Journal of Digital Technology and Computer Science
E-ISSN: 3030-8127; P-ISSN: 3031-0318
https://journal.lontaradigitech.com/DIGITECH

Pomify: Aplikasi Mobile Pomodoro Timer untuk Mengurangi Prokrastinasi dengan Metode Incremental

A. M. Farhath Zuhair[1*], Hana Awaliyah Najamuddin[2], Sultan Budiluhur [3]
[1,2,3] *Universitas Negeri Makassar, Indonesia*

Corresponding e-mail : xxx@gmail.com

INFO ARTIKEL

Kata Kunci:

Aplikasi Seluler
Manajemen Waktu
Metode Tambahan
Pengatur Waktu
Pomodoro

Riwayat Artikel

Menerima: Desember 20, 2023
Revisi: Februari 10, 2024
Diterima: Maret 05, 2024

ABSTRAK

Penelitian ini mengembangkan aplikasi mobile Pomify yang menerapkan teknik Pomodoro untuk membantu mengatur waktu dan mengurangi penundaan. Penelitian ini dilatarbelakangi oleh permasalahan tingginya tingkat penundaan akibat kurangnya manajemen waktu yang efektif, sehingga bertujuan untuk merancang solusi digital berupa aplikasi berbasis teknik Pomodoro yang tidak hanya mendukung fokus pengguna, tetapi juga memenuhi standar kualitas perangkat lunak melalui whitebox dan pengujian ISO/IEC 25010. Metode penelitian ini menggunakan model pengembangan inkremental dan whitebox dan pengujian ISO/IEC 25010 untuk memastikan kualitas fungsional dan kinerja aplikasi. Pengujian kotak putih pada fitur Daftar dan Login menunjukkan hasil yang valid dengan tingkat kompleksitas rendah (V(G) = 4 dan V(G) = 3), yang menunjukkan bahwa aplikasi berfungsi dengan baik tanpa kesalahan. Pengujian ISO/IEC 25010 menunjukkan bahwa Pomify memenuhi standar kualitas perangkat lunak. Kesesuaian fungsional mencapai 100%, menunjukkan bahwa semua fitur berfungsi sesuai dengan kebutuhan pengguna. Pengujian kegunaan mencetak 91,2%, menunjukkan tingkat kepuasan pengguna yang sangat baik. Dari segi keandalan, aplikasi berkinerja sangat baik dengan 98,67% responden memberikan penilaian positif terhadap aspek keandalan. Pengujian efisiensi kinerja dengan Android Profiler menunjukkan penggunaan CPU rata-rata 8% dan konsumsi RAM yang stabil (65-69MB), dengan waktu pemuatan yang cepat, yang tertinggi hanya 463ms. Secara keseluruhan, aplikasi Pomify memenuhi standar kualitas tinggi dalam hal kesesuaian fungsional, kegunaan, keandalan, dan efisiensi kinerja, dan efektif dalam mendukung manajemen waktu dan mengurangi penundaan.

To cite this article : Author. (20xx). Title. Information Technology Education Journal, X(X), XX-XX. Doi. xxxx

PENDAHULUAN

Manajemen waktu yang efektif menjadi salah satu faktor penentu kesuksesan dalam berbagai aspek kehidupan, baik dalam dunia pendidikan, pekerjaan, maupun kehidupan pribadi. Manajemen waktu adalah keterampilan dalam mengatur dan memanfaatkan waktu secara terencana agar dapat digunakan seefektif mungkin [3]. Salah satu tantangan utama yang dihadapi banyak individu dalam manajemen waktu adalah kesulitan dalam mengatur prioritas dan alokasi waktu secara efektif. Salah satu bentuknya adalah prokrastinasi [11]. Istilah prokrastinasi berasal dari bahasa latin "procrastinatire," yang terdiri dari kata "pro" yang berarti maju atau ke depan, dan "crastinus" yang berarti besok atau hari yang akan datang. Dengan demikian, prokrastinasi secara harfiah berarti menunda pekerjaan hingga hari esok atau memilih untuk melakukannya di kemudian hari [15]. Tuckman berpendapat bahwa prokrastinasi merupakan kecenderungan seseorang untuk menghindari atau meninggalkan tugas secara sengaja [4].

Namun, masih banyak individu yang kesulitan dalam mengelola waktu secara efektif dan terjebak dalam prokrastinasi, baik di ranah akademik maupun dalam kehidupan sehari-hari. Penelitian Dharma (2020) mengungkapkan bahwa 60% mahasiswa berada dalam kategori prokrastinasi akademik yang tinggi, 10% sangat tinggi, 13,3% sedang, 6,7% rendah, dan 10% sangat rendah. Prokrastinasi dipengaruhi oleh faktor eksternal seperti pola asuh, lingkungan, dan dukungan sosial, serta faktor internal yang terkait dengan kondisi fisik dan psikologis individu [5]. Salah satu faktor yang berasal dari dalam diri adalah kurangnya motivasi untuk mengerjakan tugas dan anggapan bahwa masih ada waktu untuk menyelesaikannya nanti, daripada mengerjakannya segera [13].

Prokrastinasi ini tidak hanya menghambat produktivitas, tetapi juga sering kali menyebabkan stres dan tekanan akibat pekerjaan yang tertunda [6]. Dampak negatif yang ditimbulkan oleh prokrastinasi, jika dibiarkan, dapat memengaruhi kebahagiaan, kesejahteraan, dan pencapaian individu dalam berbagai aspek kehidupan [7]. Selain itu, prokrastinasi dalam akademik juga dapat berdampak pada kondisi psikis, seperti munculnya gangguan emosi negatif seperti kecemasan dan stres [12]. Oleh karena itu, mengatasi prokrastinasi menjadi langkah yang sangat penting untuk mencapai potensi maksimal dan meraih tujuan secara lebih efektif [6].

Untuk mengatasi masalah tersebut, diusulkan sebuah sistem yang menggunakan teknik Pomodoro dalam pengerjaan tugas atau pekerjaan [8]. Peneliti mengembangkan aplikasi mobile dengan nama “Pomify” yang menerapkan teknik manajemen waktu Pomodoro dalam penyelesaian tugas. Teknik Pomodoro sendiri diperkenalkan oleh Francesco Cirillo dan dikenal sebagai metode yang efisien dalam mengelola waktu, dengan membagi pekerjaan ke dalam sesi fokus selama 25 menit, diselingi istirahat singkat selama 5 menit guna menjaga konsentrasi [9][10]. Pomify dirancang untuk membantu pelajar, profesional, dan individu dengan rutinitas padat agar dapat bekerja lebih terstruktur, fokus, dan bebas dari gangguan, dengan fitur utama berupa timer yang dapat disesuaikan, manajemen tugas, serta laporan aktivitas yang menampilkan statistik waktu dan progres tugas secara periodik.

Berbagai penelitian sebelumnya telah menunjukkan efektifitas penggunaan teknik pomodoro. Penelitian yang dilakukan oleh Yuniar, Wahyuni, & Permatasari (2023) mengembangkan sistem informasi berbasis teknik Pomodoro, "Ayo Nugas", yang bertujuan untuk manajemen waktu dan mengurangi prokrastinasi. Sistem ini memperoleh skor rata-rata 0,83 (Excellent) dalam uji kegunaan WEBUSE, dan hasil dari penyebaran kuesioner post-test menunjukkan bahwa 40% responden melaporkan pengurangan prokrastinasi setelah menggunakan sistem ini. Temuan ini mendukung efektivitas teknik Pomodoro dalam mengurangi prokrastinasi. Namun, sistem tersebut dikembangkan menggunakan metode Waterfall, yang memiliki kekurangan, seperti kesulitan dalam menangani perubahan atau kesalahan pada tahap akhir yang memerlukan revisi besar, berpotensi menambah waktu dan biaya [8][16].

