tim pengembang yang ramping, kombinasi Flutter dan Firebase lebih sesuai dibanding alternatif seperti React Native dengan Supabase atau backend mandiri berbasis REST API. Flutter dan Firebase unggul pada kestabilan sinkronisasi real time, kemudahan penerapan security rules untuk pola RBAC, serta dukungan pembaruan konten over the air tanpa rilis aplikasi. Alternatif lain memberi keleluasaan arsitektural, tetapi cenderung menambah kompleksitas orkestrasi layanan dan biaya operasional untuk mencapai tingkat responsivitas yang sama. Dengan pertimbangan tersebut, pilihan teknologi pada Andana menawarkan rasio usaha terhadap hasil yang optimal pada fase R&D sekaligus jalur penskalaan yang jelas untuk pengembangan berikutnya.

4.6. *Evaluasi Hasil Pengembangan*

Evaluasi sistem dilakukan melalui serangkaian pengujian fungsional dengan metode *black-box testing*, di mana setiap *use case* diuji dari perspektif pengguna akhir. Hasil pengujian menunjukkan bahwa seluruh fungsionalitas berjalan sesuai dengan rancangan. Proses autentikasi, akses konten pembelajaran, pengerjaan semua jenis ujian, dan fitur personalisasi berfungsi tanpa kesalahan. Pengujian performa kualitatif juga menunjukkan waktu muat dan transisi antarlayar berada dalam batas yang dapat diterima. Mekanisme sinkronisasi data *real-time* melalui *Firestore* terbukti efektif tanpa penundaan signifikan pada kondisi jaringan normal.

Selain pengujian fungsionalitas, evaluasi juga dilakukan terhadap mekanisme *feedback* hasil belajar pengguna. Evaluasi ini bertujuan untuk melihat bagaimana sistem menampilkan hasil latihan dan memberikan umpan balik terhadap performa pengguna. Aspek yang diperhatikan meliputi kejelasan skor, kecepatan tampilan hasil, serta akurasi transkripsi pada latihan berbasis suara. Mekanisme umpan balik hasil belajar ini divisualisasikan pada Gambar 9.

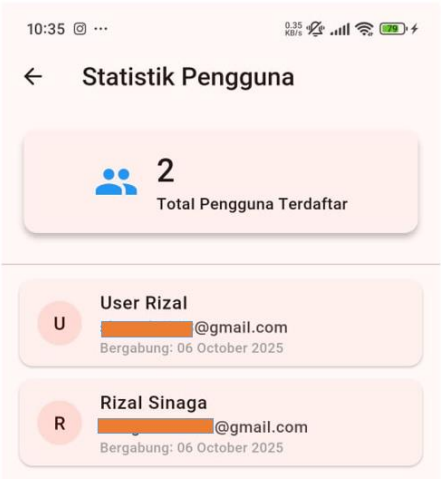
p-ISSN : 2803-1507 e-ISSN : 2803-1531



Gambar 9. Umpan Balik dan Evaluasi Hasil Belajar

Mekanisme umpan balik dan evaluasi hasil belajar ditunjukkan pada Gambar 9. Setelah pengguna menyelesaikan serangkaian soal, seperti kuis atau ujian *listening*, aplikasi menampilkan halaman hasil yang berisi skor akhir. Tampilan skor yang jelas, misalnya "1/1 (100,0%)", memberikan umpan balik *kuantitatif* yang instan mengenai performa pengguna. Pada latihan *speaking*, aplikasi menampilkan transkripsi dari ucapan pengguna disertai penilaian *kualitatif* secara *real-time*, yang efektif membantu pengguna pemula dalam mengoreksi serta meningkatkan pelafalan mereka.

Selanjutnya, hasil evaluasi juga mencakup pengujian pada sisi pengelolaan data melalui fitur *Admin Panel*. Fitur ini berfungsi untuk menampilkan statistik pengguna dan aktivitas yang terjadi di dalam aplikasi. Data yang disajikan meliputi jumlah total pengguna terdaftar, informasi dasar pengguna seperti nama, alamat email, serta tanggal bergabung. Hasil tampilan implementasi fitur ini ditunjukkan pada Gambar 10.



