

Perancangan Aplikasi Pembelajaran Materi TOEFL Structure and Written Expressions Berbasis Android

Muchlis Polin^a, Sri Ariyanti Sabiku^b

^a Program Studi Sistem Informasi, Universitas Negeri Gorontalo

^b Program Studi Pariwisata Perhotelan, Universitas Negeri Gorontalo

Abstract

The Test of English as a Foreign Language (TOEFL) is an internationally standardized test designed to measure English proficiency, particularly for individuals who wish to pursue higher education or work in an English-speaking environment. One of the most challenging sections for test-takers is the Structure and Written Expressions section, which assesses grammar and sentence structure comprehension. This study aims to design and develop an Android-based learning application to assist users in understanding and practicing TOEFL Structure and Written Expressions questions. The application development follows the prototype method, which consists of five main stages: (1) communication, (2) quick plan, (3) modeling quick design, (4) construction of prototype, and (5) deployment delivery & feedback. To speed up the development process, the application was designed using Kotlin and Jetpack Compose. Users can select various practice questions and view their scores for each exercise. Future development of this application may include additional learning materials and in-depth analysis of practice results, allowing users to access detailed explanations of their mistakes and comprehensive information about correct answers.

Keywords : TOEFL, Structure and Written Expressions, Android, Digital Learning, Prototype.

Abstrak

TOEFL (*Test of English as a Foreign Language*) merupakan salah satu tes standar internasional untuk mengukur kemampuan berbahasa Inggris, terutama bagi mereka yang ingin melanjutkan studi atau bekerja di lingkungan berbahasa Inggris. Salah satu bagian yang sering menjadi tantangan bagi peserta adalah bagian *Structure and Written Expressions*, yang menguji pemahaman terhadap tata bahasa dan struktur kalimat. Penelitian ini bertujuan untuk merancang dan mengembangkan aplikasi pembelajaran berbasis Android agar dapat membantu pengguna dalam memahami dan berlatih soal *TOEFL Structure and Written Expressions*. Pengembangan aplikasi menggunakan metode prototype, yang terdiri dari lima tahap utama: *communication, quick plan, modeling quick design, construction of prototype, deployment delivery & feedback*. Agar pengembangan aplikasi bisa berjalan dengan cepat, aplikasi dikembangkan dengan menggunakan Kotlin dan Jetpack Compose. Dalam aplikasi ini, pengguna dapat memilih beberapa latihan soal dan kemudian melihat skor hasil latihan untuk setiap soal. Kedepannya, aplikasi ini dapat dikembangkan lebih lanjut dengan menambahkan fitur materi pembelajaran dan analisis hasil latihan lebih mendalam, dimana pengguna dapat melihat penjelasan lebih rinci mengenai kesalahan yang telah dibuat maupun informasi detail mengenai jawaban yang benar.

Keywords: TOEFL, Structure and Written Expressions, Android, pembelajaran digital, prototipe

1. Pendahuluan

Ujian TOEFL ITP (Institutional Testing Program) adalah ujian kemampuan Bahasa Inggris yang sering digunakan untuk kepentingan akademik maupun pekerjaan di Indonesia. Tes ini terdiri dari tiga bagian, yaitu Listening, Structure & Written Expressions, dan Reading (Dalimunte dkk., 2023). Bagian *Structure and Written Expressions* bertujuan untuk mengukur kemampuan peserta dalam memahami tata bahasa dan struktur kalimat yang benar sesuai dengan kaidah bahasa Inggris. Kompetensi ini

penting, terutama bagi mahasiswa dan profesional yang berencana untuk melanjutkan pendidikan atau bekerja di lingkungan internasional.

Meskipun TOEFL dapat menjadi salah satu indikator tingkat kemampuan bahasa Inggris seseorang, masih banyak yang mengalami kesulitan dalam menghadapi tes ini. Hal ini disebabkan oleh kurangnya persiapan, pengetahuan kosakata, waktu pengerjaan yang dirasa kurang panjang, dan tes yang dirasa kurang familiar (Slamet & Sulistyarningsih, 2021).

