

Pengembangan Aplikasi *Daily Notes* berbasis Mobile App Inventor dengan TinyDB

Agus Suharto¹, Christien Rozali²

^{1,2}Program Studi Sistem Informasi, Universitas Pamulang, Indonesia

Email : dosen01539@unpam.ac.id , dosen02828@unpam.ac.id

Abstrak

Penelitian ini berfokus pada pengembangan aplikasi mobile dengan penerapan penyimpanan data lokal persisten melalui komponen TinyDB MIT App Inventor. Permasalahan adalah sifat sementara data aplikasi, yang biasanya hilang saat aplikasi ditutup, sehingga memerlukan metode untuk mempertahankan konten dan preferensi yang dibuat pengguna di seluruh sesi tanpa ketergantungan internet yang berkelanjutan. Tujuannya adalah merancang dan mengembangkan aplikasi *daily notes* berbasis mobile yang secara efektif untuk penyimpanan, pengambilan, dan manipulasi data lokal. Metode penelitian menggunakan pendekatan Research & Development melalui model ADDIE dengan tahapan Analisis, Design, Development, Implementation dan Evaluation. Hasil penelitian di evaluasi menggunakan EUCS dan Skala likert sebagai pengukurannya yaitu : Content 83% "Puas", Accuracy 74% "Puas", Ease Of Use 87% "Sangat Puas", Format 86% "Puas", Timeliness 69% "Puas", sehingga hasil akhir rata rata indeks yaitu 74,8%. Temuan ini menyoroti kemampuan TinyDB untuk meningkatkan pengalaman pengguna dengan memastikan kontinuitas data dan mengurangi ketergantungan pada server eksternal untuk manajemen data dasar.

Kata kunci—tinydb, eucs, mit app, Addie

Abstract

This research focuses on the development of mobile applications with the implementation of persistent local data storage through the MIT App Inventor TinyDB component. The problem is the temporary nature of application data, which is usually lost when the application is closed, thus requiring a method to maintain user-created content and preferences across sessions without continued internet dependency. The goal is to design and develop a mobile-based *daily notes* application that is effective for local data storage, retrieval, and manipulation. The research method uses the Research & Development approach through the ADDIE model with the stages of Analysis, Design, Development, Implementation and Evaluation. The results of the study were evaluated using EUCS and Likert Scale as measurements, namely: Content 83% "Satisfied", Accuracy 74% "Satisfied", Ease Of Use 87% "Very Satisfied", Format 86% "Satisfied", Timeliness 69% "Satisfied", so that the final result of the average index is 74.8%. These findings highlight TinyDB's ability to improve the user experience by ensuring data continuity and reducing dependence on external servers for basic data management.

Keywords—tinydb, eucs, mit app, Addie

1. PENDAHULUAN

Penggunaan perangkat mobile Smartphone tidak hanya berfungsi sebagai alat komunikasi, tetapi juga sebagai sarana produktivitas yang mendukung berbagai aktifitas, dalam menjalankan aktifitas sehari-hari pentingnya menulis kembali hal-hal yang telah dilihat atau dilakukan[1], termasuk pencatatan informasi dimana kegiatan mencatat merupakan kebutuhan dasar yang dilakukan oleh hampir semua orang, baik untuk keperluan pribadi, pendidikan, sekolah maupun proyek.

Aplikasi pencatat digital (*digital notes*) seperti Google Keep atau Evernote memang sudah tersedia luas, namun sebagian besar aplikasi tersebut membutuhkan koneksi internet dan memiliki fitur yang kompleks sehingga tidak selalu sesuai dengan kebutuhan pengguna yang menginginkan aplikasi sederhana, ringan, dan dapat digunakan secara offline. Sebagai catatan, untuk dapat menggunakan google keep, selain memahami fitur dan cara menggunakannya, kita juga memerlukan kuota internet yang mencukupi dan jaringan internet yang stabil[2]. Oleh karena itu, diperlukan solusi alternatif berupa aplikasi pencatat digital yang mudah digunakan, berfungsi secara offline, dan dapat dikembangkan dengan platform yang ramah bagi pemula.