Gambar 10. Fitur Statistik Pengguna di Panel Admin

Fitur statistik pengguna pada *Admin Panel* ditunjukkan pada Gambar 10. Halaman ini menyajikan jumlah total pengguna yang terdaftar di dalam sistem, serta daftar pengguna individual yang memuat informasi dasar seperti nama, alamat email, dan tanggal bergabung. Fungsionalitas ini membantu *admin* dalam memantau perkembangan basis pengguna dan menjadi dasar pengambilan keputusan terkait kurasi konten serta

p-ISSN : 2803-1507 e-ISSN : 2803-1531

penjadwalan pembaruan materi. Berdasarkan hasil pengujian, fitur *Admin Panel*, termasuk statistik pengguna, berfungsi penuh dan setiap perubahan data yang dilakukan *admin* langsung tercermin pada aplikasi pengguna secara *real-time*.

5. DISKUI

Hasil penelitian menunjukkan bahwa sistem yang dikembangkan mampu menjalankan seluruh fungsionalitas secara konsisten, terutama pada aspek sinkronisasi real-time, pengelolaan materi, dan responsivitas antarmuka. Temuan ini menjelaskan bahwa arsitektur berbasis Flutter–Firebase dapat memenuhi tujuan penelitian untuk menghadirkan platform pembelajaran bahasa yang adaptif dan interaktif. Konsistensi performa sistem memperkuat temuan studi sebelumnya [20], [23] yang menekankan keunggulan arsitektur *cloud-based* dalam mendukung aplikasi edukasi. Implementasi ini menunjukkan bahwa pendekatan *cross-platform* berbasis Flutter mampu bekerja stabil untuk kebutuhan *mobile learning* dengan beban data dinamis.

Integrasi fitur pembelajaran seperti *flashcard*, latihan *listening*, dan *speaking* menunjukkan bahwa aplikasi mendukung pembelajaran aktif yang menggabungkan retensi memori jangka panjang dan stimulasi keterampilan reseptif-produktif. Hasil pengujian fitur ini memiliki koherensi dengan studi [4], [9], [15], [17] yang menegaskan bahwa gamifikasi, *spaced repetition*, dan latihan multimodal dapat meningkatkan retensi kosakata dan motivasi belajar. Temuan aplikasi menunjukkan performa sistem yang stabil tanpa latensi signifikan, yang dalam beberapa aspek menantang laporan studi sebelumnya mengenai kendala performa pada aplikasi ML berbasis multimedia. Kinerja ini menunjukkan bahwa pemanfaatan kombinasi Flutter dan Firebase dapat mengatasi sebagian besar kendala teknis yang umum ditemukan pada aplikasi pembelajaran bahasa.

Pengujian terhadap mekanisme *feedback* belajar memperlihatkan bahwa sistem dapat memberikan hasil kuis, transkripsi *speaking*, dan informasi performa pengguna secara instan. Respons cepat ini sesuai dengan hasil penelitian [10], [11], [17] yang menekankan pentingnya umpan balik langsung untuk peningkatan performa belajar dan koreksi pelafalan. Implementasi *speech-to-text* berbasis *Web Speech API* menunjukkan konsistensi transkripsi yang cukup baik meskipun terdapat variasi kualitas suara pengguna pada beberapa perangkat. Ketidakkonsistenan minor pada perangkat tertentu mencerminkan fenomena yang juga dilaporkan pada penelitian terdahulu, terutama terkait sensitivitas mikrofon dan latensi pengolahan suara.