Persiapan yang baik untuk TOEFL, termasuk pemahaman yang mendalam tentang bagian *Structure and Written Expressions*, dapat meningkatkan performa peserta dalam ujian ini. Salah satu studi yang relevan adalah penelitian oleh Sudrajat dan Astuti yang menunjukkan bahwa siswa yang mengikuti kursus persiapan TOEFL secara online merasa lebih percaya diri dan termotivasi dalam memahami tata bahasa Inggris, yang mencakup materi struktur dan ekspresi tertulis (Sudrajat & Astuti, 2018).

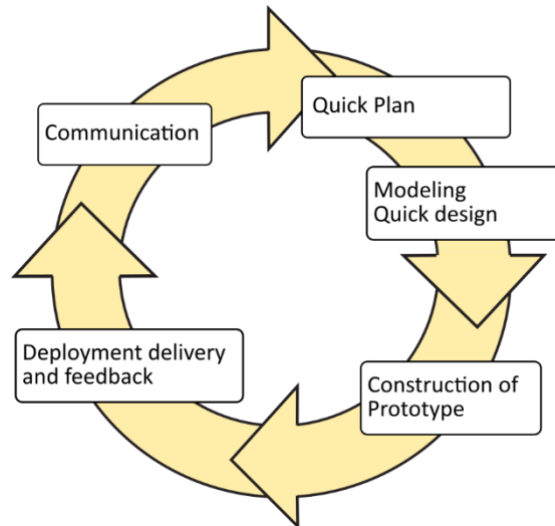
Dalam era digital saat ini, penggunaan teknologi berbasis aplikasi Android telah menjadi solusi praktis dalam pembelajaran (Khairani, 2022; Kirana, 2021). Aplikasi pembelajaran berbasis Android, misalnya, dapat menawarkan pendekatan yang lebih fleksibel, interaktif, dan personal bagi pengguna. Dengan memanfaatkan aplikasi berbasis Android, pengguna dapat belajar kapan saja dan di mana saja, serta mendapatkan feedback langsung atas latihan yang dilakukan.

Berangkat dari kebutuhan tersebut, penelitian ini bertujuan untuk merancang aplikasi pembelajaran "*TOEFL Structure and Written Expressions*" berbasis Android. Aplikasi ini diharapkan mampu memberikan pengalaman belajar yang lebih menarik, interaktif, dan efektif, sehingga dapat membantu pengguna meningkatkan kemampuan mereka dalam menghadapi tes TOEFL, khususnya pada bagian struktur dan ekspresi tertulis. Penelitian ini juga bertujuan untuk memberikan kontribusi dalam pengembangan media pembelajaran berbasis teknologi yang sesuai dengan kebutuhan era digital, sekaligus mendukung pengembangan kemampuan bahasa Inggris bagi pelajar dan masyarakat luas.

2. Metode Penelitian

2.1 Metode Prototype

Penelitian ini menggunakan metode pengembangan sistem prototype yang diusulkan oleh Pressman (Pressman, 2010), dengan tahapan: *Communication, Quick Plan, Modeling Quick Design, Construction of Prototype, Deployment Delivery & Feedback*, dan dilanjutkan dengan siklus berikutnya untuk perbaikan aplikasi melalui tahap *Communication* lagi. Tahapan-tahapan ini dirancang untuk memastikan pengembangan aplikasi dilakukan secara iteratif dan partisipatif, melibatkan calon pengguna dalam proses evaluasi untuk menghasilkan produk akhir yang sesuai kebutuhan. Berikut adalah penjelasan masing-masing tahap:



Gambar 1. Metode Prototype (Pressman, 2010)

1. *Communication*

Tahap ini bertujuan untuk memahami kebutuhan pengguna dan menetapkan spesifikasi awal aplikasi. Hasil yang diharapkan dari tahapan ini adalah dokumen spesifikasi awal, meliputi kebutuhan fungsional (contoh: latihan soal interaktif, analisis skor) dan kebutuhan non-fungsional (contoh: kemudahan penggunaan, akses offline).