Penelitian selanjutnya yang dilakukan oleh Gunawan, Indrawan, & Sariyasa (2021) mengembangkan Sistem Informasi Kemajuan Akademik (SIsKA) di Program Studi Ilmu Komputer pada Program Pascasarjana Universitas Pendidikan Ganesha menggunakan model pengembangan incremental . Tujuan utama penelitian ini adalah untuk memperbaiki pengelolaan data akademik dan penelitian melalui sistem yang sesuai dengan kebutuhan pengguna. Pengembangan tiga fase incremental dengan evaluasi berkelanjutan menunjukkan bahwa sistem sesuai dengan kebutuhan pengguna dan siap diimplementasikan, sementara pengujian White Box Testing mengidentifikasi beberapa fungsi dengan kompleksitas tinggi yang memerlukan perbaikan. Penelitian ini memiliki relevansi dengan penelitian yang sedang dilakukan, karena mengadopsi metode incremental yang sama dalam pengembangan aplikasi Pomify, serta

menerapkan pengujian White Box Testing untuk memastikan kualitas kode dan meminimalkan kompleksitas.

Selain itu penelitian oleh Reswara, Suakanto, & Alam (2022) mengembangkan backend aplikasi presensi karyawan berbasis mobile pada KSPPS Karya Usaha Mandiri (KUM) menggunakan metode Iterative Incremental [18]. Penelitian ini bertujuan untuk menggantikan pencatatan kehadiran manual dengan sistem yang lebih efisien, menghasilkan aplikasi Absenin yang dilengkapi dengan fitur seperti authentication, activity, report, profile, paid leave, dan dashboard. Metodologi yang digunakan sejalan dengan pendekatan yang diadopsi dalam penelitian ini, yang juga memanfaatkan Iterative Incremental untuk mendukung pengembangan sistem terstruktur dan memungkinkan evaluasi berkelanjutan.

Penelitian lain yang dilakukan oleh Roselline Fabelia Valentina, Nugroho Hari Purnomo, dan Santi Pramanasari (2024) mengkaji penerapan teknik Pomodoro untuk meningkatkan fokus dan konsentrasi belajar pada mata pelajaran IPS di kelas VII-I SMPN 36 Surabaya . Tujuan penelitian ini adalah untuk meningkatkan fokus peserta didik dengan menggunakan teknik Pomodoro, yang terbukti efektif dalam meningkatkan keaktifan dan konsentrasi. Hasil penelitian menunjukkan bahwa setelah penerapan teknik Pomodoro, persentase aktivitas siswa yang termasuk dalam kategori tinggi meningkat dari 13% pada siklus pertama menjadi 25% pada siklus kedua. Penelitian ini relevan dengan penelitian kami karena keduanya menggunakan teknik Pomodoro, dan aplikasi Pomify yang kami kembangkan akan mempermudah penerapan teknik Pomodoro dalam kegiatan produktif seperti belajar, dengan menyediakan akses mudah melalui platform mobile [19].

Penelitian oleh Rafliyanto dan Mukhlis (2023) mengkaji penerapan teknik Pomodoro dalam pembelajaran Pendidikan Agama Islam di SMK Muhammadiyah 2 Malang. Tujuan utama penelitian ini adalah mengeksplorasi efektivitas teknik Pomodoro untuk meningkatkan fokus siswa selama sesi belajar. Hasil penelitian menunjukkan bahwa teknik ini berhasil meningkatkan konsentrasi siswa dengan membagi waktu belajar menjadi sesi 25 menit diikuti dengan istirahat 5 menit, yang membantu mencegah kebosanan dan kelelahan mental. Penelitian ini relevan dengan penelitian kami karena keduanya menggunakan teknik Pomodoro, dan aplikasi Pomify yang kami kembangkan akan mempermudah penerapan teknik tersebut dalam kegiatan produktif melalui platform mobile [20].

Penelitian-penelitian terkait ini memberikan dasar yang kuat bagi pengembangan aplikasi mobile “Pomify” yang menerapkan teknik Pomodoro Timer. Meskipun teknik Pomodoro Timer telah diterapkan dalam banyak konteks, namun belum ada pengembangan aplikasi mobile yang yang mengintegrasikan teknik ini dengan pengujian whitebox untuk mengevaluasi serta memverifikasi kode internal dan memastikan tidak ada bug, serta pengujian ISO/IEC 25010 yang bertujuan untuk menilai kualitas perangkat lunak secara menyeluruh, mencakup aspek fungsionalitas, kegunaan (usability), keandalan (reliability), serta efisiensi kinerja (performance efficiency). Pengujian berdasarkan standar ini memastikan bahwa aplikasi Pomify tidak hanya efektif dalam implementasi teknik Pomodoro, tetapi juga memenuhi standar kualitas yang tinggi dalam berbagai dimensi, yang penting untuk meningkatkan kepuasan pengguna dan keberhasilan aplikasi di pasar.

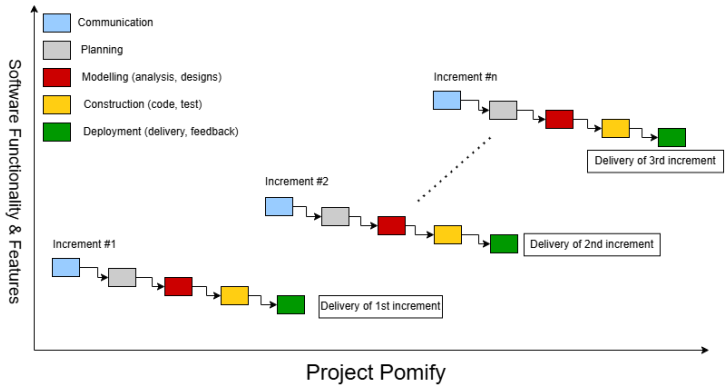
Dengan demikian, penelitian ini bertujuan untuk mengembangkan aplikasi mobile “Pomify” yang mengintegrasikan teknik Pomodoro, dengan harapan dapat membantu pengguna dalam mengelola waktu secara lebih efektif dan mengurangi prokrastinasi. Aplikasi ini tidak hanya menyediakan timer Pomodoro, tetapi juga dilengkapi dengan fitur pengelolaan tugas dan laporan yang memberikan wawasan tentang waktu yang digunakan serta tugas yang telah diselesaikan.

Dengan kebaruan berupa pengujian whitebox dan ISO/IEC 25010, Pomify memastikan kualitas kode, mendeteksi bug secara menyeluruh, serta mengevaluasi aspek fungsionalitas, keandalan, kegunaan, dan efisiensi kinerja. Kombinasi pengujian ini mendukung pengembangan aplikasi yang andal, produktif, dan efektif dalam membantu pengguna mengatasi prokrastinasi, baik dalam konteks akademik maupun profesional.

METODE

Model Pengembangan

Model pengembangan yang digunakan dalam penelitian ini adalah model pengembangan incremental. Metode incremental merupakan model pengembangan sistem dalam rekayasa perangkat lunak yang didasarkan pada pemecahan kebutuhan perangkat lunak menjadi sejumlah fungsi atau komponen, sehingga proses pengembangannya dilakukan secara bertahap [22]. Model Incremental menerapkan rekayasa perangkat lunak dengan membagi tugas secara bertahap hingga terbentuk perangkat lunak yang utuh. Proses ini akan berakhir ketika seluruh fungsi yang diinginkan telah tercapai [14]. Metode incremental memungkinkan pengguna untuk segera melakukan evaluasi terhadap aplikasi yang sedang dikembangkan (Wasesha, 2021). Gambar 1 menunjukkan tahapan model incremental.



Gambar 1. Tahapan Incremental Model

Tahapan pada model incremental menurut Washesha (2021) adalah sebagai berikut:

1. *Communication*

   Pada tahap ini, kebutuhan dikumpulkan melalui observasi dan wawancara, yang selanjutnya menghasilkan daftar kebutuhan fungsional beserta kategori pengguna yang terlibat.
2. *Planning*

   Pada tahap ini, disusun rencana mengenai tim yang akan menangani setiap tahap beserta estimasi waktu yang diperlukan untuk menyelesaikan setiap tugas.
3. *Modelling*

   Pengembang memanfaatkan berbagai diagram UML untuk menerjemahkan kebutuhan fungsional yang ada menjadi desain perangkat lunak. Desain ini mencakup aspek alur (perilaku) dan struktur dari perangkat lunak yang akan dibangun.
4. *Construction*

   Code: Setelah desain sistem selesai, langkah berikutnya adalah pengkodean. Coding adalah proses menyusun ide atau pemikiran secara terstruktur dalam bentuk sintaks yang sesuai dengan bahasa pemrograman yang digunakan, yang kemudian dapat diubah menjadi program yang dapat

dijalankan oleh komputer (Setyawan, 2024). Bahasa pemrograman yang digunakan adalah Java untuk aplikasi Android, dengan IDE Android Studio sebagai alat pengembangannya.
Test: Pada tahap ini, dilakukan berbagai pengujian, seperti pengujian instalasi, kompatibilitas, dan fungsionalitas. Tahap pengujian bertujuan untuk memastikan dan mengonfirmasi fungsionalitas sistem yang telah dibuat (Peta, Zidanie, & Setiawan, 2023).

