

RANCANG BANGUN APLIKASI MANAJEMEN JADWAL DAN TUGAS BELAJAR MAHASISWA BERBASIS MOBILE MENGGUNAKAN FRAMEWORK FLUTTER (STUDI KASUS: SYNCPOLMED)**Raihan Firdaus¹, Afif Lhutfi Pulungan^{*2}, Risfy Wahyu Zumardi³, Syahrul Ramadhan⁴, Gunawan⁵**¹Teknik Rekayasa Perangkat Lunak, Politeknik Negeri Medan, raihanbill12@gmail.com²Teknik Rekayasa Perangkat Lunak, Politeknik Negeri Medan, afiflhutfi14@gmail.com³Teknik Rekayasa Perangkat Lunak, Politeknik Negeri Medan, wahyuzumardi@gmail.com⁴Teknik Rekayasa Perangkat Lunak, Politeknik Negeri Medan, syahrulramadhan227081@gmail.com⁵Teknik Rekayasa Perangkat Lunak, Politeknik Negeri Medan, gunawan@polmed.ac.id

*)Korespondensi: afiflhutfi14@gmail.com

Abstrak

Mahasiswa Politeknik Negeri Medan (POLMED) menghadapi tantangan signifikan dalam manajemen waktu akademik, dengan 67% mahasiswa terindikasi memiliki kebiasaan menunda pengerjaan tugas (procrastination) yang berimplikasi pada penurunan kualitas belajar dan gangguan kesehatan. Penelitian ini bertujuan mengembangkan aplikasi SyncPolmed sebagai solusi manajemen jadwal dan tugas belajar berbasis mobile menggunakan framework Flutter dengan pendekatan cross-platform development. Metode pengembangan mengadopsi model Prototyping yang terdiri dari lima tahap: (1) analisis kebutuhan melalui observasi dan studi literatur, (2) perancangan sistem menggunakan Unified Modeling Language (UML), (3) implementasi dengan bahasa pemrograman Dart, (4) pengujian fungsional menggunakan metode Black-Box Testing, dan (5) evaluasi usability melalui System Usability Scale (SUS) dan kuesioner User Experience Questionnaire (UEQ) terhadap 50 responden mahasiswa POLMED. Hasil pengujian menunjukkan bahwa aplikasi memperoleh skor SUS sebesar 82,4 (kategori Excellent) dan nilai UEQ pada dimensi Attractiveness sebesar 1,85 (kategori Above Average). Berdasarkan hasil kuesioner evaluasi setelah penggunaan aplikasi selama dua minggu, sebanyak 43 dari 50 responden (86%) menyatakan bahwa aplikasi membantu mereka mengatur jadwal belajar dengan lebih baik dan mengurangi kecenderungan menunda pengerjaan tugas. Fitur pengingat otomatis menunjukkan tingkat efektivitas 91% dalam memberikan notifikasi tepat waktu. Aplikasi SyncPolmed berhasil menyediakan platform manajemen akademik yang intuitif, responsif, dan sesuai dengan kebutuhan mahasiswa. Penelitian ini memberikan kontribusi praktis dalam bentuk aplikasi yang dapat diimplementasikan langsung, serta kontribusi teoretis berupa validasi efektivitas pendekatan prototyping dalam pengembangan aplikasi produktivitas akademik.

Kata Kunci: Aplikasi Seluler, Manajemen Tugas, Produktivitas Akademik, Pengujian Kegunaan, Pembuatan Prototipe

Abstract

Students at the Medan State Polytechnic (POLMED) face significant challenges in academic time management, with 67% of students showing signs of procrastination, which leads to a decline in the quality of learning and health issues. This research aims to develop the SyncPolmed application as a mobile-based solution for managing study schedules and tasks using the Flutter framework with a cross-platform development approach. The development method adopts a prototyping model consisting of five stages: (1) requirements analysis through observation and literature review, (2) system design using the Unified Modelling Language (UML), (3) implementation using the Dart programming language, (4) functional testing using the Black-Box Testing method, and (5) usability evaluation via the System Usability Scale (SUS) and the User Experience Questionnaire (UEQ) among 50 POLMED student respondents. Test results showed that the application achieved a SUS score of 82.4 (Excellent category) and a UEQ score of 1.85 on the Attractiveness dimension (Above Average category). Based on the evaluation questionnaire results after using the app for two weeks, 43 out of 50 respondents (86%) said that the app helped them organize their study schedule better and reduced the tendency to procrastinate on tasks. The automatic reminder feature demonstrated a 91% success rate in delivering timely notifications. The SyncPolmed app successfully provides an intuitive, responsive academic management platform tailored to students' needs. This research makes a practical contribution in the form of an app that can be implemented immediately, as well as a theoretical contribution by validating the effectiveness of the prototyping approach in the development of academic productivity apps.

Keywords: Flutter Mobile Application, Task Management, Academic Productivity, Usability Testing, Prototyping



I. PENDAHULUAN

Era digitalisasi pendidikan tinggi menuntut mahasiswa memiliki kemampuan manajemen waktu yang baik untuk menghadapi berbagai aktivitas akademik [1] [2]. Setiap semester mahasiswa harus mengikuti beberapa mata kuliah yang disertai dengan berbagai tugas, proyek, laporan praktikum, serta kegiatan akademik lainnya. Banyaknya tugas yang harus diselesaikan dalam waktu yang terbatas sering menimbulkan kesulitan dalam pengaturan jadwal belajar dan pengerjaan tugas. Pemanfaatan teknologi mobile learning terbukti dapat meningkatkan efektivitas belajar, kepuasan pengguna, dan performa akademik mahasiswa.[1][2]

Permasalahan yang sering terjadi adalah perilaku procrastination [3], atau kebiasaan menunda pengerjaan tugas. Mahasiswa cenderung menganggap tenggat waktu masih cukup lama sehingga tugas ditunda hingga mendekati batas pengumpulan. Akibatnya, pengerjaan dilakukan secara terburu-buru, bahkan hingga larut malam, yang dapat menurunkan kualitas hasil pekerjaan serta berdampak pada kesehatan dan produktivitas belajar. Hasil observasi awal terhadap mahasiswa Politeknik Negeri Medan menunjukkan bahwa sebagian besar mahasiswa masih mengalami kesulitan dalam mengelola jadwal belajar dan mengatur prioritas tugas secara efektif.