MIT App Inventor adalah *tools* pemrograman visual drag-and-drop yang mudah digunakan untuk merancang dan membangun aplikasi seluler untuk Android dan iOS. Tools ini dapat mengubah ide seseorang menjadi aplikasi yang berfungsi tanpa perlu coding[3] atau merupakan salah satu platform pengembangan aplikasi mobile berbasis android dan apple yang bersifat open-source dan dirancang untuk memudahkan proses pembuatan aplikasi tanpa memerlukan kemampuan pemrograman tingkat lanjut. Salah satu komponen penting dalam MIT App Inventor adalah TinyDB, yaitu database lokal yang memungkinkan penyimpanan data secara permanen di perangkat pengguna. TinyDB adalah komponen database lokal sederhana yang tidak terlihat (*non visible component*) dimana dapat menyimpan data untuk suatu aplikasi. Ponsel atau tablet biasanya memiliki dua jenis memori utama yaitu RAM dan *flash*RAM. RAM adalah singkatan dari "*Random Access Memory*, RAM mempertahankan nilai selama daya dialirkan ke sirkuit RAM. Begitu kita mematikan daya, nilai yang tersimpan di RAM akan hilang. Aplikasi yang dibuat dengan App Inventor diinisialisasi setiap kali dijalankan tersimpan dalam RAM. Artinya jika aplikasi menetapkan nilai variabel dan pengguna kemudian keluar dari aplikasi, nilai variabel tersebut tidak akan tersimpan saat aplikasi dijalankan lagi. Sebaliknya, TinyDB adalah penyimpanan data persisten (nilai data tetap ada) tersimpan dalam flash RAM pada aplikasi tersebut. Data yang disimpan di TinyDB akan tersedia setiap kali aplikasi dijalankan. Contohnya adalah permainan yang menyimpan skor tinggi dan mengambilnya kembali setiap kali permainan dimainkan[4]. Dengan memanfaatkan TinyDB, aplikasi pencatat digital dapat menyimpan catatan, mengedit, dan menghapus data tanpa bergantung pada server eksternal.

Inti dari kemampuan penyimpanan MIT App Inventor terletak pada TinyDB, yakni komponen non-relasional bawaan yang berfungsi sebagai basis data NoSQL ringan untuk persistensi lokal perangkat. Tidak seperti alternatif berbasis cloud yang membutuhkan akses internet dan autentikasi, TinyDB menyimpan data langsung di ponsel pintar atau tablet pengguna dalam format kunci-nilai yang lugas, di mana "tag" berfungsi sebagai pengenal unik untuk mengambil nilai seperti daftar, angka, atau teks. Kesederhanaan ini selaras sempurna dengan etos App Inventor, yang memungkinkan bahkan mereka yang tidak memiliki latar belakang pemrograman untuk mengimplementasikan fitur-fitur seperti menyimpan skor permainan, pengaturan pengguna, atau daftar tugas dengan upaya minimal. Misalnya, dalam aplikasi agenda, TinyDB dapat secara otomatis memuat tugas yang telah dimasukkan sebelumnya saat diluncurkan ulang, meningkatkan kegunaan dan mengurangi risiko kehilangan data.

Melalui penelitian ini, penulis berupaya merancang dan membangun aplikasi *daily notes* berbasis Mobile App dengan memanfaatkan TinyDB sebagai media penyimpanan lokal. Aplikasi ini diharapkan dapat menjadi solusi praktis bagi pengguna yang membutuhkan pencatatan cepat, sederhana, dan dapat diakses kapan saja tanpa koneksi internet. Selain itu,

penelitian ini juga diharapkan dapat memberikan kontribusi dalam pengembangan aplikasi mobile berbasis MIT App Inventor, khususnya dalam pemanfaatan dan riset TinyDB untuk kebutuhan penyimpanan data lokal.



Gambar 1. TinyDB Mit App Inventor

2. METODE PENELITIAN

Untuk melakukan pendekatan sistematis terhadap penelitian, perancangan, dan pengembangan aplikasi mobile menggunakan TinyDB di MIT App Inventor adalah dengan metode R&D yang diadaptasi model ADDIE. Meskipun ADDIE biasanya digunakan untuk pengembangan program pendidikan atau pelatihan, fase-fase terstrukturnya menyediakan kerangka kerja yang kuat untuk penelitian dan pengembangan berbasis proyek. Model ini terkenal karena fleksibilitas dan kemampuan adaptasinya, yang memungkinkan perancang instruksional untuk menyesuaikan setiap fase agar sesuai dengan kebutuhan spesifik. Namun, dengan evolusi teknologi yang cepat, fase-fase ini memerlukan perpaduan yang harmonis antara alat dan metodologi digital[5]. Model ini juga dapat dengan efektif diterapkan dalam membangun aplikasi website dan aplikasi lain nya seperti IoT , karena tahapannya memberikan panduan yang terstruktur. Akronim dari ADDIE berasal dari lima tahapan yaitu *Analysis, Design, Development, Implementation, Evaluation*[6] seperti gambar 2 berikut dibawah ini :