5. *Deployment*

Pada tahap ini, perangkat lunak akan diserahkan kepada pengguna aplikasi "Pomify". Penggunaan perangkat lunak akan terus diawasi dan dipantau. Meskipun demikian, kemungkinan munculnya bug di masa depan tetap ada, dan jika ditemukan, proses akan kembali ke tahap awal untuk dilakukan analisis lebih lanjut. Tahap ini memungkinkan pengulangan proses pengembangan dari analisis spesifikasi untuk melakukan perubahan pada perangkat lunak yang sudah ada, namun tidak untuk menciptakan perangkat lunak baru.

Pengujian Sistem

Dalam menguji aplikasi ini, peneliti menggunakan 2 metode yaitu White Box dan ISO 25010. Kedua pengujian tersebut dilakukan untuk memastikan seluruh logika dan fungsi yang ada pada aplikasi dapat berjalan dengan baik dan memastikan bahwa perangkat lunak yang dikembangkan memenuhi standar kualitas yang tinggi dan dapat memberikan pengalaman pengguna yang optimal.

1. Whitebox Testing

Whitebox testing adalah metode pengujian yang fokus pada pengecekan logika dan kode program secara mendalam, dengan menggunakan metrik seperti Cyclomatic Complexity, Region, dan Independent Path. Pengujian ini dianggap berhasil jika nilai yang diperoleh konsisten, menunjukkan bahwa logika dan kode program telah dirancang dengan benar [24]. Whitebox testing adalah metode pengujian yang berfokus pada bagian internal sistem, khususnya pada kode sumber program. Oleh karena itu, akses terhadap kode sumber sangat dibutuhkan untuk melaksanakan pengujian ini. Metode ini memiliki sejumlah keunggulan, antara lain:

a. Mampu menghilangkan bagian-bagian tersembunyi yang tidak diperlukan dalam kode.
b. Memungkinkan pengujian secara menyeluruh karena seluruh struktur dan logika program dianalisis.
c. Membantu dalam proses optimalisasi kode
d. Serta dapat dilakukan meskipun antarmuka pengguna grafis (GUI) belum sepenuhnya selesai dikembangkan [23].

2. ISO/IEC 25010

Standar ISO 25010 yang dikembangkan oleh ISO membantu pengembang dan organisasi mengevaluasi, memahami, dan meningkatkan kualitas perangkat lunak secara terstruktur [25]. Penelitian ini memfokuskan pengujian pada empat aspek utama: functional suitability, usability, reliability, dan performance efficiency. Penilaian dilakukan secara kuantitatif melalui kuesioner kepada 30 mahasiswa dan analisis sistem menggunakan Android Profiler. Functional suitability diukur dengan Skala Guttman. Skala Guttman adalah metode pengukuran yang dirancang untuk memperoleh jawaban pasti dari responden, dengan pilihan yang terbatas pada dua opsi, seperti "setuju atau tidak setuju," "ya atau tidak," "benar atau salah," "positif atau negatif," "pernah atau tidak pernah," dan opsi serupa lainnya. sedangkan *usability* dan *reliability* menggunakan Skala Likert 1–5. untuk mengevaluasi opini, pandangan, dan pemahaman responden terhadap variabel yang diteliti [26]. Skala Likert diterapkan untuk mengevaluasi sudut pandang, opini, dan pemahaman individu atau

kelompok terkait fenomena sosial yang telah secara eksplisit ditentukan sebagai variabel dalam penelitian ini [26].

a. Functional Suitability

Merupakan kemampuan produk atau sistem untuk menyediakan fungsi yang sesuai dengan kebutuhan yang telah ditentukan, baik yang terlihat secara eksplisit maupun tersembunyi, ketika dijalankan dalam situasi tertentu (Willis, 2021). Instrumen pernyataan untuk Functional Suitability dapat dilihat pada Tabel 1. Tabel tersebut mencakup serangkaian indikator yang digunakan untuk mengukur sejauh mana aplikasi memenuhi kebutuhan fungsional yang telah ditentukan.

Tabel 1. Instrumen Functional Suitability

Aspek yang dinilai	Indikator	Deskripsi
Functionality	*Functional Appropriateness*	Informasi atau data yang tersedia pada aplikasi sudah lengkap
	Functional Correctness	Tombol atau menu yang ada pada aplikasi dapat digunakan.

b. Usability

Mengukur sejauh mana produk atau sistem dapat digunakan oleh pengguna untuk mencapai tujuan tertentu dengan cara yang efektif, efisien, dan memuaskan dalam konteks penggunaannya (Willis, 2021). Aspek yang dinilai mencakup usefulness, ease of use, ease of learning, dan satisfaction, yang merefleksikan persepsi responden terhadap kemudahan, kegunaan, pembelajaran, serta kepuasan dalam menggunakan sistem. Instrumen pengukuran untuk aspek-aspek tersebut dapat dilihat pada Tabel 2.

Tabel 2. Instrumen *Usability*

Aspek yang dinilai	Indikator	Deskripsi
Functionality	*Usefullnes*	Sejauh mana kemampuan perangkat lunak dalam menarik pengguna berinteraksi dengan menyenangkan
	Easy of Use	Tombol atau menu yang ada pada aplikasi dapat digunakan.
	Easy of Learning	Seberapa jauh sebuah perangkat lunak dapat dipelajari dengan baik oleh penggunanya
	Satisfication	Sejauh mana kepuasan pengguna terhadap aplikasi

c. Reliability

Menggambarkan kemampuan produk, sistem, atau komponen untuk beroperasi secara konsisten dan menjalankan fungsinya dengan baik dalam kondisi tertentu selama periode waktu yang telah ditentukan (Willis, 2021). Aspek yang dinilai mencakup maturity, fault tolerance,

recoverability, availability, dan eror rate Instrumen pengukuran untuk aspek-aspek tersebut dapat dilihat pada Tabel 3.

Tabel 3. Instrumen *Reliability*

Aspek yang dinilai	Indikator	Deskripsi
Reliability	*Maturity*	Mengukur persepsi pengguna terhadap stabilitas dan keberlanjutan fungsionalitas aplikasi dalam jangka panjang.
	Fault Tolerance	Mengukur persepsi pengguna mengenai kemampuan aplikasi untuk tetap berfungsi meskipun terjadi kesalahan atau gangguan kecil dalam sistem, seperti koneksi internet yang buruk atau interupsi lainnya.
	Recoverability	Mengukur seberapa cepat dan efektif aplikasi dapat kembali berfungsi setelah mengalami gangguan atau kesalahan.
	Availability	Mengukur persepsi pengguna mengenai sejauh mana aplikasi dapat diakses dan digunakan tanpa adanya gangguan atau downtime yang signifikan.
	Eror Rate	Mengukur persepsi pengguna tentang seberapa sering aplikasi mengalami kesalahan, seperti bug atau crash, yang mengganggu pengalaman pengguna.

d. Performance Efficiency

Performance Eficiency menunjukkan tingkat kinerja produk atau sistem dalam memanfaatkan sumber daya yang tersedia secara optimal di bawah kondisi yang telah ditetapkan (Willis, 2021). Aspek performance efficiency yang diukur pada aplikasi mengacu pada tiga kriteria utama, yaitu penggunaan CPU, konsumsi memori, dan waktu muat (load time). Analisis aspek Performance Eficiency dapat dilihat pada Tabel 6.