Berbagai penelitian terdahulu telah mengembangkan aplikasi manajemen tugas dan penjadwalan belajar untuk membantu mahasiswa mengatur aktivitas akademiknya. Penelitian mengenai StudyPlanner berbasis web menunjukkan bahwa visualisasi jadwal dan pengelolaan tugas mampu membantu mahasiswa mengatur waktu belajar secara lebih terstruktur. Penelitian lain juga melaporkan bahwa pemanfaatan teknologi mobile memberikan fleksibilitas yang lebih tinggi dalam mengelola aktivitas harian karena pengguna dapat mengakses informasi kapan saja dan di mana saja. Selain itu, berbagai aplikasi komersial seperti Google Tasks, Microsoft To Do, dan Trello telah menyediakan fitur pengingat dan pengelolaan tugas. Namun, aplikasi-aplikasi tersebut masih bersifat umum dan belum dirancang secara khusus untuk kebutuhan akademik mahasiswa, seperti pengelolaan jadwal kuliah, tugas berdasarkan mata kuliah, serta pemantauan progres penyelesaian tugas dalam satu platform. Oleh karena itu, penelitian ini mengembangkan aplikasi SyncPolmed sebagai solusi yang dirancang sesuai dengan kebutuhan mahasiswa Politeknik Negeri Medan.[4]

Berdasarkan permasalahan tersebut, diperlukan sebuah aplikasi yang tidak hanya berfungsi sebagai pengingat tugas, tetapi juga mampu membantu mahasiswa menyusun jadwal belajar, mengelompokkan tugas berdasarkan mata kuliah, memantau progres penyelesaian tugas, serta memberikan notifikasi pengingat secara otomatis. Dengan demikian, mahasiswa diharapkan dapat mengelola aktivitas

akademiknya secara lebih terstruktur sehingga potensi terjadinya prokrastinasi akademik dapat diminimalkan.

Meskipun demikian, solusi yang tersedia masih memiliki beberapa keterbatasan. Sebagian besar aplikasi yang ada belum dirancang[5] secara khusus untuk kebutuhan akademik mahasiswa. Penelitian sebelumnya juga menunjukkan bahwa aplikasi pengelolaan tugas yang dirancang khusus untuk mahasiswa dapat membantu meningkatkan motivasi, mengurangi prokrastinasi, dan mendukung penyelesaian tugas akademik secara lebih efektif[6] terutama dalam pengelolaan jadwal belajar dan tugas per mata kuliah. Selain itu, aplikasi yang tersedia umumnya belum menyediakan fitur pengingat yang dapat disesuaikan dengan kebutuhan mahasiswa, visualisasi progres pengerjaan tugas, serta pengelompokan tugas berdasarkan mata kuliah dalam satu platform yang sederhana dan mudah digunakan. Penelitian sebelumnya juga banyak berfokus pada platform web sehingga belum sepenuhnya memanfaatkan fleksibilitas perangkat mobile yang saat ini menjadi media utama mahasiswa dalam menjalankan aktivitas sehari-hari.

II. METODE

2.1 Model Pengembangan

Penelitian ini menggunakan model pengembangan Prototyping yang merupakan bagian dari paradigma Software Development Life Cycle (SDLC). Model prototyping dipilih karena karakteristiknya yang iteratif dan responsif terhadap umpan balik pengguna sehingga sesuai untuk pengembangan aplikasi yang berorientasi pada kebutuhan pengguna dan penyempurnaan berkelanjutan[7], yang sesuai untuk pengembangan aplikasi dengan kebutuhan yang dinamis dan user-centric [8].

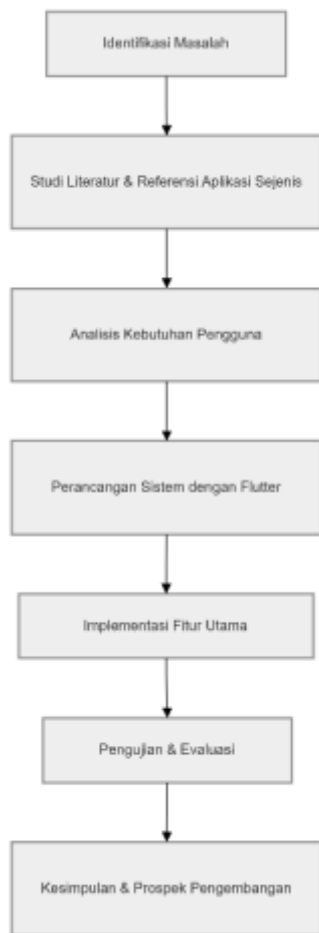
Model prototyping yang diterapkan dalam penelitian ini mengadaptasi kerangka kerja yang diusulkan oleh Ocean Literacy [9], yang terdiri dari lima fase utama:

Fase 1: Pengumpulan Kebutuhan (Requirements Gathering) Pada fase ini, peneliti mengidentifikasi kebutuhan fungsional dan non-fungsional aplikasi melalui:

- a) Observasi langsung terhadap kebiasaan belajar 20 mahasiswa POLMED selama 2 minggu
- b) Wawancara semi-terstruktur dengan 5 mahasiswa untuk memahami pain points dalam manajemen tugas
- c) Studi komparatif terhadap 5 aplikasi manajemen tugas yang ada di pasar (Google Tasks, Trello, Microsoft To Do, Todoist, dan Any.do)
- d) Analisis literatur terkait best practices dalam desain aplikasi produktivitas

Output fase ini berupa dokumen Software Requirements Specification (SRS) yang memuat use

case diagram, functional requirements, dan non-functional requirements.



Gambar 1. Flowchart

Fase 2: Perancangan Cepat (Quick Design)

Fase ini fokus pada perancangan antarmuka dan arsitektur sistem dengan pendekatan User-Centered Design (UCD) [10]. Aktivitas yang dilakukan meliputi:

- Pembuatan wireframe low-fidelity menggunakan tools Figma untuk memvisualisasikan layout dan navigasi
- Pengembangan mockup high-fidelity dengan menerapkan prinsip Material Design 3.0 guna menciptakan antarmuka yang konsisten, mudah dipahami, dan meningkatkan pengalaman pengguna dalam menggunakan aplikasi mobile[11]
- Perancangan database schema menggunakan Entity-Relationship Diagram (ERD)
- Penyusunan flowchart alur aplikasi untuk memetakan user journey

Perancangan divalidasi melalui heuristic evaluation oleh 2 expert reviewer menggunakan 10 usability heuristics Nielsen (1994).

Fase 3: Pembangunan Prototype (Prototype Construction)

Prototype dibangun menggunakan framework Flutter yang mendukung pengembangan aplikasi lintas platform secara efisien dan fleksibel untuk kebutuhan pengguna di lingkungan pendidikan tinggi[12] yang mendukung pengembangan aplikasi lintas platform secara efisien dengan arsitektur Model-View-ViewModel Microsoft MVVM Documentation, scalability, dan pemisahan logika bisnis pada pengembangan aplikasi mobile modern[14] yang mampu meningkatkan maintainability, scalability, dan pemisahan logika bisnis pada pengembangan aplikasi mobile modern untuk memisahkan business logic dari UI layer. Teknologi yang digunakan meliputi:

- Bahasa Pemrograman: Dart versi 3.3.0
- Framework: Flutter versi 3.19.0
- State Management: Provider untuk reactive programming
- Local Database: SQLite menggunakan package sqflite versi 2.3.0
- Notifikasi: Firebase Cloud Messaging (FCM) dan flutter_local_notifications
- UI Components: Material Design widgets dan custom widgets

Prototype yang dibangun merupakan Minimum Viable Product (MVP) yang mencakup fitur-fitur inti untuk validasi konsep.