Gambar 2. Tahapan Model ADDIE

1. Analysis (Analisis)

Pada Tahap ini adalah tahap awal model yakni mengidentifikasi kebutuhan pengguna terhadap aplikasi *daily notes*. Menganalisis masalah aplikasi Notes dan keterbatasan aplikasi notes yang ada serta menentukan spesifikasi fungsional aplikasi (fitur tambah, edit, hapus, pencarian catatan). Tahap ini juga merupakan gambaran analisis kebutuhan penelitian terhadap permasalahan yang perlu dilakukan, yang dapat dijabarkan pada Gambar 1 di atas [7].

2. Design (Perancangan)

Tahap selanjutnya adalah tahap Desain tahapan ini adalah melakukan rancangan struktur aplikasi menggunakan MIT App Inventor, mendesain antarmuka pengguna (UI) yang sederhana dan mudah dipahami, menentukan alur kerja aplikasi. Pada tahap Desain ini menguraikan bagaimana desain dapat mencapai tujuan yang telah ditentukan berdasarkan hasil analisis. Hal-hal yang ditekankan antara lain seperti target pengguna[8]. Selain itu, instruksi sistem menentukan penggunaan TinyDB sebagai media penyimpanan lokal.

3. Development (Pengembangan)

Setelah tahapan Desain selanjutnya adalah tahapan pengembangan dimana tahapan ini melakukan membangun aplikasi sesuai rancangan dengan MIT App Inventor, mengimplementasikan blok program untuk fungsi CRUD (Create, Read, Update, Delete) serta melakukan uji coba internal untuk memastikan aplikasi berjalan sesuai rancangan. Tahap pengembangan model ini dilakukan penyusunan model berbasis mobile yang siap untuk divalidasi dan direvisi berdasarkan masukan validator, selanjutnya dilakukan uji coba terbatas terhadap produk yang dikembangkan[9]

4. Implementation (Implementasi)

Setelah tahap pengembangan selanjutnya ke tahap implementasi dimana melakukan instalasi dan uji aplikasi pada perangkat Android, melibatkan sejumlah pengguna mahasiswa yang sedang mengambil mata kuliah mobile programming untuk mencoba aplikasi, mengumpulkan feedback terkait kemudahan penggunaan dan fungsionalitas.

5. Evaluation (Evaluasi)

Tahap evaluasi melakukan evaluasi sumatif yakni pengujian setelah implementasi. Tahapan pengujian menggunakan variabel penelitian yang digunakan dalam instrumen model EUCS adalah: Content/isi, Accurate/Akurasi, Format/Tampilan, Ease Of Use/kemudahan penggunaan, dan TimeLiness/ketepatan waktu.

Model EUCS mampu mengukur tingkat kepuasan pengguna terhadap suatu sistem dengan membandingkan antara harapan dan kenyataan dari sebuah sistem informasi yang menekankan pada kepuasan (*satisfaction*) pengguna, dengan cara menganalisa sistem berdasarkan isi (konten), keakuratan, tampilan, kemudahan pengguna dan ketepatan waktu [10]. Data kuesioner responden disesuaikan dengan pertanyaan dalam model EUCS. Hasil jawaban tersebut menjadi pedoman bagi Peneliti. Interval Pengukuran menggunakan Skala Likert Skala Likert adalah urutan kategori peringkat yang diurutkan, di mana hanya alat nonparametrik yang dapat digunakan. Menyimpulkan bahwa skala memiliki sifat yang sama jaraknya di antara kategori-kategori tersebut memungkinkan penggunaan alat parametrik yang lebih kuat dan tepat pada kemungkinan operasi matematika[11].