Tabel 4. Analisis aspek *Performance Eficiency*

Respon Waktu (Detik)	Predikat
<3	Sangat Puas
3-9	Puas
10-12	Cukup Puas
>12	Tidak Puas

HASIL DAN DISKUSI

Communication

Pada tahap ini, pengumpulan kebutuhan sistem dilakukan melalui wawancara terstruktur dengan calon pengguna. Hanya terdapat satu aktor utama, yaitu pengguna aplikasi, yang merupakan individu produktif yang berminat mengimplementasikan teknik Pomodoro dalam aktivitas sehari-hari mereka. Hasil wawancara dianalisis untuk merumuskan daftar kebutuhan fungsional, yang kemudian disajikan dalam Tabel 5.

Tabel 5. Kebutuhan Fungsional

Kebutuhan Fungsional	Deskripsi
Menampilkan tugas	Sistem harus menampilkan daftar tugas yang dapat diprioritaskan oleh pengguna, dengan opsi untuk mengurutkan berdasarkan prioritas tinggi, sedang, dan rendah
Menambahkan tugas ke daftar tugas	Sistem harus memungkinkan pengguna untuk menambahkan tugas ke dalam daftar tugas yang harus diselesaikan, dengan memasukkan nama tugas dan deskripsi tugas
Mengatur durasi Timer pomodoro,Istirahat pendek dan istirahat panjang	Sistem harus memungkinkan pengguna untuk mengatur timer Pomodoro dengan durasi yang dapat disesuaikan (misalnya 25 menit untuk sesi kerja dan 5 menit untuk istirahat). Pengguna harus dapat memulai, menghentikan, atau melanjutkan ke tahap selanjutnya. Sistem juga harus memberi notifikasi suara saat sesi kerja atau istirahat selesai
Fitur laporan yang mencatat aktivitas penggunaan pomodoro dalam per hari,minggu dan bulan	Sistem harus membuat laporan waktu dan aktivitas yang mencatat penggunaan Pomodoro timer, termasuk waktu kerja, waktu istirahat pendek, dan waktu istirahat panjang. Laporan harus dapat di filter berdasarkan rentang waktu (harian, mingguan, bulanan), dan menampilkan data dalam format teks.
Dapat registrasi belum punya akun dan login ke homepage menggunakan akun yang dibuat	Sistem harus memungkinkan pengguna untuk melakukan registrasi dengan memasukkan nama, nama pengguna, kata sandi, dan konfirmasi kata sandi. Setelah registrasi, pengguna dapat login menggunakan email dan kata sandi yang telah terdaftar.
Menampilkan timer sesuai dengan sesi dan pengguna dapat memulai dan menghentikan timer setiap sesi	Sistem harus menampilkan timer terpisah untuk Pomodoro (25 menit), short break (5 menit), dan long break (15 menit). Pengguna harus dapat memulai, menghentikan, atau melanjutkan ke tahap selanjutnya. Timer harus menunjukkan waktu yang tersisa.
Menu navigasi bar	Sistem harus menampilkan menu beranda dengan tombol navigasi yang jelas untuk fitur utama seperti tugas, profil dan laporan. Dan menyertakan ikon untuk setiap navigasinya
List Daftar tugas	Sistem harus menampilkan daftar tugas yang telah ditambahkan oleh pengguna, mencakup nama tugas, deskripsi dan prioritas. Tugas harus dapat diurutkan berdasarkan prioritas. Pengguna juga dapat mengedit atau menghapus tugas dari daftar.
Fitur log out dan kembali ke halaman login	Sistem harus memungkinkan pengguna untuk melakukan log out dengan satu klik pada tombol log out. Setelah log out, pengguna harus diarahkan kembali ke halaman login, dan semua sesi pengguna harus diakhiri dengan aman. Sistem juga harus memberikan notifikasi konfirmasi sebelum log out untuk memastikan bahwa pengguna ingin keluar dari aplikasi.
Login dengan akun yang berbeda beda	Sistem harus mendukung login dengan akun pengguna yang berbeda-beda, yang masing-masing memiliki username yang unik dan telah terdaftar di sistem. Pengguna dapat login

	menggunakan username yang terdaftar dan memasukkan kata sandi yang sesuai
Fitur panduan penggunaan	Sistem harus menampilkan panduan cara penggunaan di profil pengguna, dalam format yang mudah dibaca, yang mencakup langkah-langkah penggunaan utama.
Fitur profile	Sistem harus menampilkan halaman profil yang berisi informasi pengguna, termasuk nama. Halaman profil juga harus menyediakan opsi untuk mengedit informasi pengguna.
Fitur Mengkategorikan tugas	Sistem harus memungkinkan pengguna untuk mengkategorikan tugas dengan memilih kategori yang telah ditentukan atau membuat kategori baru. Setiap tugas yang ditambahkan harus dapat dikategorikan dalam satu atau lebih kategori, seperti pekerjaan, studi, pribadi, atau kategori kustom lainnya.
Fitur prioritas tugas	Sistem harus menampilkan daftar tugas yang dapat diprioritaskan oleh pengguna, dengan opsi untuk mengurutkan berdasarkan prioritas tinggi, sedang, dan rendah

Planning

Tahap ini melibatkan penetapan struktur tim pengembang dan perkiraan waktu pengerjaan untuk setiap fitur aplikasi. Berdasarkan cakupan kebutuhan fungsional, dibuat jadwal pengerjaan yang mengalokasikan tugas dan durasi kerja untuk setiap anggota tim. Tabel 6 menunjukkan rincian penjadwalan tersebut.

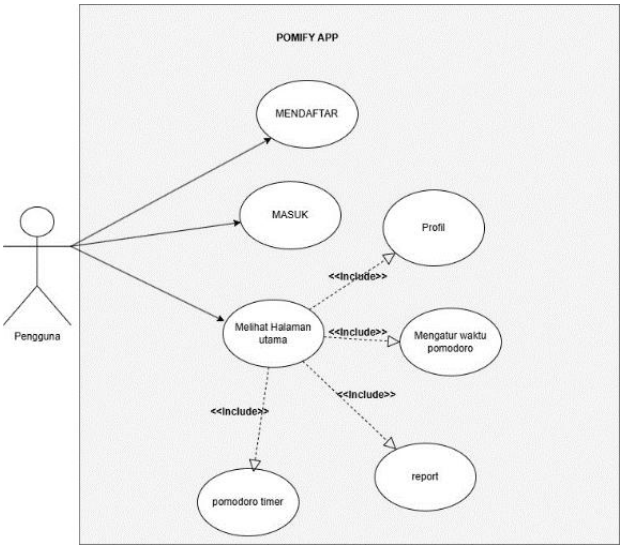
Tabel 6. Timeline Proyek

No	Fitur/Kegiatan	Februari	Maret				April				Mei			
		4	1	2	3	4	1	2	3	4	1	2	3	4
1.	Fitur untuk mengatur prioritas tugas													
2.	Fitur untuk mengategorikan tugas berdasarkan jenis aktivitas (belajar, olahraga, dll.)													
3.	Fitur untuk menambahkan tugas baru yang perlu diselesaikan													
4.	Fitur untuk menyesuaikan durasi timer Pomodoro dan sesi istirahat													
5.	Fitur untuk registrasi dan login ke aplikasi dengan berbagai akun pengguna													

6.	Fitur untuk menampilkan timer Pomodoro, short break, dan long break	
7.	Menampilkan semua fitur di beranda	
8.	Menampilkan tugas yang ditambahkan	
9.	Fitur log out	
10.	Fitur untuk login dengan akun pengguna yang berbeda-beda	
11.	Fitur laporan aktivitas dan waktu	
12.	Fitur untuk menampilkan cara penggunaan aplikasi di profil pengguna	
13.	Halaman profil yang menampilkan informasi pengguna	

Modelling

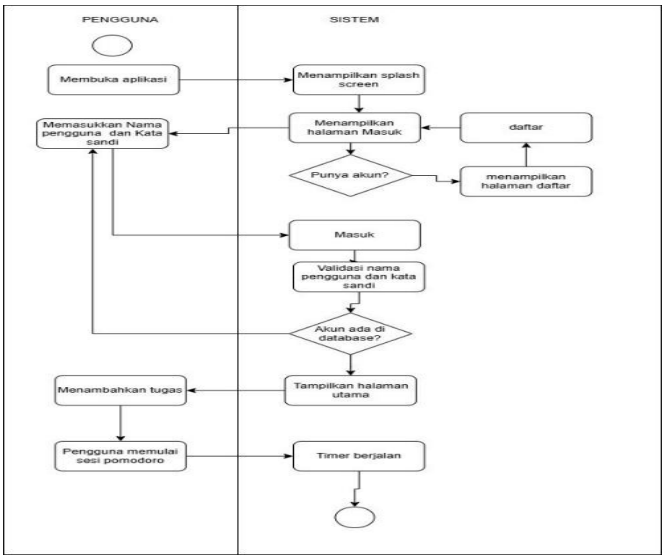
Berdasarkan daftar kebutuhan fungsional yang telah dirumuskan, sistem direpresentasikan secara konseptual melalui diagram UML untuk menggambarkan struktur dan perilaku aplikasi. Gambaran umum interaksi antaraktor dan proses dalam aplikasi diilustrasikan melalui Use Case Diagram pada Gambar 2, yang selanjutnya menjadi acuan bagi tahap implementasi dan pengujian sistem.