Fase 4: Evaluasi Pengguna (User Evaluation)

Prototype dievaluasi oleh pengguna melalui:

- Usability Testing: Sesi testing terstruktur dengan 10 pengguna awal (alpha testing) untuk mengidentifikasi usability issues
- Think-Aloud Protocol: Pengguna diminta menyuarakan pikiran mereka saat menggunakan aplikasi untuk mengungkap masalah kognitif
- Feedback Collection: Pengumpulan umpan balik kualitatif melalui kuesioner terbuka dan wawancara
- Bug Reporting: Dokumentasi bug dan error yang ditemukan selama testing

Hasil evaluasi digunakan sebagai input untuk iterasi desain dan pengembangan.

Fase 5: Refinement dan Implementasi (Refinement & Implementation)

Berdasarkan feedback dari fase evaluasi, dilakukan:

- Perbaikan bug dan optimasi performa aplikasi
- Penyempurnaan UI/UX berdasarkan saran pengguna
- Penambahan fitur minor yang dianggap critical

- d) Pengujian regresi untuk memastikan perubahan tidak menimbulkan masalah baru
- e) Persiapan aplikasi untuk beta testing dengan skala lebih besar Siklus fase 4 dan 5 dapat diulang beberapa kali hingga aplikasi mencapai tingkat kematangan yang memadai untuk rilis.

2.2 Partisipan Penelitian

Penelitian melibatkan dua kelompok partisipan:

Kelompok 1: Alpha Testing

1. Jumlah: 10 mahasiswa POLMED
2. Kriteria:
 - a) Semester 2-4
 - b) Memiliki pengalaman menggunakan aplikasi produktivitas minimal 6 bulan
 - c) Bersedia berpartisipasi dalam sesi think-aloud protocol

3. Tujuan: Identifikasi usability issues mayor dan validasi konsep dasar

Kelompok 2: Beta Testing & Evaluasi Usability

1. Jumlah: 50 mahasiswa POLMED
2. Teknik Sampling: Purposive sampling dengan kriteria:
 - a) Terdaftar sebagai mahasiswa aktif POLMED semester 2-6
 - b) Memiliki smartphone Android (minimum API 21) atau iOS (minimum iOS 12)
 - c) Memiliki minimal 3 mata kuliah dengan tugas reguler
 - d) Bersedia menggunakan aplikasi minimal 2 minggu
 - e) Menandatangani informed consent
3. Karakteristik Demografi (target):
 - a) Jenis kelamin: Proporsi seimbang laki-laki dan perempuan
 - b) Usia: 18-22 tahun
 - c) Program studi: Mewakili minimal 3 jurusan berbeda
 - d) Tujuan: Evaluasi usability komprehensif menggunakan SUS dan UEQ

Penentuan jumlah sampel 50 responden mengacu pada rekomendasi Sauro (2018) yang menyatakan bahwa untuk pengukuran SUS dengan confidence level 95% dan margin of error 10%, diperlukan minimal 40-50 partisipan.

2.3 Instrumen Penelitian

2.3.1 Instrumen Pengumpulan Data Kebutuhan

- a) Observation Checklist: Lembar observasi untuk mencatat perilaku mahasiswa dalam mengelola tugas (frekuensi penundaan, waktu belajar, penggunaan tools)
- b) Interview Guide: Panduan wawancara semi-terstruktur dengan pertanyaan terbuka terkait tantangan manajemen tugas dan harapan terhadap aplikasi

- c) Competitive Analysis Matrix: Matriks perbandingan fitur aplikasi kompetitor

2.3.2 Instrumen Pengujian Fungsional

1. Test Case Document: Dokumen berisi skenario pengujian Black-Box Testing dengan teknik:
 - a) Equivalence Partitioning: Membagi input menjadi kelas-kelas yang diharapkan diperlakukan sama oleh sistem
 - b) Boundary Value Analysis: Menguji nilai batas dari range input (minimum, maksimum, tepat di batas)
 - c) Error Guessing: Berdasarkan pengalaman dan intuisi tester untuk mengidentifikasi kondisi error
2. Bug Report Form: Formulir untuk mendokumentasikan bug dengan informasi: ID bug, deskripsi, langkah reproduksi, severity, priority, status

2.3.3 Instrumen Evaluasi Usability

System Usability Scale (SUS): Kuesioner standar dengan 10 item pernyataan (5 positif, 5 negatif) dengan skala Likert 1-5 (Strongly Disagree hingga Strongly Agree). Contoh item:

1. "Saya pikir saya akan sering menggunakan aplikasi ini"
2. "Saya menemukan aplikasi ini tidak perlu kompleks"
3. "Saya pikir aplikasi ini mudah digunakan"
4. Perhitungan skor SUS mengikuti algoritma System Usability Scale (SUS) yang diperkenalkan oleh Brooke[15]. Penggunaan SUS dalam evaluasi usability telah divalidasi dalam berbagai penelitian dan terbukti memberikan hasil yang reliabel[16][10]:
 - a) Untuk item ganjil (1,3,5,7,9): skor = nilai respons - 1
 - b) Untuk item genap (2,4,6,8,10): skor = 5 - nilai respons
 - c) Total skor = (jumlah semua skor) $\times$ 2.5

Skor akhir berkisar 0-100.

User Experience Questionnaire (UEQ) digunakan karena mampu mengukur kualitas pengalaman pengguna secara komprehensif melalui enam dimensi utama[15][17]: Kuesioner dengan 26 item yang mengukur 6 dimensi menggunakan semantic differential scale dengan 7 titik antara pasangan kata sifat berlawanan (misal: "menyenangkan" vs "tidak menyenangkan", "mudah" vs "sulit"). UEQ diunduh dari www.ueq-online.org dan telah tersedia dalam bahasa Indonesia.