Penelitian ini tetap memiliki beberapa keterbatasan yang perlu dicermati untuk memahami konteks capaian hasil. Evaluasi performa dan fungsionalitas belum melibatkan variasi skala pengguna yang lebih besar sehingga potensi kendala terkait beban server belum sepenuhnya teridentifikasi. Pengujian dilakukan dalam kondisi perangkat dan jaringan yang relatif stabil, sehingga belum mencerminkan seluruh dinamika penggunaan dalam lingkungan nyata. Selain itu, efektivitas pedagogis fitur seperti *spaced repetition*, gamifikasi, dan latihan multimodal belum diuji langsung terhadap capaian belajar jangka panjang pengguna.

6. KESIMPULAN DAN SARAN

6.1 *Kesimpulan*

Penelitian ini menghasilkan sebuah aplikasi pembelajaran bahasa Jepang berbasis *mobile learning* bernama Andana, yang dikembangkan menggunakan *Flutter* dan *Firebase*. Aplikasi ini dirancang untuk menghadirkan pengalaman belajar interaktif melalui integrasi fitur multimedia, gamifikasi, serta sistem penyimpanan data berbasis cloud. Andana mampu menjawab kebutuhan pembelajaran modern yang menuntut fleksibilitas, adaptivitas, dan aksesibilitas tinggi bagi pengguna. Pendekatan ini memungkinkan konten belajar disesuaikan dengan tingkat kemampuan pengguna dan memfasilitasi proses evaluasi secara berkesinambungan.

Dari hasil analisis dan pengembangan yang dilakukan, penerapan *mobile learning* dengan dukungan *Flutter–Firebase* secara efektif meningkatkan efisiensi pengembangan sistem sekaligus memberikan performa stabil di berbagai perangkat. Fitur gamifikasi terbukti memperkuat motivasi belajar pengguna, sementara penerapan prinsip *spaced repetition system* (*SRS*) membantu pengguna mempertahankan kosakata dalam jangka panjang. Integrasi *real-time database* pada Firebase juga memungkinkan pengguna melacak progres belajar secara otomatis dan sinkron antarperangkat. Penelitian ini berhasil mengembangkan platform pembelajaran bahasa Jepang yang interaktif, adaptif, dan berorientasi pengguna, sekaligus menyediakan kerangka kerja yang mendukung pengembangan lanjutan.

p-ISSN : 2803-1507 e-ISSN : 2803-1531

6.2 *Saran*

Berdasarkan hasil penelitian, terdapat beberapa saran yang dapat menjadi bahan pertimbangan untuk pengembangan lebih lanjut. Secara praktis, aplikasi Andana dapat ditingkatkan dengan menambahkan fitur *speech recognition* berbasis *artificial intelligence* untuk membantu latihan pengucapan dan mendengarkan. Secara metodologis, penelitian berikutnya disarankan untuk melakukan pengujian terhadap lebih banyak sampel pengguna dengan latar belakang belajar berbeda guna memperluas validitas hasil. Dari sisi teknis, integrasi dengan *learning analytics* dapat dilakukan untuk memantau pola belajar pengguna secara mendalam dan memberikan rekomendasi materi yang bersifat adaptif.

Pengembangan selanjutnya juga dapat mempertimbangkan penerapan *cloud computing* berbasis multi-region untuk meningkatkan kecepatan akses dan skalabilitas sistem. Pendekatan multi-region juga membantu mendukung pertumbuhan jumlah pengguna tanpa menurunkan performa sistem. Implementasi ini dapat menjadi dasar bagi pengembangan fitur lanjutan yang menuntut ketersediaan data secara cepat dan konsisten di berbagai lokasi. Pengembangan ini diharapkan memungkinkan Andana berfungsi tidak hanya sebagai media belajar bahasa Jepang, tetapi juga sebagai model referensi bagi penelitian dan inovasi lanjutan dalam bidang *educational technology* di Indonesia.

DAFTAR PUSTAKA

[1] K. Mohiuddin, H. Fatima, M. A. Khan, M. A. Khaleel, O. A. Nasr, and S. Shahwar, "Mobile Learning Evolution and Emerging Computing Paradigms: An Edge-Based cloud Architecture for Reduced Latencies and Quick Response Time," *Array*, vol. 16, p. 100259, Dec. 2022, doi: 10.1016/j.array.2022.100259.