Gambar 2. Diagram Usecase

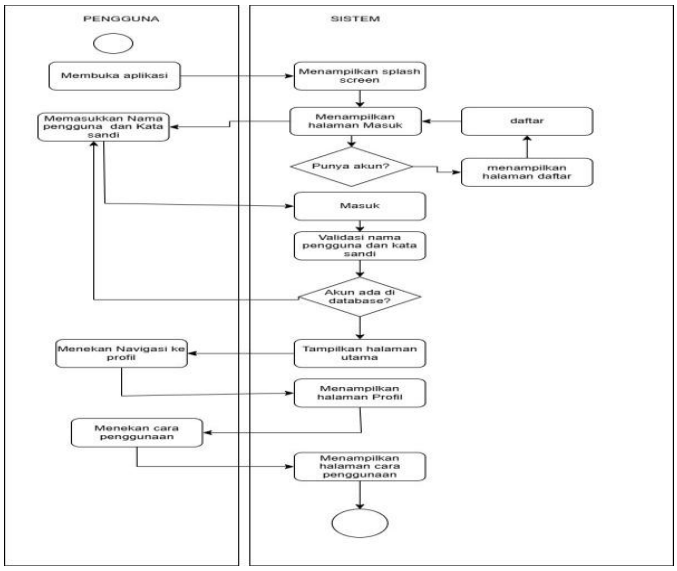
Pengguna harus mendaftar untuk membuat akun dan dapat mengakses fitur lanjutan seperti Profil dan Report. Setelah registrasi, pengguna akan melakukan autentikasi (login) dengan

kredensial yang valid. Setelah login, pengguna akan diarahkan ke Halaman Utama, di mana mereka dapat mengakses Profil untuk memperbarui informasi akun, mengatur waktu Pomodoro, serta memulai dan menghentikan Pomodoro Timer untuk manajemen waktu. Pengguna juga dapat melihat Report yang menampilkan ringkasan statistik penggunaan. Alur dari tiap kebutuhan fungsional dijabarkan lagi dengan diagram activity. Disini peneliti akan menjabarkan alur dari menambahkan tugas dan memulai sesi pomodoro seperti pada gambar 3.



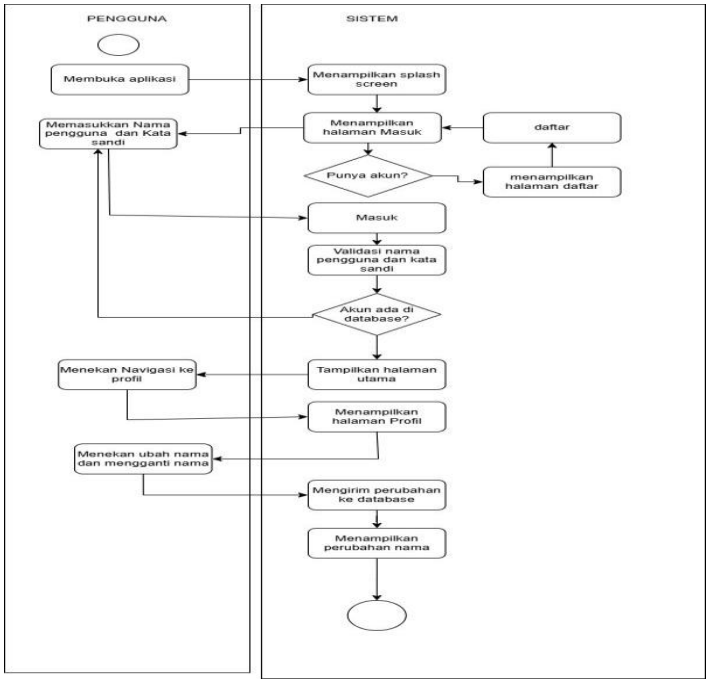
Gambar 3. Diagram Activity Menambahkan Tugas dan Memulai Sesi Pomodoro

Untuk penjabaran alur fitur panduan pengguna akan dijabarkan pada gambar 4.



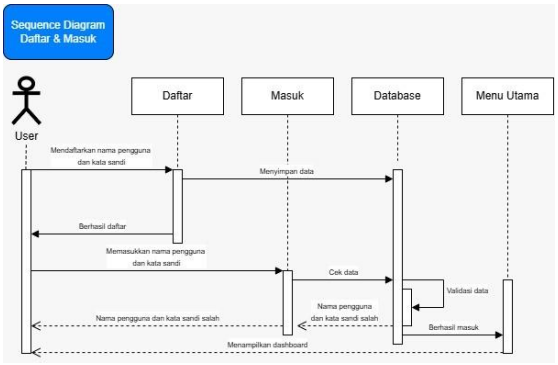
Gambar 4. Diagram Activity Menampilkan Panduan Penggunaan Pomodor Timer

Untuk penjabaran alur fitur profile yaitu mengubah dan mengganti nama akan dijabarkan pada gambar 5.



Gambar 5. Diagram Activity Mengganti dan Mengubah Nama dalam Fitur Profile

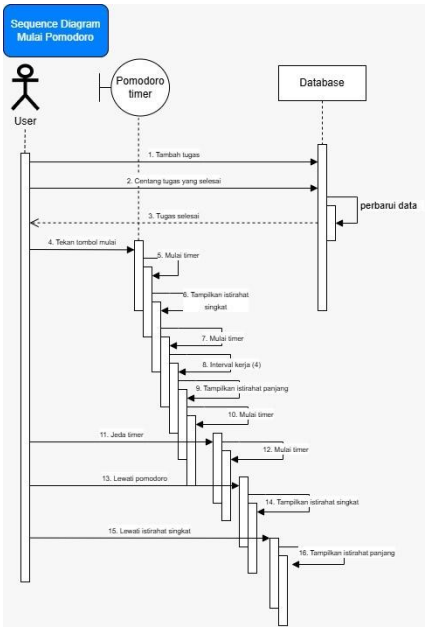
Secara detail alur dari proses daftar dan masuk dijelaskan kembali pada Gambar 6 diagram sequence proses daftar dan masuk. Pengguna pertama kali melakukan pendaftaran dengan memasukkan nama pengguna dan kata sandi. Setelah pendaftaran berhasil, sistem mengirimkan notifikasi, dan data pengguna disimpan di *database*. Setelah itu, pengguna dapat masuk dengan kredensial yang sudah terdaftar. Sistem memverifikasi data yang dimasukkan dengan yang ada di *database*. Jika data salah, pesan kesalahan akan ditampilkan. Jika benar, sistem mengonfirmasi dan menampilkan menu utama, yang memberikan akses ke fitur aplikasi.