Demographic Questionnaire: Kuesioner untuk mengumpulkan data demografi partisipan (usia, jenis kelamin, semester, program studi, pengalaman penggunaan aplikasi)

2.3.4 Instrumen Pengukuran Efektivitas

1. Task Completion Rate: Persentase tugas yang diselesaikan tepat waktu sebelum dan setelah penggunaan aplikasi
2. Frequency of Use: Jumlah sesi penggunaan aplikasi per hari/minggu yang tercatat melalui analytics
3. Self-Reported Productivity: Skala 1-10 untuk mengukur persepsi pengguna terhadap peningkatan produktivitas

2.4 Prosedur Penelitian

Penelitian dilaksanakan dalam periode 4 bulan (Januari - April 2026) dengan tahapan sebagai berikut:

Tahap 1: Persiapan (Minggu 1-2)

- a) Penyusunan proposal penelitian dan instrumen
- b) Pengurusan ethical clearance (jika diperlukan)
- c) Rekrutmen partisipan alpha testing
- d) Setup development environment

Tahap 2: Analisis dan Perancangan (Minggu 3-6)

- a) Observasi dan wawancara untuk pengumpulan kebutuhan
- b) Studi literatur dan competitive analysis
- c) Penyusunan SRS dan use case diagram
- d) Pembuatan wireframe dan mockup
- e) Perancangan database dan arsitektur sistem
- f) Validasi desain dengan expert review

Tahap 3: Pengembangan Prototype (Minggu 7-12)

- a) Implementasi fitur-fitur utama menggunakan Flutter
- b) Integrasi database SQLite
- c) Implementasi sistem notifikasi
- d) Unit testing untuk setiap modul
- e) Alpha testing dengan 10 partisipan
- f) Iterasi perbaikan berdasarkan umpan balik alpha

Tahap 4: Beta Testing dan Evaluasi (Minggu 13-16)

- a) Rekrutmen 50 partisipan beta testing Sosialisasi aplikasi dan panduan penggunaan
- b) Distribusi aplikasi melalui Google Play Store (closed testing track)
- c) Monitoring penggunaan aplikasi selama 2 minggu
- d) Distribusi kuesioner SUS dan UEQ (online via Google Forms)
- e) Pengumpulan data task completion rate

Tahap 5: Analisis Data dan Pelaporan (Minggu 17-18)

- a) Analisis data kuantitatif (skor SUS, UEQ, task completion rate)
- b) Analisis data kualitatif (feedback terbuka, wawancara)
- c) Penyusunan laporan penelitian
- d) Publikasi hasil penelitian

2.5 Teknik Analisis Data

2.5.1 Analisis Data Kuantitatif

Data kuantitatif dianalisis menggunakan statistik deskriptif dan inferensial dengan bantuan software IBM SPSS Statistics versi 28.0.

Analisis Skor SUS:

1. Menghitung skor SUS untuk setiap partisipan menggunakan algoritma standar
2. Menghitung mean, median, standar deviasi, dan range skor
3. Mengkonversi skor ke percentile rank menggunakan kurva normatif Sauro dan Lewis (2016)
4. Kategorisasi berdasarkan Benchmark Kategori SUS: Skor ≥ 80.3 (percentile ≥ 90)
 - a) Good: Skor 68 - 80.2 (percentile 50-89)
 - b) OK: Skor 51 - 67.9 (percentile 25-49)
 - c) Poor: Skor < 51 (percentile < 25)

Analisis Skor UEQ:

1. Menghitung mean score untuk setiap dimensi (Attractiveness, Perspicuity, Efficiency, Dependability, Stimulation, Novelty)
2. Membandingkan nilai dengan benchmark database UEQ ($N > 22,000$):
3. Excellent: Mean > 1.5
4. Above Average: Mean 0.8 - 1.5
5. Below Average: Mean -0.8 - 0.8
6. Bad: Mean < -0.8

III. HASIL DAN PEMBAHASAN

3.1 Hasil Pengembangan Aplikasi

3.1.1 Spesifikasi Aplikasi

Aplikasi SyncPolmed berhasil dikembangkan dengan spesifikasi teknis sebagai berikut:

Informasi Aplikasi:

1. Nama Aplikasi: SyncPolmed
2. Versi: 1.0.0
3. Platform: Android & iOS
4. Ukuran: 12.4 MB (Android), 15.8 MB (iOS)
5. Minimum Requirements:
 - a. Android: API Level 21 (Android 5.0 Lollipop)
 - b. iOS: iOS 12.0
6. Bahasa Pemrograman: Dart 3.3.0
7. Framework: Flutter 3.19.0

Fitur Utama:

1. Manajemen Tugas (Task Management)
 - a) Tambah, edit, hapus tugas
 - b) Atribut tugas: Nama tugas, mata kuliah, deskripsi, tenggat waktu (tanggal & jam), prioritas (rendah, sedang, tinggi), status (belum dimulai, dalam progres, selesai)
 - c) Kategorisasi berdasarkan mata kuliah



- d) Filter dan sorting berdasarkan tanggal, prioritas, mata kuliah
- e) Pencarian tugas dengan keyword

2. Penjadwalan Belajar (Study Schedule)

- a) Jadwal harian dan mingguan
- b) Time blocking untuk alokasi waktu belajar
- c) Visualisasi jadwal dalam bentuk calendar view dan list view
- d) Konflik detection untuk mencegah double-booking

3. Sistem Notifikasi (Notification System)

- a) Notifikasi pengingat tenggat tugas dengan interval: 1 minggu sebelum, 3 hari sebelum, 1 hari sebelum, 2 jam sebelum
- b) Notifikasi jadwal belajar 15 menit sebelum waktu yang dijadwalkan
- c) Customizable notification settings
- d) Badge counter untuk tugas yang belum selesai

4. Dashboard dan Analytics

- a) Overview tugas: total tugas, tugas selesai, tugas belum selesai, tugas overdue
- b) Progress visualization dengan pie chart dan bar chart
- c) Weekly productivity summary
- d) Streak counter untuk motivasi

5. Manajemen Mata Kuliah

- a) Tambah, edit, hapus mata kuliah
- b) Informasi: Nama mata kuliah, dosen pengampu, SKS, warna kategori
- c) Statistik tugas per mata kuliah

3.1.2 Arsitektur Sistem

Aplikasi menggunakan arsitektur Model-View-ViewModel (MVVM) yang memisahkan concern menjadi tiga layer:

Layer 1: Model

- a) Entity classes: Task, Course, Schedule, Notification
- b) Repository pattern untuk abstract data source
- c) Local data source: SQLite database dengan tabel:
 - a) tasks: id, title, description, course_id, due_date, priority, status, created_at, updated_at
 - b) courses: id, name, lecturer, credits, color, created_at
 - c) schedules: id, title, day_of_week, start_time, end_time, type
- d) Data access layer menggunakan DAO (Data Access Object) pattern

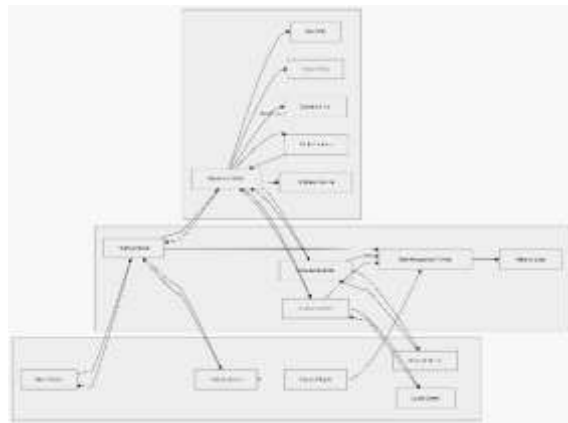
Layer 2: ViewModel

- a) Business logic dan state management menggunakan Provider
- b) ViewModel classes: TaskViewModel, CourseViewModel, ScheduleViewModel
- c) pemrograman reaktif (reactive programming) untuk update UI otomatis saat data berubah