[2] A. (Shahab, H. Jia, R. Nacheva, M. Bakaev, R. I. Maxim, and J. Arnedo-Moreno, "EDTF: A User-Centered Approach to Digital Educational Games Design and Development," *Information*, vol. 16, no. 9, p. 794, Sep. 2025, doi: 10.3390/info16090794.

[3] C. Song, S. Y. Shin, and K. S. Shin, "Implementing the Dynamic Feedback-Driven Learning Optimization Framework: A Machine Learning Approach to Personalize Educational Pathways," *Appl. Sci.*, vol. 14, no. 2, p. 916, Jan. 2024, doi: 10.3390/app14020916.

[4] V. N. Santhosh, D. Coutinho, A. V. Ankola, Y. K. Parimala, S. Shankkari, and K. Ragu, "Effectiveness of Spaced Repetition Learning Using a Mobile Flashcard Application Among Dental Students: A Randomized Controlled Trial," *J. Dent. Educ.*, vol. 88, no. 9, pp. 1267–1276, Sep. 2024, doi: 10.1002/jdd.13561.

[5] A. Alam and A. Mohanty, "Educational Technology: Exploring the convergence of Technology and Pedagogy Through Mobility, Interactivity, AI, and Learning Tools," *Cogent Eng.*, vol. 10, no. 2, p. 2283282, Dec. 2023, doi: 10.1080/23311916.2023.2283282.

[6] T. Shaik *et al.*, "A Review of the Trends and Challenges in Adopting Natural Language Processing Methods for Education Feedback Analysis," *IEEE Access*, vol. 10, pp. 56720–56739, 2022, doi: 10.1109/access.2022.3177752.

[7] G. Jošt and V. Taneski, "State-of-the-Art Cross-Platform Mobile Application Development Frameworks: A Comparative Study of Market and Developer Trends," *Informatics*, vol. 12, no. 2, p. 45, Apr. 2025, doi: 10.3390/informatics12020045.

[8] M. Asadullah, M. Yeasmin, A. F. Alam, A. Alsolami, N. Ahmad, and I. Atoum, "Towards a Sustainable Future: A Systematic Review of Mobile Learning and Studies in Higher Education," *Sustainability*, vol. 15, no. 17, p. 12847, Aug. 2023, doi: 10.3390/su151712847.

[9] S. Figueiredo, "The Effect of Mobile-Assisted Learning in Real Language Attainment: A Systematic Review," *J. Comput. Assist. Learn.*, vol. 39, no. 4, pp. 1083–1102, Aug. 2023, doi: 10.1111/jcal.12811.

[10] T. Inoue, Y. Chen, and T. Ohyanagi, "Assessing Handwriting Skills in a Web Browser: Development and Validation of an Automated Online Test in Japanese Kanji," *Behav. Res. Methods*, vol. 57, no. 1, pp. 1–17, Dec. 2024, doi: 10.3758/s13428-024-02562-6.

[11] C. M. Chen, L. Ming-Chaun, and C. P. Kuo, "A Game-Based Learning System Based on Octalysis

p-ISSN : 2803-1507 e-ISSN : 2803-1531

Gamification Framework to Promote Employees' Japanese Learning," *Comput. Educ.*, vol. 205, p. 104899, Nov. 2023, doi: 10.1016/j.compedu.2023.104899.

[12] Q. Zhou, J. Zhang, and Z. Zhu, "Evaluating Urban Visual Attractiveness Perception Using Multimodal Large Language Model and Street View Images," *Buildings*, vol. 15, no. 16, p. 2970, Aug. 2025, doi: 10.3390/buildings15162970.

[13] H. Ren, W. Wei, and W. Zhang, "The Influencing Factors on the Application of Information Technology Among Japanese Language Teachers in Chinese Universities," *Asia-Pacific Educ. Res.*, vol. 34, no. 2, pp. 825–841, Aug. 2024, doi: 10.1007/s40299-024-00899-2.