2. *Quick Plan*

Tahap ini berfokus pada perencanaan awal pengembangan aplikasi untuk memastikan alur kerja jelas dan terarah. Tahapan ini akan menghasilkan roadmap pengembangan aplikasi dengan alur yang jelas.

3. *Modeling Quick Design*

Tahap ini menghasilkan rancangan awal aplikasi yang menggambarkan tampilan dan alur kerja yaitu berupa desain user interface sederhana serta alur kerja aplikasi dengan menggunakan tools Unified Modeling Language (UML).

4. *Construction of Prototype*

Pada tahap ini, prototype awal aplikasi dikembangkan berdasarkan desain yang telah dibuat.

5. *Deployment, Delivery & Feedback*

Tahap ini melibatkan pengujian prototype oleh pengguna dan pengumpulan umpan balik untuk perbaikan aplikasi pada siklus pengembangan berikutnya.

2.2 Perangkat Lunak

Untuk dapat mengembangkan aplikasi pada platform Android, diperlukan beberapa perangkat lunak yaitu:

1. Windows 11 sebagai sistem operasi
2. Android SDK
3. Android SDK Tools

4. Android Studio

Sejak tahun 2019, Google telah mengumumkan penggunaan Kotlin sebagai bahasa pemrograman utama untuk Android (Google for Developers, 2024b). Oleh karena itu, aplikasi ini dikembangkan menggunakan Kotlin.

Penggunaan Kotlin memberikan berbagai keuntungan dalam pengembangan aplikasi Android. Selain membuat proses pengembangan lebih efisien dan cepat, Kotlin juga memiliki sintaks yang lebih ringkas dan mudah dipahami dibandingkan Java, sehingga dapat mengurangi jumlah kode yang harus ditulis (Putranto dkk., 2020). Selain itu, Kotlin mendukung interoperabilitas dengan Java, memungkinkan pengembang menggunakan kode Java yang sudah ada tanpa perlu menulis ulang (Oliveira dkk., 2020).

2.3 Perangkat Keras

Untuk dapat mendukung proses pengembangan aplikasi dengan baik, maka dibutuhkan spesifikasi perangkat keras agar melebihi kebutuhan minimum pengembangan sistem dari Android Studio (Google for Developers, 2024a). Untuk itu, perangkat keras yang digunakan untuk pengembangan adalah sebagai berikut:

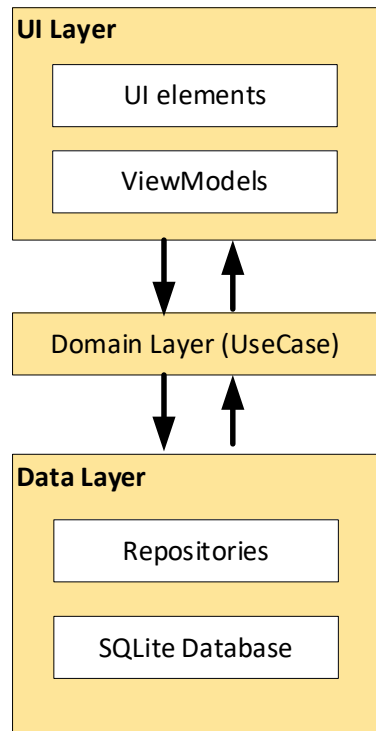
1. AMD Ryzen 7 8700F
2. RAM 16 GB
3. Harddisk 512 GB
4. Kartu Grafis Nvidia RTX 3050
5. Resolusi layar 1920x1080

3. Hasil dan Pembahasan

3.1 Arsitektur Aplikasi

Aplikasi didesain dengan arsitektur yang disarankan oleh Google (Google for Developers, 2025). Arsitektur ini memiliki beberapa prinsip seperti:

- Arsitektur yang bersifat reaktif dan terdiri dari beberapa lapisan.
- Aliran data yang searah pada semua lapisan aplikasi.
- Lapisan *user interface* yang memiliki state holders untuk manajemen kompleksitas *user interface*.
- Penggunaan *coroutine dan flow*.
- Penggunaan *dependency injection* sesuai *best practices* di industri.