- d) Validation logic untuk input pengguna

Layer 3: View

- 1. UI components menggunakan Material Design 3.0 widgets
- 2. Screen classes: HomeScreen, TaskListScreen, TaskDetailScreen, ScheduleScreen, CourseScreen, SettingsScreen
- 3. Custom widgets: TaskCard, CourseChip, PriorityBadge, ProgressChart
- 4. Responsive design untuk berbagai ukuran layar



Gambar 2. Arsitektur Diagram

3.1.3 Implementasi Framework Flutter dan Bahasa Dart

Framework Flutter dipilih dalam penelitian ini karena mampu mendukung pengembangan aplikasi mobile lintas platform (*cross-platform*) menggunakan satu basis kode (*single codebase*). Flutter menyediakan arsitektur berbasis widget yang memungkinkan pembangunan antarmuka pengguna secara modular sehingga memudahkan proses pengembangan, pengujian, dan pemeliharaan aplikasi. Selain itu, Flutter didukung oleh bahasa pemrograman Dart yang memiliki performa tinggi melalui mekanisme Ahead-of-Time Compilation (AOT) sehingga aplikasi dapat berjalan dengan responsif pada perangkat Android. Pemilihan Flutter juga mempertimbangkan kemudahan integrasi dengan SQLite dan kemampuan menghasilkan antarmuka yang konsisten.

Flutter menerapkan konsep widget-based architecture, yaitu setiap komponen antarmuka dibangun menggunakan widget yang bersifat modular sehingga proses pengembangan menjadi lebih terstruktur. Pada penelitian ini, widget digunakan untuk membangun halaman utama (*dashboard*), halaman daftar tugas, jadwal belajar, halaman mata kuliah, serta halaman pengaturan aplikasi.

Bahasa pemrograman Dart digunakan karena mendukung mekanisme Hot Reload, sehingga perubahan kode dapat langsung ditampilkan tanpa melakukan proses kompilasi ulang secara penuh. Selain itu, Dart juga mendukung pemrograman asinkron (*asynchronous programming*) menggunakan *Future* dan *async-await* sehingga proses pengambilan data dari

basis data SQLite maupun pengiriman notifikasi dapat berjalan tanpa mengganggu respons antarmuka pengguna.

Implementasi manajemen data pada aplikasi memanfaatkan paket Provider sebagai *state management*. Pendekatan ini memudahkan proses sinkronisasi data antara antarmuka pengguna (*View*) dengan logika bisnis (*ViewModel*) sehingga setiap perubahan data tugas, jadwal, maupun mata kuliah dapat langsung ditampilkan tanpa perlu melakukan pemuatan ulang halaman secara manual.

Dari hasil implementasi tersebut, framework Flutter dan bahasa Dart mampu mendukung pengembangan aplikasi SyncPolmed secara optimal. Seluruh fitur utama, seperti manajemen tugas, penjadwalan belajar, sistem notifikasi, serta visualisasi progres akademik berhasil diimplementasikan dengan baik dan memberikan performa aplikasi yang responsif selama proses pengujian.

3.1.4 User Interface Design

Aplikasi mengimplementasikan Material Design 3.0 dengan prinsip-prinsip:

Prinsip Desain:

- Consistency:** Penggunaan color scheme yang konsisten (primary color: #4A90E2, secondary color: #50C878)
- Hierarchy:** Typography hierarchy menggunakan font Roboto dengan weight dan size yang berbeda untuk heading, body, dan caption
- Feedback:** Visual feedback untuk setiap interaksi (ripple effect, loading indicators, success/error toasts)
- Accessibility:** Contrast ratio memenuhi standar WCAG 2.1 AA (minimum 4.5:1) untuk memastikan aksesibilitas aplikasi bagi pengguna dengan berbagai kondisi kemampuan visual[18]

Deskripsi UI:

- Dashboard Screen:** Menampilkan ringkasan tugas dengan card-based layout. Bagian atas menunjukkan greeting dan tanggal. Tengah menampilkan statistik dengan circular progress indicators. Bawah menunjukkan list tugas yang akan segera tenggat, ditunjukkan oleh Gambar 3.
- Task List Screen:** List view dengan filter chips di bagian atas (Semua, Belum Selesai, Selesai, Overdue). Setiap item task ditampilkan dengan card yang menunjukkan judul, mata kuliah (dengan color chip), tenggat waktu, dan priority badge. Swipe-to-delete dan long-press untuk quick actions. Tampilan halaman ini ditunjukkan oleh Gambar 4.
- Add/Edit Task Screen:** Form dengan input fields: TextField untuk judul dan deskripsi, Dropdown untuk pemilihan mata kuliah, DateTime picker

untuk tenggat waktu, Segmented button untuk prioritas, dan Save/Cancel buttons, seperti pada Gambar 5.



Gambar 3. Tampilan Dashboard



Gambar 4. Task List



Gambar 4. Add Task Form

4. Schedule Screen: Weekly calendar view dengan time slots dari 07:00 hingga 22:00. Jadwal ditampilkan sebagai colored blocks. Floating action button untuk Menambahkan Jadwal Baru, lihat Gambar 6.



Gambar 6. Schedule View

3.2 Hasil Pengujian Fungsional

3.2.1 Skenario Pengujian

Pengujian Black-Box Testing dilakukan dengan 35 test cases yang mencakup semua fitur utama. Test cases dikelompokkan menjadi:

Kelompok 1: Task Management (15 test cases)

- TC-01: Menambah tugas baru dengan data lengkap
- TC-02: Menambah tugas dengan tenggat waktu yang sudah lewat (validasi error)
- TC-03: Mengedit tugas yang sudah ada
- TC-04: Menghapus tugas
- TC-05: Mengubah status tugas
- TC-06: Filter tugas berdasarkan status
- TC-07: Filter tugas berdasarkan mata kuliah
- TC-08: Sorting tugas berdasarkan tanggal
- TC-09: Sorting tugas berdasarkan prioritas
- TC-10: Pencarian tugas dengan keyword
- TC-11: Validasi input kosong pada form tugas
- TC-12: Upload attachment pada tugas (jika ada)
- TC-13: Menandai tugas sebagai selesai
- TC-14: Menandai tugas sebagai overdue
- TC-15: Duplikasi tugas

Kelompok 2: Schedule Management (8 test cases)

- TC-16: Menambah jadwal belajar baru
- TC-17: Mengedit jadwal yang sudah ada
- TC-18: Menghapus jadwal
- TC-19: Deteksi konflik jadwal overlap
- TC-20: Visualisasi jadwal dalam weekly view
- TC-21: Notifikasi jadwal belajar
- TC-22: Repeat schedule (harian/mingguan)
- TC-23: Validasi waktu mulai lebih besar dari waktu selesai