Gambar 6. Diagram Sequence Daftar dan Masuk

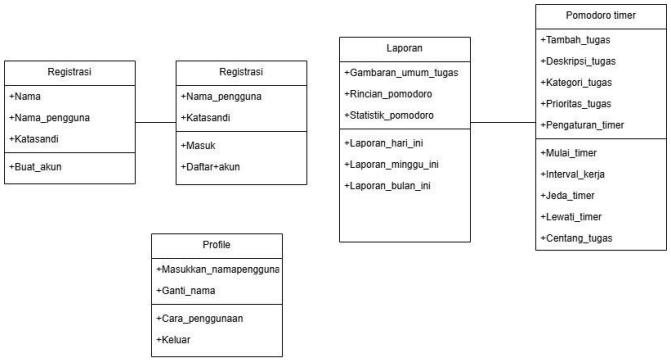
Alur interaksi antara pengguna, Pomodoro timer, dan database dalam sistem yang mengimplementasikan teknik Pomodoro dijelaskan secara rinci dalam Gambar 7, yang menggambarkan diagram sequence "Mulai Pomodoro Timer". Diagram ini menunjukkan proses pengguna yang pertama kali menambahkan tugas, kemudian menandai tugas yang telah selesai. Pengguna kemudian memulai timer Pomodoro untuk sesi kerja selama 25 menit, diikuti dengan istirahat singkat selama 5 menit, dan setelah beberapa sesi, sistem memberikan istirahat panjang. Database mencatat tugas yang selesai dan memperbarui status timer secara berkelanjutan. Setelah menyelesaikan satu sesi, pengguna dapat memilih untuk melanjutkan ke sesi berikutnya atau melewati sesi tersebut. Diagram ini menggambarkan penerapan teknik Pomodoro dalam

manajemen waktu dan pengelolaan tugas, dengan pembaruan data yang dilakukan secara terus-menerus oleh database.



Gambar 7. Diagram Sequence Mulai Pomodoro Timer

Rancangan sistem berikutnya dilihat dari segistruktural, penulis menggunakan diagram class untuk memodelkannya seperti terlihat pada gambar 8.



Gambar 8. Diagram Class

Rancangan sistem secara struktural dimodelkan menggunakan diagram kelas, seperti yang terlihat pada gambar yang diunggah. Terdapat beberapa kelas yang saling berelasi, yaitu Registrasi, Profile, Pomodoro timer, dan Laporan. Kelas Registrasi menyimpan data pengguna dan terhubung dengan Profile untuk memperbarui informasi pengguna. Pomodoro timer mengelola sesi kerja dan istirahat serta mencatat tugas yang diselesaikan, sementara Laporan menampilkan informasi penggunaan Pomodoro, termasuk laporan harian, mingguan, dan bulanan. Setiap kelas memiliki metode yang memungkinkan interaksi antar objek dalam sistem, mendukung pengelolaan waktu yang efisien dan terstruktur.

Construction

1. Code

Pada tahap ini, rancangan perangkat lunak diubah menjadi kode program menggunakan Java pada Android Studio. Sebagai backend, dipilih Firebase karena kemudahannya dalam integrasi, skala yang tinggi, dukungan data real-time, serta mekanisme keamanan terintegrasi yang menjamin kerahasiaan dan integritas data pengguna. Aplikasi Pomify menyediakan enam halaman utama yaitu:

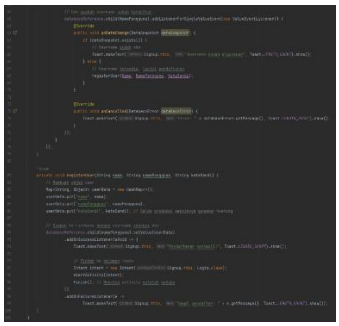
a. Masuk (Login)
   Sebelum dapat mengakses fitur aplikasi, pengguna harus melakukan autentikasi. Pada halaman ini, pengguna memasukkan username dan password yang telah terdaftar. Jika pengguna belum memiliki akun, mereka dapat kembali ke halaman daftar dengan menekan tombol "Daftar".
b. Daftar (Register)
   Pada halaman daftar, pengguna diminta untuk mengisi nama, username, dan kata sandi mereka. Setelah menekan tombol "Buat Akun", proses pembuatan akun selesai, dan pengguna kembali ke layar Masuk untuk melakukan autentikasi.
c. Beranda (Home)
   Halaman beranda memiliki Pomodoro Timer dan kontrol Pause/Start. Ada juga modul Pengaturan Waktu yang memungkinkan pengguna mengatur durasi sesi kerja dan istirahat. Selain itu, tersedia fitur Tambah Tugas dan daftar tugas (task list) yang telah dibuat. Dengan menggunakan tiga tombol navigasi, Pomodoro, Istirahat Pendek, dan Istirahat Panjang, pengguna dapat beralih antarsesi.
d. Laporan (Report)
   Pengguna dapat melihat ringkasan jumlah tugas total dan jumlah tugas yang telah diselesaikan pada halaman Report. Meskipun kategori tugas menunjukkan jenis tugas dan jumlah iterasi Pomodoro yang dijalankan, statistik Pomodoro Timer menunjukkan metrik penggunaan. Selain itu, ada tiga tombol filter: Hari Ini, Minggu Ini, dan Bulan Ini. Tombol-tombol ini digunakan untuk memfokuskan tampilan laporan berdasarkan periode waktu.
e. Profil
   Halaman profil menampilkan informasi nama pengguna dan memiliki tiga tombol aksi: Ganti Nama, Keluar, dan Cara Penggunaan. Ganti Nama memungkinkan pengguna memperbarui nama tampilan, Keluar akan mengakhiri sesi dan mengembalikan pengguna ke halaman masuk, dan Cara Penggunaan membantu pengguna menggunakan aplikasi.
f. Cara penggunaan
   Pengguna baru dapat menggunakan halaman cara penggunaan untuk memahami teknik Pomodoro secara visual dan tekstual.

Seluruh struktur layar, kode sumber, dan detail implementasi lebih lanjut disajikan pada Tabel 7.

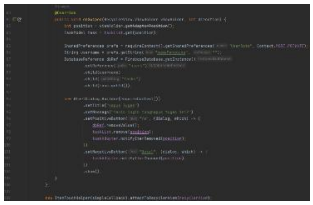
Tabel 7. Kode Program dan Tampilan Hasil