Gambar 1. Desain Arsitektur Aplikasi

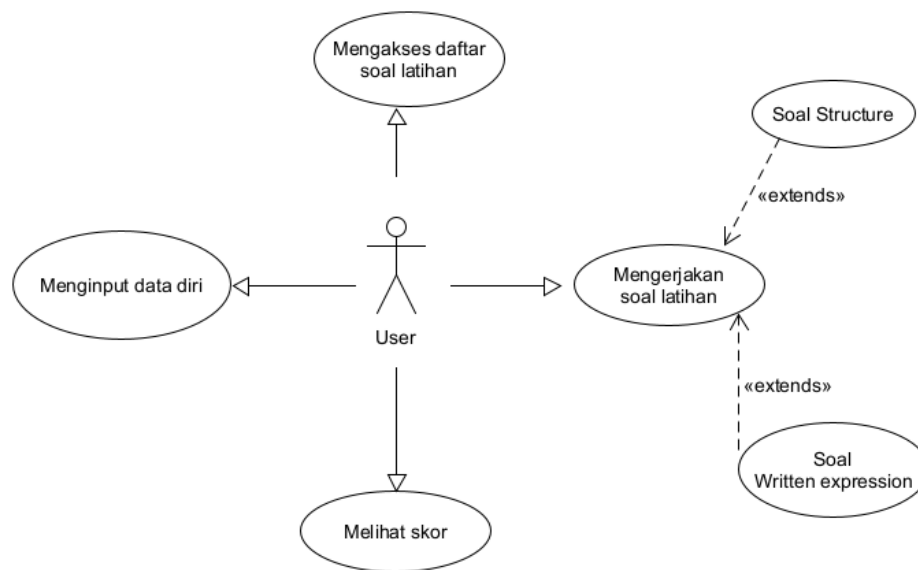
Dalam sistem ini, pengguna dapat mengajukan permintaan data melalui antarmuka pengguna (User Interface/UI). Permintaan tersebut kemudian diteruskan ke lapisan berikutnya, yaitu ViewModel, yang berfungsi sebagai penghubung antara UI dan lapisan data. ViewModel akan mengelola permintaan tersebut dan meneruskannya ke lapisan yang lebih dalam, yaitu *domain layer*, hingga mencapai tingkat terendah, yaitu database.

Setelah permintaan mencapai database, sistem akan mencari dan mengambil data yang sesuai berdasarkan permintaan pengguna. Data yang berhasil ditemukan kemudian dikirim kembali melalui proses berlapis, dimulai dari database ke lapisan repository. Dalam panduan arsitektur aplikasi android, repository bertanggung jawab untuk mengelola akses ke data, baik dari sumber lokal maupun jaringan, serta memastikan data yang dikirimkan sesuai dengan kebutuhan aplikasi. Akan tetapi, karena aplikasi ini bersifat *offline* maka tidak ada sumber data dari jaringan yang harus disinkronisasikan dengan database lokal.

Selanjutnya, data yang telah diproses oleh repository akan dikirim ke domain layer untuk kemudian diteruskan ke ViewModel. Di tahap ini, ViewModel dapat melakukan berbagai manipulasi data, seperti penyaringan data atau penyesuaian format ataupun tipe data, sebelum akhirnya mengirimkan data ke lapisan UI. Setelah data sampai di UI, informasi tersebut akan ditampilkan kepada pengguna dalam bentuk yang lebih mudah dipahami dan sesuai dengan kebutuhan aplikasi.

Dengan struktur berlapis seperti ini, sistem menjadi lebih terorganisir dan modular. Pemisahan tanggung jawab di setiap lapisan juga memudahkan proses pemeliharaan dan pengembangan aplikasi di masa mendatang.

3.2 Diagram *use case*



Gambar 2. *Use case diagram*

Use case diagram yang ditampilkan merepresentasikan model interaksi pengguna dengan sistem dalam aplikasi. Diagram ini menggambarkan berbagai fungsi utama yang dapat dilakukan oleh pengguna dalam aplikasi serta bagaimana setiap fungsi saling berkaitan. Pada aplikasi ini terdapat user sebagai satu-satunya aktor dalam sistem yang berinteraksi langsung dengan aplikasi untuk melakukan berbagai aktivitas pembelajaran *TOEFL Structure and Written Expressions*.