3. HASIL DAN PEMBAHASAN

3.1 Hasil

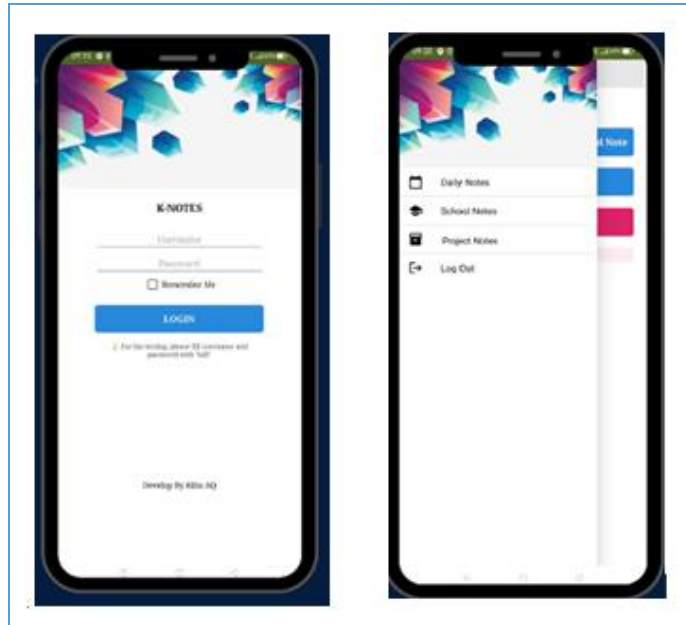
Hasil Penelitian adalah prototype Aplikasi *Daily Notes* berbasis Mobile berikut ini gambar aplikasi saat dijalankan tampilan awal *screenlogin* pada aplikasi seluler:

Selanjutnya tampilan screen beranda saat login berhasil dilakukan, pada screen terdapat 4 menu side bar yang bisa dipilih berikut adalah tabel 1 screen beranda:

Tabel 1. Halaman Beranda

| No | Teks Tampilan Aplikasi | Nama Screen | Icon |
|----|------------------------|--------------|----------------|
| 1 | Dashboard | Beranda | Home |
| 2 | Daily Notes | Daily Notes | calendar_today |
| 3 | School Notes | School Notes | School |
| 4 | Project Notes | Log Out | Logut |

Berikut ini tampilan hasil saat dijalankan pada perangkat mobile pada gambar 3 dibawah ini



Gambar 3 Login dan Beranda

3.2 Pembahasan

Pada Pembahasan peneliti melakukan tahapan metode R&D dimana menggunakan salah satu modelnya yaitu model ADDIE dengan tahapan Analisa, Desain, Pengembangan, Implementasi, Pengujian :

3.2.1 Analisis

Pada tahap analisis, dilakukan identifikasi kebutuhan pengguna terhadap aplikasi pencatat digital. Hasil analisis menunjukkan bahwa:

1. Pengguna membutuhkan aplikasi yang sederhana, ringan, dan dapat digunakan secara offline.
2. Aplikasi notes yang ada (Google Keep, Evernote) seringkali membutuhkan internet dan memiliki fitur kompleks.
3. Fitur utama yang dibutuhkan adalah tambah catatan, edit, hapus, dan pencarian catatan. Analisis ini menjadi dasar perancangan aplikasi dengan MIT App Inventor dan TinyDB.

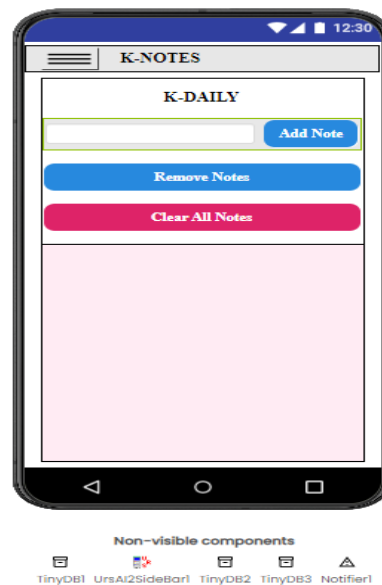
3.2.2 Desain

Tahap Desain melakukan penyusunan antar muka UI (*user interface*) terdiri dari *screen login*, *screen Home*, *daily notes*, *school notes* dan *project note* berikut ini desain antar muka login seperti pada gambar 5 dibawah ini:



Gambar 4 Desain Screen Login

Desain selanjutnya adalah screen halaman beranda berikut ini desain pada gambar 5:



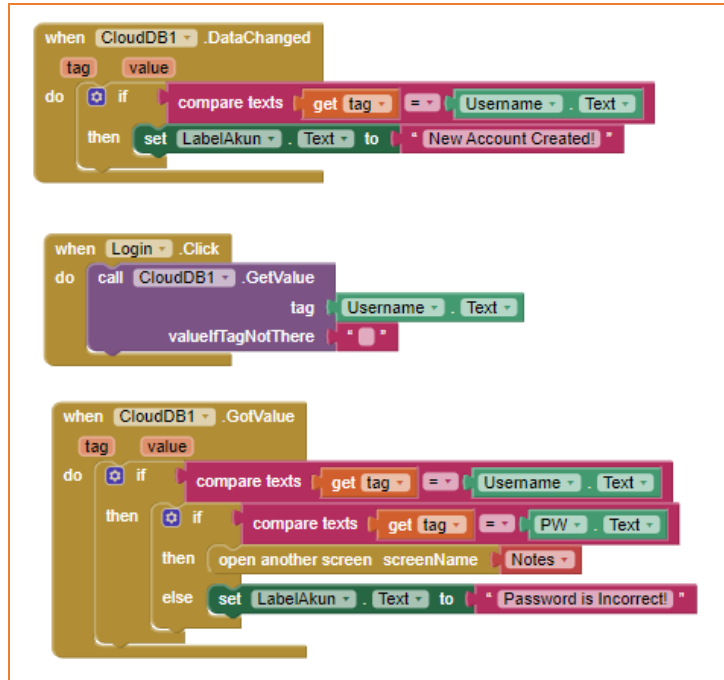
Gambar 5 Desain Halaman Beranda

Pada halaman beranda terdapat 3 penyimpanan menggunakan tinydb yaitu : TinyDb1 digunakan untuk penyimpanan *daily note*, tinydb2 untuk penyimpanan *School Note* dan tinydb3 untuk penyimpanan *project Notes*

3.2.3 Development (Pengembangan)

Setelah tahap desain dilakukan selanjutnya tahap pengembangan melibatkan pemrograman dengan perintah blocks MIT App inventor berikut ini gambar blocks untuk screen login :

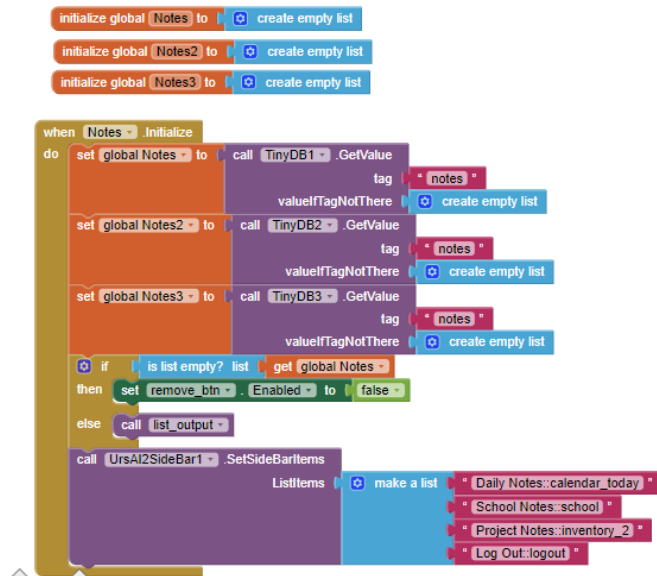
1. Screen Login



Gambar 6Block perintah *screenlogin*

Penjelasan perintah *block login* :