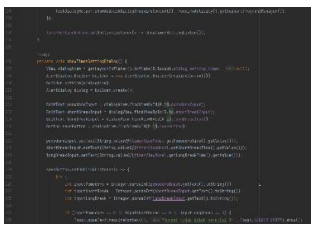
Coding	Tampilan Hasil
Masuk  	

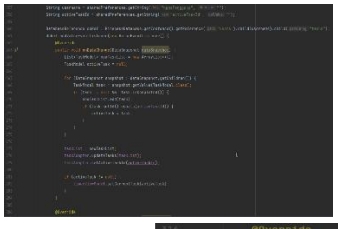
Daftar

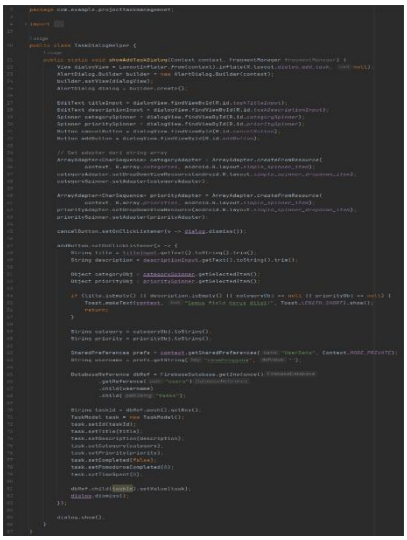


Beranda dan Pengaturan Waktu

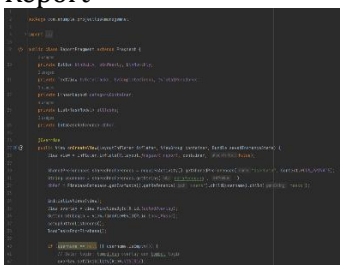


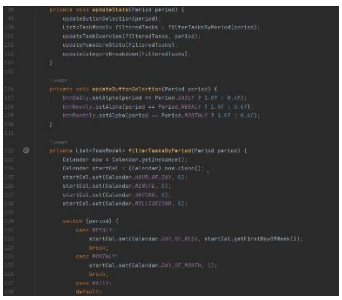


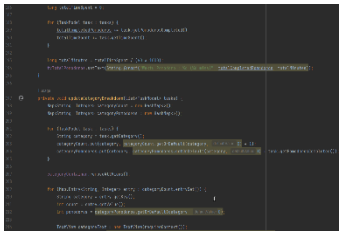


Tambah Tugas 	

Report




Profile


Panduan
`package com.example.projecttaskmanagemnet;`
`import ...`
`public class Panduan extends AppCompatActivity {`
`@Override`
`protected void onCreate(Bundle savedInstanceState) {`
`super.onCreate(savedInstanceState);`
`setContentView(R.layout.panduanpenggunaan); // Layout untuk halaman Panduan`
`// Menetapkan Toolbar sebagai Action Bar`
`Toolbar toolbar = findViewById(R.id.toolbar);`
`setSupportActionBar(toolbar);`
`// Menampilkan tombol kembali di Action Bar`
`if (getSupportActionBar() != null) {`
`getSupportActionBar().setDisplayHomeAsUpEnabled(true); // Tombol back`
`getSupportActionBar().setDisplayShowHomeEnabled(true); // Ikon home`
`getSupportActionBar().setTitle("");`
`}`
`}`
`1 usage`
`@Override`
`public boolean onSupportNavigateUp() {`
`onBackPressed(); // Ketika tombol back ditekan`
`return true;`
`}`
`}`
	

Code Github	https://github.com/Maro210/project_POMODORO_TIMER

2. Test

Dalam proses pengembangan perangkat lunak, Test merupakan tahap penting yang bertujuan untuk memastikan bahwa aplikasi bekerja sesuai dengan kebutuhan dan dapat digunakan secara optimal oleh pengguna. Pengujian dilakukan untuk mengevaluasi kualitas perangkat lunak dari berbagai aspek, salah satunya adalah pengujian fungsional dan pengujian kompatibilitas. Kedua jenis pengujian ini dilakukan untuk mengidentifikasi dan memperbaiki potensi kesalahan (bug) serta memastikan bahwa aplikasi dapat berjalan secara konsisten di berbagai kondisi penggunaan.

Pengujian kompatibilitas dilakukan untuk memastikan bahwa aplikasi Pomify dapat dijalankan secara konsisten dan optimal pada berbagai perangkat dan versi sistem operasi Android. Uji coba dilakukan pada beberapa perangkat dengan variasi ukuran layar, spesifikasi perangkat keras, serta versi Android yang berbeda. Fokus pengujian ini mencakup tampilan antarmuka, kestabilan aplikasi, serta kemampuan aplikasi dalam menangani berbagai konfigurasi perangkat. Berdasarkan hasil pengujian, aplikasi Pomify menunjukkan kinerja yang stabil dan responsif pada sebagian besar perangkat yang diuji, tanpa mengalami gangguan tampilan maupun crash, yang menunjukkan bahwa aplikasi memiliki kompatibilitas yang baik terhadap lingkungan pengguna yang beragam. Hasil pengujian kompabilitas dapat dilihat pada Tabel 8.

Tabel 8. Hasil pengujian kompabilitas

	Test 1	**Test 2**
Merk dan tipe	Oppo	Vivo T1
CPU	Helio G35	Dimensity 810
Ram	3GB	8GB
Layar	6,56 inci	6,58 inci
Versi android	12	11

Hasil	Berjalan dengan baik	Berjalan dengan baik

Pengujian fungsional bertujuan untuk memverifikasi bahwa seluruh fitur dan fungsi yang telah dirancang dalam aplikasi Pomify berjalan sesuai dengan spesifikasi yang ditetapkan. Pada tahap ini, dilakukan pengujian terhadap fitur-fitur utama seperti login dan registrasi, pengaturan waktu Pomodoro, penambahan dan penghapusan tugas, serta tampilan laporan durasi dan penyelesaian tugas. Setiap fungsi diuji berdasarkan input yang diberikan oleh pengguna untuk memastikan bahwa output yang dihasilkan sesuai dengan harapan. Hasil pengujian menunjukkan bahwa seluruh fitur inti aplikasi dapat berfungsi dengan baik, tanpa ditemukan kegagalan fungsi yang berarti. Hasil pengujian fungsional dapat dilihat pada Tabel 9.

Tabel 9. Pengujian fungsional

Skenario test	Hasil yang diinginkan	Hasil yang didapatkan	Hasil aktual
Menggunakan Timer			
Pada halaman utama Tekan tombol “Mulai”	Timer akan berjalan	Timer berjalan	Sesuai
Menekan tombol istirahat pendek dan panjang	Akan berpindah sesi dan timernya akan sesuai dengan sesi tombol yang ditekan	Timer dan sesinya sesuai dengan Tombol yang ditekan	Sesuai
Menekan tombol “Lewati”	Sesi akan berpindah	Sesi berpindah	Sesuai
Menekan tombol “Pengaturan waktu” dan memasukkan nilai waktu yang ingin di ubah dan menekan tombol”simpan”	Waktu sesi akan berubah sesuai dengan nilai yang dimasukkan di Pengaturan waktu	Waktu sesi berubah sesuai dengan nilai yang dimasukkan	Sesuai
Menekan Tombol “Berhenti”	Timer akan berhenti	Timer berhenti	Sesuai
Masuk dan daftar			
Saat Menjalankan aplikasi untuk pertama kali dan belum “Masuk”maka tidak bisa mengakses Profil dan report	Akan muncu layar abu abu dengan gambar kunci dan Tombol “Masuk/daftar”	Muncul layar abu abu dengan gambar kunci dan tombol masuk	Sesuai
Menekan tombol “masuk/daftar”saat mengakses report atau profil	Akan di arahkan ke halaman Login	Di arahkan ke halaman login	Sesuai

Menekan Teks “sign up here”	Akan di arahkan ke halaman daftar	Di arahkan ke halaman daftar	Sesuai
Memasukkan nama,Nama pengguna, dan kata sandi di halaman Daftar	Akan menyimpan input pengguna ke database dan mengarahkan ke halaman login	menyimpan input pengguna ke database dan mengarahkan ke halaman login	Sesuai
Memasukkan Nama pengguna dan kata sandi yang telah dibuat	Akan mengarahkan ke halaman utama dan layar hitam hilang saat mengakses profil dan report	mengarahkan ke halaman utama dan layar hitam hilang saat mengakses profil dan report	Sesuai
Mengganti nama dan Akses panduan			
Menekan “ganti nama”	Form nama bisa diubah	Form nama bisa diubah	Sesuai
Menekan Cara penggunaan	Mengarahkan ke Halaman cara penggunaan	Masuk ke halaman cara penggunaan	Sesuai
Menekan tombol “keluar”	Mengeluarkan pengguna dan di arahkan ke halaman Masuk	Pengguna dikeluarkan dan masuk ke halaman Masuk	Sesuai
Fitur report			
Menekan navbar report	Menampilkan Tugas,Statistik pomodoro dan kategori tugas dalam rentang waktu hari ini	Menampilkan Tugas,Statistik pomodoro dan kategori tugas dalam rentang waktu hari ini	Sesuai
Menekan tombol”mingguini”	Menampilkan Tugas,Statistik pomodoro dan kategori tugas dalam rentang waktu minggu ini	Menampilkan Tugas,Statistik pomodoro dan kategori tugas dalam rentang waktu Minggu ini	Sesuai
Menekan tombol”Bulan ini”	Menampilkan Tugas,Statistik pomodoro dan kategori tugas dalam rentang waktu bulan ini	Menampilkan Tugas,Statistik pomodoro dan kategori tugas dalam rentang waktu bulan ini	Sesuai

Setelah dilakukan pengujian terhadap aspek fungsional dan kompatibilitas, diperoleh hasil bahwa seluruh fitur utama dalam aplikasi Pomify berjalan dengan baik dan sesuai dengan

kebutuhan yang telah dirancang. Tidak ditemukan kendala berarti dalam proses penggunaan fitur seperti login, manajemen tugas, timer Pomodoro, maupun pelaporan aktivitas. Selain itu, hasil pengujian kompatibilitas menunjukkan bahwa aplikasi dapat berfungsi secara optimal di 2 perangkat tanpa terjadi masalah tampilan atau performa. Temuan ini mengindikasikan bahwa Pomify telah memenuhi standar kelayakan dalam hal fungsionalitas dan kompatibilitas, sehingga siap digunakan oleh pengguna dalam kondisi yang beragam.