Berikut ini adalah penjelasan untuk setiap *use case* dalam aplikasi:

Use case Menginput Data Diri

Pengguna perlu memasukkan informasi pribadi yang diperlukan untuk mengakses fitur-fitur dalam aplikasi, seperti nama, email, dan tingkat kemampuan awal.

Use case Mengakses Daftar Soal Latihan

Pengguna dapat melihat daftar soal yang tersedia sebelum memilih dan mengerjakan latihan yang sesuai dengan kebutuhan mereka.

Use case Mengerjakan Soal Latihan

Fitur utama aplikasi yang memungkinkan pengguna berlatih mengerjakan soal terkait *TOEFL Structure and Written Expressions*. Melalui relasi *extends*, use case ini diperluas menjadi dua jenis soal utama dalam *TOEFL*:

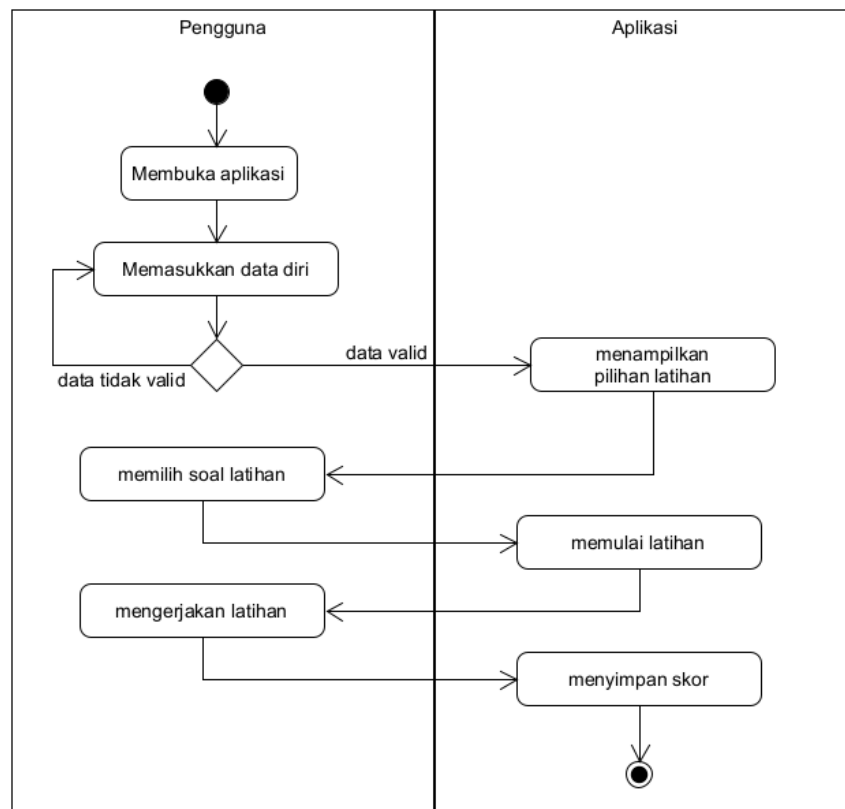
1. *Structure*: soal yang menguji pemahaman pengguna terhadap struktur gramatikal dalam bahasa Inggris.
2. *Written expressions*: soal yang mengukur kemampuan pengguna dalam mengenali kesalahan dalam ekspresi tertulis bahasa Inggris.

Use case Melihat Skor

Setelah menyelesaikan latihan, pengguna dapat mengakses skor hasil latihan mereka untuk memantau progres dan evaluasi kemampuan.

Use case diagram ini menunjukkan bahwa aplikasi yang dirancang memiliki fitur utama yang berfokus pada proses latihan dan evaluasi kemampuan pengguna dalam aplikasi ini. Diagram ini juga menekankan fleksibilitas dalam latihan dengan memungkinkan pengguna memilih jenis soal yang ingin dikerjakan serta menyediakan fitur analisis skor untuk membantu pemantauan progres pembelajaran.