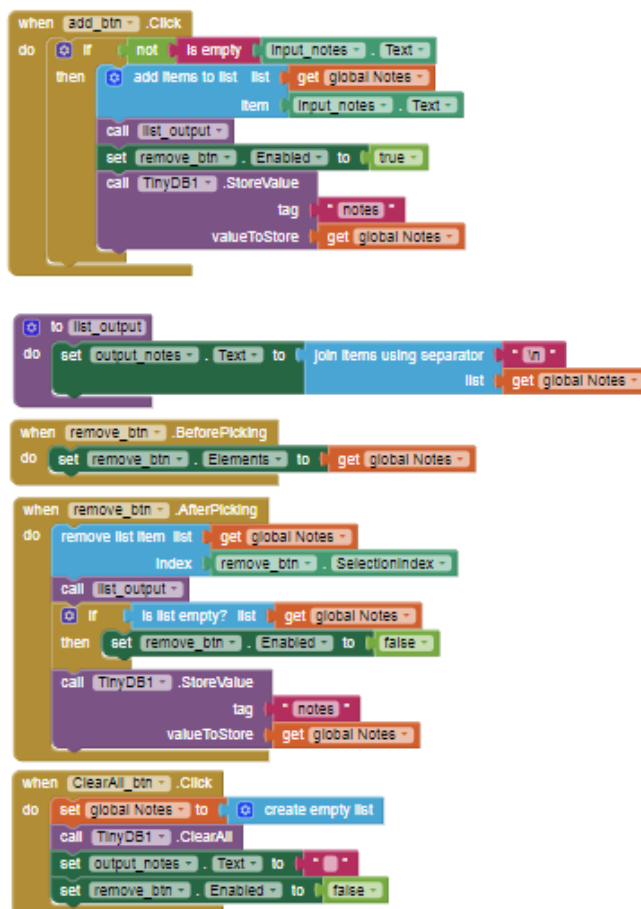
1. *When.cloud.datachanged* , event ini digunakan ketika ingin merubah / menambah user name dan password , jika berhasil akan menampilkan teks *New Account Created*
 2. *When.login.click* , *event* ini digunakan untuk mengambil teks pada *username* jika tag kunci tidak kosong
 3. *When.cloudDB.GotValue* , *event* ini digunakan ketika sudah mendapatkan teks yang di input kemudian dibandingkan dengan *username* apakah sudah ada pada cloudDB, jika *username* sesuai maka akan memanggil screen Notes(Beranda) jika tidak ada maka menampilkan pada kotak teks *Password Incorrect*
2. Screen Beranda
Berikut ini tampilan block perintah screen beranda jika password benar dan ditemukan seperti pada gambar 7



Gambar 7 Inisialisasi Global untuk 3 notes

Penjelasan :

Inisialisasi 3 variabel Global yaitu notes untuk Daily yang disimpan kedalam tinydb1 , notes2 untuk *schoolnotes* yang tersimpan kedalam tinydb2, lalu notes3 yang disimpan kedalam tinydb3.



Gambar 8. Proses penyimpanan, penghapusan, pembersihan data baru

3.2.4 Implementasi

Tahap implementasi dilakukan dengan menginstal aplikasi pada beberapa perangkat Android dan melibatkan pengguna mahasiswa mata kuliah *mobile programming* kemudian dilakukan test formatif yang dilakukan selama proses pengembangan untuk memperbaiki bug dan kesalahan jika ditemukan

3.2.5 Evaluasi

Pada Tahap Evaluasi penelitian ini melakukan sumatif evaluasi setelah tahap implementasi dilakukan test sumatif adalah menilai kinerja, kegunaan, dan skalabilitas aplikasi. Mengumpulkan umpan balik dari pengguna dan memantau analisis data untuk melakukan perbaikan. Evaluasi menggunakan pengujian beta dimana produk atau aplikasi diuji oleh sekelompok pengguna nyata di luar tim pengembangan sebelum peluncuran resminya. Untuk mengukur tingkat kepuasan pengguna akhir evaluasi dilakukan dengan kuesioner metode EUCS (End User Computing Satisfaction) evaluasi dimensi utama yaitu : Content, Accuracy, Format, Ease Of Use, Timeliness Interval penilaian dengan skala likert 1 s/d 5, oleh 20 responden mahasiswa yang mengikuti mata kuliah mobile programming . Beberapa penelitian telah mendefinisikan ulang atau memperluas model EUCS untuk melayani kebutuhan mereka dan aspek teknologi baru yang muncul. Model EUCS telah digunakan dalam mengevaluasi kepuasan pengguna terhadap berbagai sistem dan dengan berbagai cara untuk menafsirkan berbagai aspek teknologi.