[14] M. C. Demirdag, S. Kucuk, and A. Tasgin, "An Investigation of the Effectiveness of Augmented Reality Technology Supported English Language Learning Activities on Preschool Children," *Int. J. Hum. Comput. Interact.*, vol. 41, no. 4, pp. 2410–2423, 2025, doi: 10.1080/10447318.2024.2323278.

[15] W. Liu *et al.*, "Location- and Physical-Activity-Based Application for Japanese Vocabulary Acquisition for Non-Japanese Speakers," *Multimodal Technol. Interact.*, vol. 7, no. 3, p. 29, Mar. 2023, doi: 10.3390/mti7030029.

[16] -K ; Lo, C.-K. Flipped, L.-K. Ng, and C.-K. Lo, "Flipped Classroom and Gamification Approach: Its Impact on Performance and Academic Commitment on Sustainable Learning in Education," *Sustainability*, vol. 14, no. 9, p. 5428, Apr. 2022, doi: 10.3390/su14095428.

[17] C. Halkiopoulos and E. Gkintoni, "Leveraging AI in E-Learning: Personalized Learning and Adaptive Assessment through Cognitive Neuropsychology—A Systematic Analysis," *Electronics*, vol. 13, no. 18, p. 3762, Sep. 2024, doi: 10.3390/electronics13183762.

[18] N. El-Haggar, L. Amouri, A. Alsumayt, F. H. Alghamedy, and S. S. Aljameel, "The Effectiveness and Privacy Preservation of IoT on Ubiquitous Learning: Modern Learning Paradigm to Enhance Higher Education," *Appl. Sci.*, vol. 13, no. 15, p. 9003, Aug. 2023, doi: 10.3390/app13159003.

[19] M. Abdollahi, S. F. Yeganli, M. Baharloo, and A. Baniasadi, "Hardware Design and Verification with Large Language Models: A Scoping Review, Challenges, and Open Issues," *Electronics*, vol. 14, no. 1, p. 120, Jan. 2025, doi: 10.3390/electronics14010120.

[20] K. I. Manik, R. Wahyudi, F. Amanah, M. Y. Noor, and A. Perdana, "Implementasi Flutterflow dan Firebase Firestore dalam Pengembangan Aplikasi Manajemen Keuangan Berbasis No-Code," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 9, no. 4, pp. 5929–5935, May 2025, doi: 10.36040/jati.v9i4.13961.

[21] N. Nikhlis, M. Jamal, and U. Ghofur, "Deteksi Dini Pencurian Data pada Perangkat Seluler Menggunakan Machine Learning," *J. Ilm. Sist. Inf.*, vol. 4, no. 2, pp. 110–129, May 2025, doi: 10.51903/fz0xzt26.

[22] M. Ur Rehman, S. Kuhn, E. Dritsas, and M. Trigka, "A Survey on the Applications of Cloud Computing in the Industrial Internet of Things," *Big Data Cogn. Comput.*, vol. 9, no. 2, p. 44, Feb. 2025, doi: 10.3390/bdcc9020044.

[23] S. Cazco *et al.*, "Optimization of Emergency Notification Processes in University Campuses Through Multiplatform Mobile Applications: A Case Study," *Computers*, vol. 14, no. 11, p. 453, Oct. 2025, doi: 10.3390/computers14110453.

[24] A. S. Hakim, P. J. Mustaqim, and A. Naufal, "Peran Pendidikan Digital dalam Melindungi Privasi Pengguna dan Mencegah Dampak Sosial," *J. Ilm. Sist. Inf.*, vol. 4, no. 2, pp. 162–174, May 2025, doi: 10.51903/k9cvkc90.

[25] S. U. Anggono and S. Supriyanto, "Peluang dan Tantangan Sistem Pelatihan Berbasis AI dalam Meningkatkan Keterampilan Tenaga Kerja di Era Digital," *J. Ilm. Sist. Inf.*, vol. 4, no. 2, pp. 224–231, May 2025, doi: 10.51903/6mbmzk12.