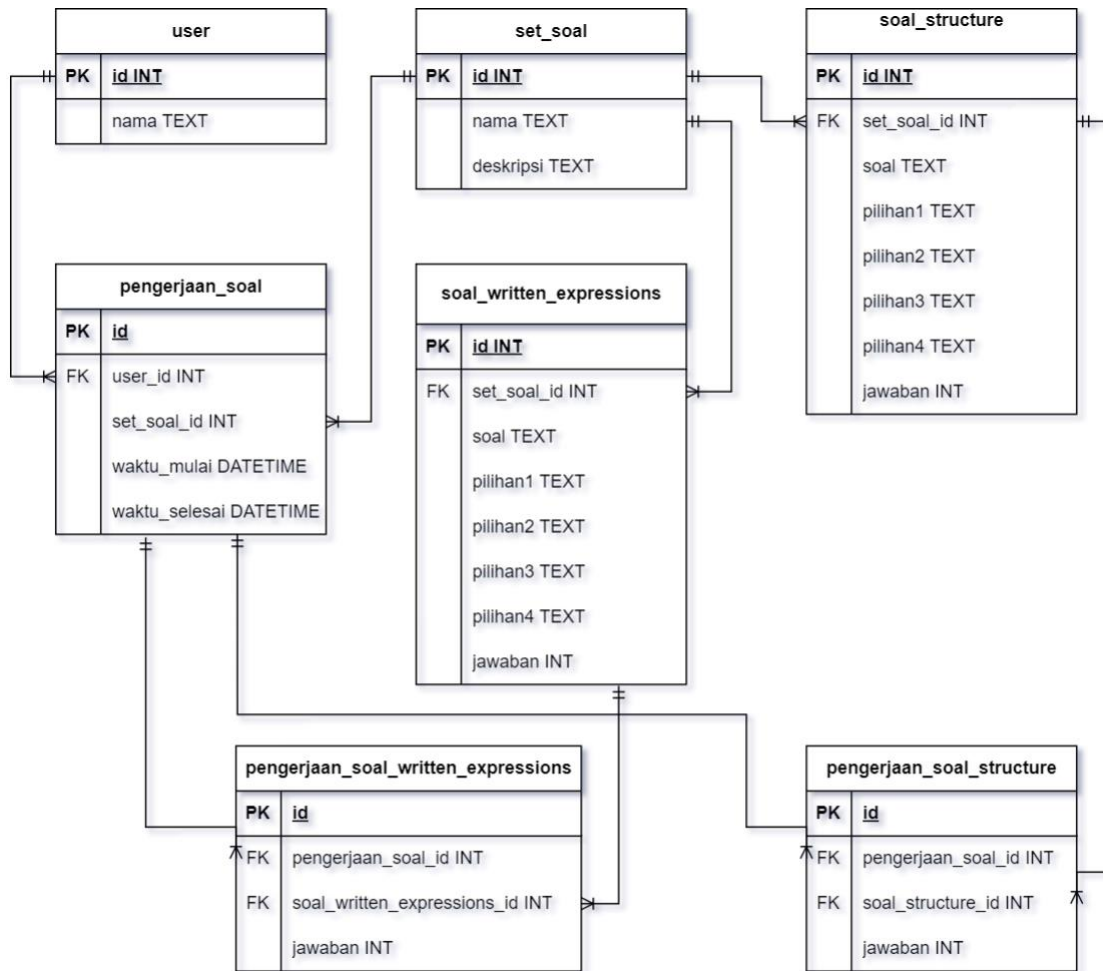
3.3 Activity Diagram



Gambar 3. Activity diagram

Activity diagram ini menggambarkan alur interaksi antara pengguna dan aplikasi dalam proses pembelajaran TOEFL Structure and Written Expressions berbasis Android. Pengguna memulai dengan membuka aplikasi dan memasukkan data diri. Jika data tidak valid, pengguna harus mengulanginya, sedangkan jika data valid, aplikasi menampilkan pilihan latihan. Pengguna kemudian memilih soal latihan, memulai latihan, dan mengerjakan soal. Setelah selesai, aplikasi menyimpan skor latihan sebagai hasil evaluasi, menandai akhir dari proses pembelajaran dalam aplikasi.