Berikut ini tabel interval skala likert

Tabel 2 Skala Likert

| No | Indeks Penilaian | Kategori |
|----|------------------|-------------------|
| 1 | 0%-19,99% | Sangat Tidak Puas |
| 2 | 20%-39,99% | Tidak Puas |
| 3 | 40%-59,99% | Netral |
| 4 | 60%-79,99% | Puas |
| 5 | 80%-100% | Sangat Puas |

Berikut tabel Hasil Evaluasi Kuesioner EUCS hasil uji penelitian beta :

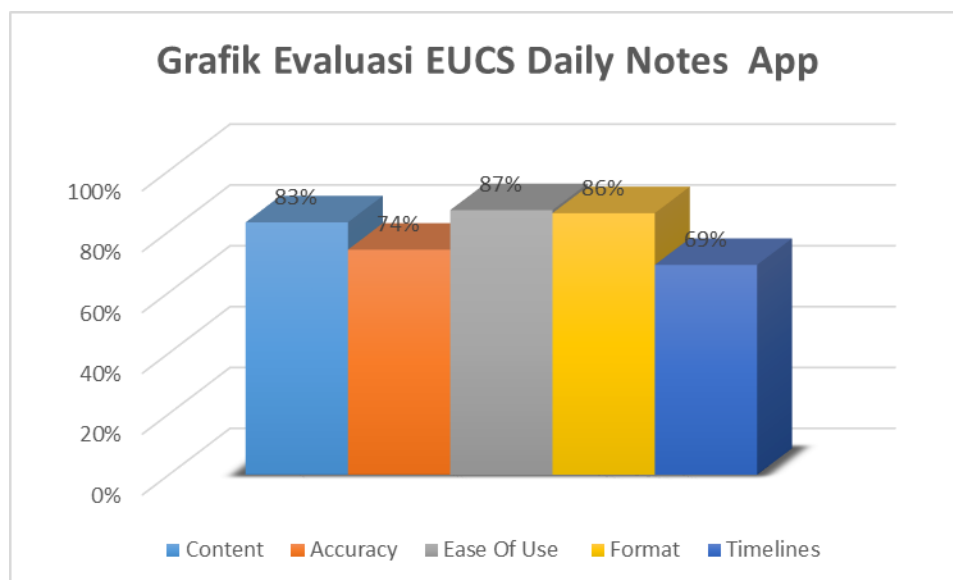
Tabel 3 Hasil Evaluasi Kuesioner EUCS

| No | Pernyataan | Likert | | | | | % Hasil Perhitungan | Indeks |
|----|--|--------|---|----|----|-----|---|--------|
| | | SS | S | CS | TS | STS | | |
| 1 | <i>Content</i> : Informasi tampilan halaman dashboard, <i>daily notes</i> , <i>School Notes</i> , <i>project notes</i> ditampilkan dengan jelas | 8 | 7 | 5 | 0 | 0 | $(8 \times 5) + (7 \times 4) + (5 \times 3) = 83 / 100 = 0,83$ | 83% |
| 2 | <i>Accuracy</i> : Informasi aplikasi ditampilkan sesuai dengan kebutuhan | 6 | 6 | 4 | 4 | 0 | $(6 \times 5) + (6 \times 4) + (4 \times 3) + (4 \times 2) = 74 / 100 = 0,74$ | 74% |

| | | | | | | | | |
|------------------|---|----|---|---|---|---|--|-------|
| 3 | <i>Ease Of Use</i> : Navigasi antar muka Tidak rumit, dapat memahami fungsi secara instan | 10 | 7 | 3 | 0 | 0 | $(10 \times 5) + (7 \times 4) + (3 \times 3) = 87/100 = 0,87$ | 87% |
| 4 | <i>Format</i> : Ikon dan teks pada setiap halaman mudah dikenali | 10 | 6 | 4 | 0 | 0 | $(10 \times 5) + (6 \times 4) + (4 \times 3) = 86/100 = 0,86$ | 86% |
| 5 | <i>TimeLiness</i> : Respons Aplikasi cepat terhadap perintah pengguna serta pembaruan status perangkat cepat | 5 | 5 | 5 | 5 | 0 | $(15 \times 5) + (4 \times 4) + (6 \times 3) + (5 \times 2) = 69/100 = 0,69$ | 69% |
| Rata-Rata Indeks | | | | | | | | 79,8% |

Dari hasil kuesioner dimensi EUCS penelitian maka hasil rata rata indeks adalah 79,8% dimana jika mengacu pada tabel interval skala likert masuk pada kategori “Puas”.

Berikut ini grafik bar evaluasi EUCS setiap dimensi pada gambar 9 :



Gambar 9 Grafik hasil Evaluasi EUCS

4. KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut:

Penelitian ini berhasil merancang dan membangun aplikasi Daily Notes Mobile App berbasis MIT App Inventor dengan memanfaatkan TinyDB sebagai media penyimpanan data lokal.