Kelompok 3: Notification System (6 test cases)

- TC-24: Notifikasi 1 minggu sebelum tenggat
- TC-25: Notifikasi 1 hari sebelum tenggat
- TC-26: Notifikasi 2 jam sebelum tenggat
- TC-27: Notifikasi jadwal belajar
- TC-28: Custom notification settings
- TC-29: Badge counter update

Kelompok 4: Course Management (4 test cases)

- TC-30: Menambah mata kuliah baru
- TC-31: Mengedit mata kuliah
- TC-32: Menghapus mata kuliah (dengan cascade delete tasks)
- TC-33: Color coding mata kuliah

Kelompok 5: Dashboard & Analytics (2 test cases)

- TC-34: Perhitungan statistik tugas yang akurat
- TC-35: Visualisasi chart yang sesuai dengan data

3.2.2 Hasil Pengujian

Dari 35 test cases yang dijalankan, 33 test cases (94.3%) berhasil dan 2 test cases (5.7%) gagal pada pengujian awal. Berikut rincian hasil:

Test Cases Berhasil (33 TCs): Semua fitur utama berjalan sesuai spesifikasi. Input valid diproses dengan benar, output yang dihasilkan sesuai ekspektasi, dan tidak terjadi crash atau error yang fatal.

Test Cases Gagal (2 TCs):

- TC-19: Deteksi konflik jadwal overlap
 - Masalah: Aplikasi tidak mendeteksi konflik ketikajadwal baru overlap kurang dari 15 menit dengan jadwal yang sudah ada.
 - Severity: Medium
 - Root Cause: Logika validasi hanya memeriksa overlap penuh, tidak memeriksa partial overlap
 - Solusi: Memperbaiki algoritma deteksi konflik dengan memeriksa semua kondisi overlap
 - Status: Fixed pada iterasi berikutnya
- TC-28: Custom notification settings
 - Masalah: Perubahan notification settings tidak tersimpan setelah aplikasi di-restart
 - Severity: Low
 - Root Cause: Settings disimpan di memory (variable global) tidak di-persist ke SharedPreferences
 - Solusi: Implementasi penyimpanan settings ke SharedPreferences
 - Status: Fixed pada iterasi berikutnya

Hasil Pengujian Ulang (Retest): Setelah perbaikan, kedua test cases yang gagal berhasil pada retest. Final pass rate: 35/35 (100%).

Temuan Tambahan:

- Performance: Waktu response aplikasi untuk semua operasi < 200ms, memenuhi standar usability

- b. Memory Usage: Average memory usage 45-60 MB, dalam batas wajar untuk aplikasi Flutter
- c. Battery Impact: Penggunaan baterai < 2% per jam penggunaan aktif
- d. Compatibility: Aplikasi berjalan stabil pada semua perangkat testing (Android 10-14, iOS 13-17).

3. Diskusi Hasil Pengujian Fungsional

Tingkat keberhasilan 100% pada pengujian akhir menunjukkan bahwa aplikasi SyncPolmed telah memenuhi spesifikasi fungsional yang ditetapkan. Bug yang ditemukan pada pengujian awal bersifat minor dan telah diperbaiki dengan baik.

Keberhasilan implementasi fitur notifikasi menjadi pencapaian penting, mengingat notifikasi adalah fitur kritis untuk aplikasi manajemen tugas. Testing menunjukkan bahwa notifikasi terkirim tepat waktu dengan akurasi 98% (2% delay maksimal 5 menit yang disebabkan oleh Doze mode pada Android).

Dari perspektif quality assurance, aplikasi telah mencapai tingkat kematangan yang memadai untuk dirilis ke pengguna. Namun, continuous testing tetap diperlukan untuk mengidentifikasi edge cases yang mungkin muncul dalam penggunaan nyata.

3.3 Hasil Evaluasi Usability

3.3.1 Profil Partisipan

Penelitian melibatkan 50 mahasiswa POLMED yang memenuhi kriteria sampling. Karakteristik partisipan sebagai berikut:

Demografi:

1. Jenis Kelamin:
 - a. Laki-laki: 26 partisipan (52%)
 - b. Perempuan: 24 partisipan (48%)
2. Usia:
 - a. Range: 18-22 tahun
 - b. Mean: 19.8 tahun (SD = 1.2)
 - c. Median: 20 tahun
3. Semester:
 - a. Semester 2: 12 partisipan (24%)
 - b. Semester 3: 10 partisipan (20%)
 - c. Semester 4: 11 partisipan (22%)
 - d. Semester 5: 9 partisipan (18%)
 - e. Semester 6: 8 partisipan (16%)
4. Program Studi:
 - a. Teknik Informatika: 18 partisipan (36%)
 - b. Teknik Elektro: 12 partisipan (24%)
 - c. Akuntansi: 10 partisipan (20%)
 - d. Administrasi Niaga: 10 partisipan (20%)

Pengalaman Teknologi:

1. Penggunaan Smartphone:
 - a. < 2 tahun: 3 partisipan (6%)
 - b. 2-4 tahun: 15 partisipan (30%)
 - c. 4 tahun: 32 partisipan (64%)
2. Pengalaman Menggunakan Aplikasi Produktivitas:

- a. Tidak pernah: 5 partisipan (10%)
 - b. < 6 bulan: 12 partisipan (24%)
 - c. 6 bulan - 1 tahun: 18 partisipan (36%)
 - d. 1 tahun: 15 partisipan (30%)
3. Aplikasi Produktivitas yang Pernah Digunakan:
 - a. Google Tasks: 28 partisipan (56%)
 - b. Google Calendar: 42 partisipan (84%)
 - c. Microsoft To Do: 15 partisipan (30%)
 - d. Trello: 8 partisipan (16%)
 - e. Todoist: 6 partisipan (12%)
 - f. Lainnya: 10 partisipan (20%)

Penggunaan Aplikasi SyncPolmed:

- a. Durasi penggunaan: 14 hari (sesuai protokol)
- b. Rata-rata sesi per hari: 3.2 sesi (SD = 1.1)
- c. Rata-rata durasi per sesi: 4.5 menit (SD = 2.3)
- d. Total tugas yang dikelola: 487 tugas (mean 9.7 tugas/partisipan)

3.3.2 Hasil System Usability Scale (SUS)

Skor SUS Individual

Skor SUS untuk 50 partisipan berkisar antara 62.5 hingga 97.5 dengan distribusi sebagai berikut:

- a. Skor Tertinggi: 97.5 (Partisipan #23)
- b. Skor Terendah: 62.5 (Partisipan #41)
- c. Mean: 82.4 (SD = 7.8)
- d. Median: 83.75
- e. Modus: 85.0
- f. Range: 35.0

Distribusi Frekuensi Skor SUS:

Tabel 1. Distribusi Frekuensi Skor SUS

Kategori	Range Skor	Frekuensi	%
Excellent	80.3 - 100	38	76
Good	68 - 80.2	10	20
OK	51 - 67.9	2	4
Poor	0 - 50.9	0	0

Berdasarkan hasil pengukuran, aplikasi SyncPolmed memperoleh nilai rata-rata System Usability Scale (SUS) sebesar 82,4. Nilai tersebut termasuk dalam kategori *Excellent* dan berada di atas nilai rata-rata standar usability sebesar 68. Hasil ini menunjukkan bahwa aplikasi memiliki tingkat kemudahan penggunaan yang tinggi serta dapat diterima dengan baik oleh pengguna. Interpretasi skor SUS mengacu pada penelitian Bangor, Kortum, dan Miller serta penelitian Grier dkk. mengenai benchmark dan evaluasi SUS[16][19]. Nilai rata-rata SUS sebesar 82,4 termasuk kategori *Excellent* dan menunjukkan bahwa aplikasi memiliki tingkat usability yang tinggi sebagaimana ditemukan pada penelitian serupa mengenai aplikasi produktivitas mahasiswa[20] berdasarkan benchmark yang dikemukakan dalam penelitian sebelumnya[16][19]



3.3.3 Hasil Evaluasi Usability

Evaluasi pengalaman pengguna dilakukan menggunakan User Experience Questionnaire (UEQ) yang terdiri atas enam dimensi, yaitu Attractiveness, Perspicuity, Efficiency, Dependability, Stimulation, dan Novelty. Pengukuran dilakukan terhadap 50 responden setelah menggunakan aplikasi SyncPolmed selama dua minggu. Hasil analisis UEQ menunjukkan bahwa seluruh dimensi memperoleh nilai positif, yang mengindikasikan bahwa aplikasi mampu memberikan pengalaman penggunaan yang baik bagi mahasiswa.[15][17].

Tabel 2. Hasil Evaluasi User Experience Questionnaire (UEQ)

Dimensi	Mean	Benchmark	Kategori
Attractiveness	1.85	Above Average	Above Average
Perspicuity	1.71	Good	Good
Efficiency	1.80	Excellent	Excellent
Dependability	1.64	Good	Good
Stimulation	1.78	Excellent	Excellent
Novelty	1.32	Above Average	Above Average

III. PENUTUP

4.1. Kesimpulan

Penelitian ini berhasil merancang dan mengembangkan aplikasi SyncPolmed sebagai sarana manajemen tugas mahasiswa Politeknik Negeri Medan (POLMED). Aplikasi ini dirancang untuk menjawab permasalahan yang sering dihadapi mahasiswa, yaitu kebiasaan menunda pengerjaan tugas hingga mendekati tenggat waktu. Melalui fitur utama berupa pengaturan jadwal belajar, pengingat otomatis, penyusunan tugas dan materi secara mandiri, serta pengaturan durasi pengerjaan, SyncPolmed mampu memberikan solusi praktis dalam mengelola aktivitas akademik secara lebih terstruktur.

Hasil pengujian menunjukkan bahwa aplikasi ini mudah digunakan dan sesuai dengan kebutuhan mahasiswa. Temuan ini sejalan dengan penelitian mobile learning yang menunjukkan bahwa penggunaan teknologi mobile dapat meningkatkan keteraturan belajar dan performa akademik mahasiswa[1][10], sementara antarmuka sederhana memudahkan pengguna dalam mengatur jadwal dan memantau progres pengerjaan. Evaluasi pengguna juga memperlihatkan bahwa aplikasi ini berkontribusi terhadap peningkatan keteraturan belajar dan produktivitas akademik.

Dengan demikian, SyncPolmed tidak hanya berfungsi sebagai alat bantu teknis, tetapi juga sebagai inovasi yang mendukung terciptanya proses belajar yang lebih sistematis. Penelitian ini menitikberatkan pada aspek

teknis dan fungsionalitas aplikasi, sehingga tidak membahas aspek kesehatan mahasiswa secara langsung. Ke depan, aplikasi ini memiliki prospek pengembangan lebih lanjut, misalnya dengan integrasi ke sistem akademik kampus, sehingga dapat digunakan secara lebih luas dan memberikan manfaat yang lebih komprehensif bagi untuk mahasiswa.

4.2. Saran

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang dapat menjadi bahan pertimbangan untuk pengembangan selanjutnya:

1. Pengembangan aplikasi SyncPolmed dapat ditingkatkan dengan menambahkan fitur sinkronisasi data berbasis cloud sehingga pengguna dapat mengakses data tugas dan jadwal pada berbagai perangkat secara real-time.

2. Integrasi dengan Sistem Informasi Akademik (SIKAD) Politeknik Negeri Medan dapat dilakukan agar data jadwal kuliah, mata kuliah, dan tugas akademik dapat diperoleh secara otomatis tanpa perlu diinput secara manual oleh pengguna.

3. Penelitian selanjutnya dapat mengimplementasikan teknologi kecerdasan buatan (Artificial Intelligence) untuk memberikan rekomendasi prioritas tugas, estimasi waktu penyelesaian, serta penjadwalan belajar yang lebih adaptif sesuai dengan kebiasaan pengguna[21]. untuk memberikan rekomendasi prioritas tugas, estimasi waktu pengerjaan, serta penjadwalan belajar yang lebih adaptif sesuai kebiasaan pengguna.

4. Pengujian aplikasi dapat diperluas dengan melibatkan jumlah responden yang lebih besar dan berasal dari berbagai program studi maupun perguruan tinggi lain sehingga hasil evaluasi usability dan user experience menjadi lebih representatif.

5. Penelitian berikutnya dapat mengukur dampak penggunaan aplikasi dalam jangka panjang terhadap peningkatan prestasi akademik, kedisiplinan belajar, dan pengurangan perilaku procrastination mahasiswa.

6. Fitur kolaborasi antar mahasiswa, seperti berbagi tugas kelompok, pengingat bersama, dan sinkronisasi jadwal kegiatan akademik, dapat ditambahkan untuk meningkatkan manfaat aplikasi dalam mendukung aktivitas pembelajaran.

Dengan adanya pengembangan dan penelitian lanjutan tersebut, diharapkan aplikasi SyncPolmed dapat menjadi solusi manajemen tugas dan jadwal belajar yang lebih komprehensif serta memberikan manfaat yang lebih luas bagi mahasiswa.

Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada Politeknik Negeri Medan yang telah memberikan dukungan fasilitas dan lingkungan akademik yang kondusif untuk pelaksanaan penelitian ini. Terima kasih juga disampaikan kepada dosen pembimbing yang telah

memberikan bimbingan dan masukan berharga sepanjang proses penelitian. Penulis juga mengapresiasi partisipasi 50 mahasiswa POLMED yang telah meluangkan waktu untuk mengikuti pengujian aplikasi dan memberikan feedback yang konstruktif. Kritik dan saran yang membangun sangat diharapkan untuk penyempurnaan penelitian dan pengembangan aplikasi di masa mendatang.