3.4 Desain Database

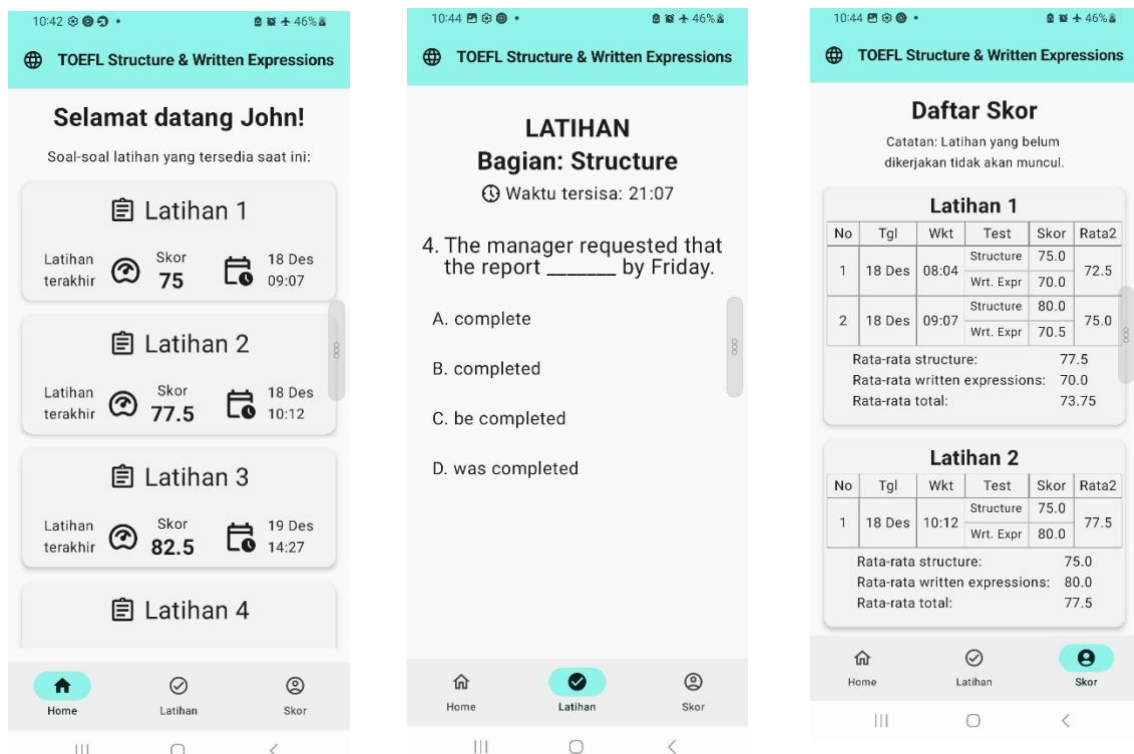


Gambar 4. Desain database aplikasi

Desain database yang digunakan dalam aplikasi ini ditampilkan pada Gambar 4. Skor yang didapatkan oleh pengguna tidak disimpan secara langsung pada database, tapi akan dihitung dari jawaban yang pernah dipilih oleh pengguna ketika skor perlu ditampilkan.

3.5 Desain Tampilan aplikasi

Setelah pengguna membuka aplikasi, pengguna akan diarahkan untuk memilih salah satu soal latihan. Pengguna juga dapat pindah ke layar Latihan ataupun Skor melalui menu navigasi pada bagian bawah aplikasi.