Penggunaan TinyDB memungkinkan data catatan tersimpan secara permanen di perangkat Android tanpa memerlukan koneksi internet, sehingga aplikasi lebih ringan dan praktis.

Hasil uji coba menunjukkan bahwa dimensi *content* 83%, *accuracy* 74%, *ease of use* 87%, *Format* 86%, *Timeliness* 69%, sehingga hasil akhir rata rata indeks yaitu 79,8% masuk pada kategori “Puas”

Model pengembangan ADDIE terbukti efektif dalam memberikan tahapan sistematis mulai dari analisis kebutuhan, perancangan, pengembangan, implementasi, hingga evaluasi aplikasi.

5.SARAN

Aplikasi dapat dikembangkan lebih lanjut dengan menambahkan fitur sinkronisasi cloud agar catatan dapat diakses lintas perangkat. Integrasi dengan fitur tambahan seperti reminder/notifikasi dan ekspor catatan ke PDF/TXT akan meningkatkan fungsionalitas aplikasi. Penelitian selanjutnya dapat memperluas uji coba dengan melibatkan lebih banyak responden dari berbagai kalangan untuk mendapatkan evaluasi yang lebih komprehensif. serta metode menggunakan pengembangan yang berbeda

DAFTAR PUSTAKA

- [1] N. H. Maulida, "Aplikasi daily note pengingat kegiatan penting," no. July, 2021.
- [2] Technology, "Google Keep: Sejarah Singkat, Fitur, dan Cara Menggunakannya," Telkomsel.
- [3] M. Kaddipujar, J. Rajan, and B. D. Kumbar, "Mobile Application Development Using MIT App Inventor: An Experiment at Raman Research Institute Library," *Bull. AAS*, pp. 1–17, 2022, doi: 10.3847/25c2cfed.d68a2a42.
- [4] E. S. Agus Suharto, "Database dan File Mit App Inventor," Eureka Media Aksara, 2024, p. 77. [Online]. Available: <https://repository.penerbiteureka.com/media/publications/568564-database-dan-file-mit-app-inventor-cloud-1e5203d4.pdf>
- [5] H. Abuhassna, S. Alnawajha, F. Awae, M. A. Bin Mohamed Adnan, and B. I. Edwards, "Synthesizing technology integration within the Addie model for instructional design: A comprehensive systematic literature review," *J. Auton. Intell.*, vol. 7, no. 5, p. 1546, 2024, doi: 10.32629/jai.v7i5.1546.
- [6] Adi Nova Trisetyanto and Handini Arga Damar Rani, "Pengembangan Modul Belajar Robotika Berbasis Internet of Things (IoT) pada Program Studi Pendidikan Informatika, Fakultas Sains dan Teknologi, Universitas Ivet," *Joined J. (Journal Informatics Educ.*, vol. 6, pp. 79–90, 2023.
- [7] J. D. Ticoh *et al.*, "Design of a Greenhouse System for Vegetable Plants Based on the Internet of Things," vol. 08, no. 01, pp. 55–65, 2025.
- [8] R. Samsudin, R. Sulaiman, T. T. Guan, A. M. Yusof, M. Firdaus, and C. Yaacob, "Mobile Application Development Trough ADDIE Model," *Int. J. Acad. Res. Progress. Educ. Dev.*, vol. 10, no. 2, pp. 1017–1027, 2021, doi: 10.6007/IJARPED/v10-i2/10328.
- [9] M. Khairil Asrori and M. B. Triyono, "Development of nahwu learning media based on android," *Utop. y Prax. Latinoam.*, vol. 25, no. Extra1, pp. 225–231, 2020, doi: 10.5281/zenodo.3774623.
- [10] I. Irumas and J. N. Utamajaya, "Penerapan Metode EUCS Untuk Evaluasi Tingkat Kepuasan Pengguna Aplikasi PNM Digi Karyawan," vol. 4, no. 1, pp. 101–108, 2022, doi: 10.47065/josyc.v4i1.2492.
- [11] M. Lionello, F. Aletta, A. Mitchell, and J. Kang, "Introducing a Method for Intervals Correction on Multiple Likert Scales : A Case Study on an Urban Soundscape Data Collection Instrument," vol. 11, no. January, 2021, doi: 10.3389/fpsyg.2020.602831.