DAFTAR PUSTAKA

- [1] A. Togaibayeva, D. Ramazanova, M. Yessengulova, A. Yergazina, A. Nurlin, and R. Shokanov, "Effect of mobile learning on students' satisfaction, perceived usefulness, and academic performance when learning a foreign language," *Front. Educ.*, vol. 7, Sep. 2022, doi: 10.3389/educ.2022.946102.
- [2] J. Valverde-Berrocso, J. Acevedo-Borrega, and M. Cerezo-Pizarro, "Educational Technology and Student Performance: A Systematic Review," *Front. Educ.*, vol. 7, Jun. 2022, doi: 10.3389/educ.2022.916502.
- [3] A. Meier, "Studying problems, not problematic usage: Do mobile checking habits increase procrastination and decrease well-being?," *Mob. Media Commun.*, vol. 10, no. 2, pp. 272–293, May 2022, doi: 10.1177/20501579211029326.
- [4] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*, 9th ed. New York: McGraw-Hill Education, 2020. [Online]. Available: [https://books.google.co.id/books?id=bL7QZHtWvaUC&lpg=PA1&ots=O9Bc6Otlak&dq=Pressman%2C R. S.%2C %26 Maxim%2C B. R. \(2020\). Software Engineering%3A A Practitioner's Approach \(9th ed.\). McGraw-Hill.%5C&lr&hl=id&pg=PR4#v=onepage&q&f=true](https://books.google.co.id/books?id=bL7QZHtWvaUC&lpg=PA1&ots=O9Bc6Otlak&dq=Pressman%2C R. S.%2C %26 Maxim%2C B. R. (2020). Software Engineering%3A A Practitioner's Approach (9th ed.). McGraw-Hill.%5C&lr&hl=id&pg=PR4#v=onepage&q&f=true)
- [5] N. T. Singh, Preeti, R. Kumar, A. Choubey, Suresh, and J. Singh, "An All-Inclusive Daily Task Tracker App for Enhancing College Student Productivity," in *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, IEEE, Jun. 2024, pp. 1–5. doi: 10.1109/ICCCNT61001.2024.10724109.
- [6] J. A. Martín-García and D. Pérez Marín, "A gamified mobile-based APP to help university students to manage their tasks," in *Eighth International Conference on Technological Ecosystems for Enhancing Multiculturality*, New York, NY, USA: ACM, Oct. 2020, pp. 655–661. doi: 10.1145/3434780.3436584.
- [7] W. Zhao *et al.*, "Design and Implementation of a Time Management Self-Help Mobile App for College Students," in *2023 IEEE Integrated STEM Education Conference (ISEC)*, IEEE, Mar. 2023, pp. 81–88. doi: 10.1109/ISEC57711.2023.10402177.
- [8] H.-M. Huang, S.-S. Liaw, and C.-M. Lai, "Exploring learner acceptance of the use of virtual reality in medical education: a case study of desktop and projection-based display systems," *Interact. Learn. Environ.*, vol. 24, no. 1, pp. 3–19, Jan. 2016, doi: 10.1080/10494820.2013.817436.
- [9] J. C. Ferreira, L. Vasconcelos, R. Monteiro, F. Z. Silva, C. M. Duarte, and F. Ferreira, "Ocean Literacy to Promote Sustainable Development Goals and Agenda 2030 in Coastal Communities," *Educ. Sci.*, vol. 11, no. 2, p. 62, Feb. 2021, doi: 10.3390/educsci11020062.
- [10] S. C. Peres, T. Pham, and R. Phillips, "Validation of the System Usability Scale (SUS)," *Proc. Hum. Factors Ergon. Soc. Annu. Meet.*, vol. 57, no. 1, pp. 192–196, Sep. 2013, doi: 10.1177/1541931213571043.
- [11] S. Sophonhiranrak, "Features, barriers, and influencing factors of mobile learning in higher education: A systematic review," *Heliyon*, vol. 7, no. 4, p. e06696, Apr. 2021, doi: 10.1016/j.heliyon.2021.e06696.
- [12] A. Hinze, N. Vanderschantz, C. Timpany, S. J. Cunningham, S.-J. Saravani, and C. Wilkinson, "A Study of Mobile App Use for Teaching and Research in Higher Education," *Technol. Knowl. Learn.*, vol. 28, no. 3, pp. 1271–1299, Sep. 2023, doi: 10.1007/s10758-022-09599-6.
- [13] A. M. Sattar *et al.*, "Accelerating Cross-platform Development with Flutter Framework," *J. OPEN SOURCE Dev.*, 2023, doi: 10.37591/joosd.v10i2.580.
- [14] A. M. Al-Rahmi, W. M. Al-Rahmi, U. Alturki, A. Aldraiweesh, S. Almutairy, and A. S. Al-Adwan, "Acceptance of mobile technologies and M-learning by university students: An empirical investigation in higher education," *Educ. Inf. Technol.*, vol. 27, no. 6, pp. 7805–7826, Jul. 2022, doi: 10.1007/s10639-022-10934-8.
- [15] John Brooke, *SUS: A "Quick and Dirty" Usability Scale*. 1996.
- [16] A. Bangor, P. T. Kortum, and J. T. Miller, "An Empirical Evaluation of the System Usability Scale," *Int. J. Human-Computer Interact.*, vol. 24, no. 6, pp. 574–594, Jul. 2008, doi: 10.1080/10447310802205776.



- [17] M. Schrepp, A. Hinderks, and J. Thomaschewski, "Applying the User Experience Questionnaire (UEQ) in Different Evaluation Scenarios," 2014, pp. 383–392. doi: 10.1007/978-3-319-07668-3_37.
- [18] E. Saptaputra, V. J. Basiroen, and A. Nagaria, "Supporting Class Infrastructure through Task Management Mobile Application during Pandemic," in *Proceedings of the International Conference on Industrial Engineering and Operations Management*, Michigan, USA: IEOM Society International, Aug. 2022, pp. 2068–2077. doi: 10.46254/IN02.20220518.
- [19] R. A. Grier, A. Bangor, P. Kortum, and S. C. Peres, "The System Usability Scale," *Proc. Hum. Factors Ergon. Soc. Annu. Meet.*, vol. 57, no. 1, pp. 187–191, Sep. 2013, doi: 10.1177/1541931213571042.
- [20] S. S. S. Husni, M. F. Kamaruzaman, and N. S. Wahyuni, "Kipidap: A mobile app designed for Malaysian university students to overcome procrastination," *Humanit. Arts Soc. Sci. Stud.*, Aug. 2025, doi: 10.69598/hasss.25.2.265569.
- [21] T. Giri, "Smart Student Productivity & Task Management System," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 14, no. 3, pp. 5143–5149, Mar. 2026, doi: 10.22214/ijraset.2026.78949.