Gambar 5. Desain tampilan aplikasi

4. Kesimpulan & Saran

Penelitian ini menghasilkan sebuah aplikasi pembelajaran berbasis Android yang dirancang untuk membantu pengguna untuk mencoba berbagai latihan soal agar dapat meningkatkan penguasaan dalam materi *TOEFL Structure and Written Expressions*. Pengembangan aplikasi menggunakan metode *prototype* yang terdiri dari siklus yang dapat diulang, sehingga memungkinkan adanya perbaikan berdasarkan saran pengguna. Fitur utama yang disediakan dalam aplikasi meliputi akses daftar soal latihan, pengerjaan soal berdasarkan kategori Structure dan Written Expressions, serta tampilan skor hasil latihan.

Untuk pengembangan lebih lanjut, beberapa perbaikan dan fitur tambahan dapat diterapkan guna meningkatkan efektivitas aplikasi ini, seperti:

1. Penambahan mode pembelajaran, yaitu menyediakan materi teori dan strategi menjawab soal sebelum pengguna mengerjakan latihan.
2. Analisis hasil latihan lebih mendalam, dimana aplikasi dapat menampilkan detail kesalahan pengguna dan memberikan penjelasan mengenai jawaban yang benar.
3. Penggunaan *machine learning*, dengan cara mengubah tingkat kesulitan soal berdasarkan performa pengguna agar latihan lebih efektif.
4. Pengembangan versi web, menyediakan versi aplikasi berbasis web agar pengguna memiliki lebih banyak opsi dalam mengakses materi pembelajaran.

Dengan pengembangan lebih lanjut, aplikasi ini diharapkan dapat menjadi alat bantu yang dapat memberikan manfaat lebih dalam mendukung peningkatan kemampuan bahasa

Inggris pengguna, khususnya dalam mempersiapkan diri dalam mengerjakan soal TOEFL.

Daftar Pustaka

- Dalimunte, A. A., Tika, A., Dalimunte, M., Tanjung, A. P., & Suryani, I. (2023). Evaluating TOEFL ITP Test: A Critical Review. *Humanitatis : Journal of Language and Literature*, 9(2), Article 2. <https://doi.org/10.30812/humanitatis.v9i2.2329>
- Google for Developers. (2024a). *Install Android Studio*. Android Developers. <https://developer.android.com/studio/install>, diakses tanggal 28 April 2024
- Google for Developers. (2024b, April 5). *Android's Kotlin-first approach*. Android Developers. <https://developer.android.com/kotlin/first>, diakses tanggal 28 April 2024
- Google for Developers. (2025). *Guide to app architecture | App architecture*. Android Developers. <https://developer.android.com/topic/architecture>, diakses tanggal 28 Januari 2025
- Khairani, N. (2022). Penerapan Hasil Media Pembelajaran Berbasis Android. *Jurnal Vokasi Informatika*, 99–105. <https://doi.org/10.24036/javit.v2i2.5>
- Kirana, C. (2021). *Implementation of M-Learning Applications as Learning Media Based on Android | JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*. <https://jurnal.mdp.ac.id/index.php/jatisi/article/view/706>
- Oliveira, V., Teixeira, L., & Ebert, F. (2020). On the Adoption of Kotlin on Android Development: A Triangulation Study. *2020 IEEE 27th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, 206–216. <https://doi.org/10.1109/SANER48275.2020.9054859>
- Pressman, R. S. (2010). *Software Engineering: A Practitioner's Approach*. McGraw-Hill Education.
- Putranto, B. P. D., Saptoto, R., Jakaria, O. C., & Andriyani, W. (2020). A Comparative Study of Java and Kotlin for Android Mobile Application Development. *2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, 383–388. <https://doi.org/10.1109/ISRITI51436.2020.9315483>
- Slamet, J., & Sulistyarningsih, S. (2021). Students' Difficulties in Answering "Structure and Written Expression" TOEFL-like at STKIP PGRI Sidoarjo. *E-Structural (English Studies on Translation, Culture, Literature, and Linguistics)*, 4(01), Article 01. <https://doi.org/10.33633/es.v4i01.4410>
- Sudrajat, W. N. A., & Astuti, E. R. (2018). Students' Perceptions of the Use of TOEFL Preparation Online Course on the Test Performance: The Case of TOEFL Structure and Written Expression Test. *Humaniora*, 9(3), Article 3. <https://doi.org/10.21512/humaniora.v9i3.4807>