3. Whitebox Testing

Pada tahap ini, setiap cabang, kondisi, dan jalur eksekusi dalam kode akan diuji secara sistematis untuk memastikan tidak ada alur logika yang terlewat. Terdapat dua fitur yang diuji, yaitu fitur sign up dan login. Fitur yang diuji pertama adalah fitur daftar, kode program untuk pengujian fitur daftar dapat dilihat pada Gambar 9.

Gambar 9. Kode Program Fitur Daftar

Flowchart dan flowgraph fitur daftar dapat dilihat pada Gambar 10.

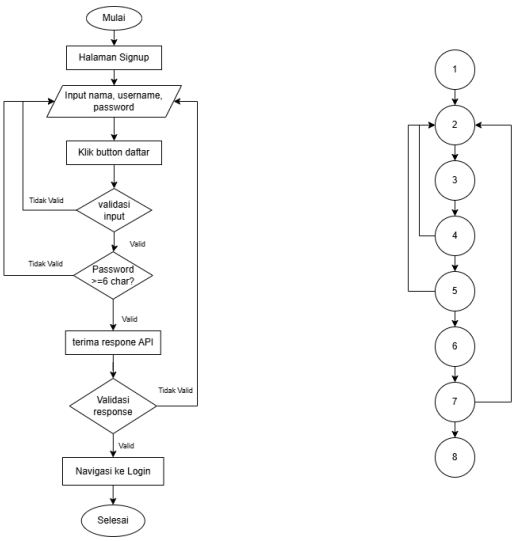


Figure 10. Flowchart dan Flowgraph Fitur Daftar

Cyclomatic Complexity:
N = 8
E = 10
V(G) = (E−N)+2
= 10−8+2 = 4

Keterangan:
V(G) = Jumlah Region
E = Jumlah edge yang ditentukan dengan gambar panah
N = Jumlah node dengan gambar lingkaran

Independent path:
Jalur 1 = 1-2-3-4-5-6-7-8
Jalur 2 = 1-2-3-4-2-3-4-5-6-7-8
Jalur 3 = 1-2-3-4-5-2-3-4-5-6-7-8
Jalur 4 = 1-2-3-4-5-6-7-2-3-4-5-6-7-8

Tabel pengujian path fitur daftar dapat dilihat pada Tabel 10.

Tabel 10. Pengujian Path Fitur Daftar

Path	**1**
Jalur	1-2-3-4-5-6-7-8
Skenario	1. Halaman signup 2. Masukkan nama, username, dan password 3. Tekan register 4. Validasi input (valid) 5. Validasi password (valid) 6. Terima response ke API 7. Validasi response (valid) 8. Berhasil, pindah ke Login
Hasil pengujian	Berhasil
Path	**2**
Jalur	1-2-3-4-2-3-4-5-6-7-8
Skenario	1. Halaman signup 2. Masukkan nama, username, dan password 3. Tekan register 4. Validasi input (tidak valid) 2. Masukkan nama, username, dan password 3. Tekan register 4. Validasi input (valid) 5. Validasi password (valid) 6. Terima response ke API 7. Validasi response (valid) 8. Berhasil, pindah ke Login
Hasil pengujian	Berhasil
Path	**3**
Jalur	1-2-3-4-5-2-3-4-5-6-7-8
Skenario	1. Halaman signup

	2. Masukkan nama, username, dan password 3. Tekan register 4. Validasi input (valid) 5. Validasi password (tidak valid) 2. Masukkan nama, username, dan password 3. Tekan register 4. Validasi input (valid) 5. Validasi password (valid) 6. Terima response ke API 7. Validasi response (valid) 8. Berhasil, pindah ke Login
Hasil pengujian	Berhasil
Path	**4**
Jalur	1-2-3-4-5-6-7-2-3-4-5-6-7-8
Skenario	1. Halaman signup 2. Masukkan nama, username, dan password 3. Tekan register 4. Validasi input (valid) 5. Validasi password (valid) 6. Terima response ke API 7. Validasi response (tidak valid) 2. Masukkan nama, username, dan password 3. Tekan register 4. Validasi input (valid) 5. Validasi password (valid) 6. Terima response ke API 7. Validasi response (valid) 8. Berhasil, pindah ke Login
Hasil pengujian	Berhasil

Setelah diketahui jumlah independent path, maka akan dilakukan perbandingan menggunakan tabel hubungan antara cyclomatic complexity dan risiko pada tabel 11.

Tabel 11. Hubungan antara Cyclomatic Complexity Fitur Daftar

V(G)	Deskripsi	Risiko
1 – 4	Prosedur sederhana	Rendah
5 – 10	Prosedur terstruktur & stabil	Rendah
11 – 20	Prosedur agak kompleks	Menengah
21 – 50	Prosedur kompleks & kritis	Menengah
> 50	Sangat rentan kesalahan	Sangat Tinggi

Berdasarkan tabel di atas, fitur Sign Up masuk ke kategori prosedur sederhana karena tingkat risikonya rendah karena jumlah independent path adalah 4. Langkah selanjutnya adalah membuat Test Case berdasarkan independent path. Test case untuk fitur dapat dilihat pada Tabel 12.

Tabel 12. Test Case

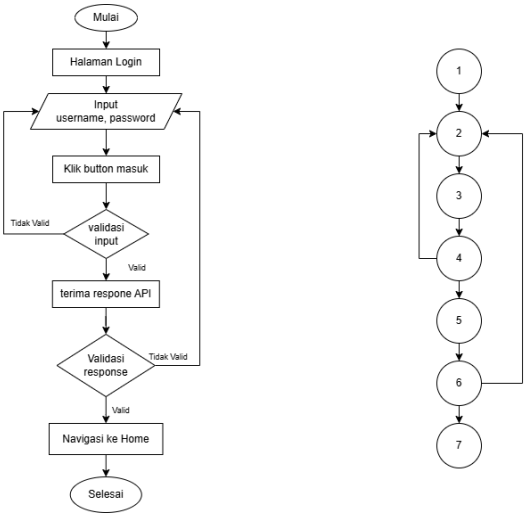
No	Nama Skenario	Kegiatan	Hasil yang diharapkan	Hasil	Keterangan
1	UR1	Pengguna masuk ke halaman sign up dan memasukkan nama, username, dan password yang belum pernah didaftar, kemudian pengguna menekan “Buat akun”	Berhasil mendaftarkan nama, username, dan password ke database, lalu pengguna diarahkan ke halaman login.	Sistem menampilkan pesan “Pendaftaran berhasil“ sebagai tanda bahwa pendaftaran telah sukses dan kemudian diarahkan ke halaman login	Valid
2	UR2	Pengguna masuk ke halaman sign up dan tidak memasukkan nama, username, dan/atau password yang belum pernah didaftar, kemudian pengguna menekan “Buat akun”	Muncul pesan error karena tidak mengisi semua kolom input. Setelah Kembali mengisi input, maka berhasil mendaftarkan nama, username, dan password ke database, lalu pengguna diarahkan ke halaman login.	Sistem menampilkan pesan “Semua Field harus disi“ sebagai tanda bahwa tidak semua kolom terisi, kemudian saat berhasil, akan diarahkan ke halaman login.	Valid
3	UR3	Pengguna masuk ke halaman sign up dan memasukkan nama, username, dan password dengan jumlah password kurang dari 6 karakter, kemudian pengguna menekan “Buat akun”	Muncul pesan error karena jumlah karakter kurang dari 6 karakter. Setelah Kembali mengisi input, maka berhasil mendaftarkan nama, username, dan password ke database, lalu pengguna diarahkan ke halaman login.	Sistem menampilkan pesan “Password harus minimal 6 karakter“ sebagai tanda bahwa jumlah karakter password kurang dari 6 karakter, kemudian saat berhasil, akan diarahkan ke halaman login.	Valid
4	UR4	Pengguna masuk ke halaman sign up dan memasukkan nama, username, dan/atau password dengan nama username yang pernah didaftar sebelumnya,	Muncul pesan error karena terdapat isi dari kolom username telah terdaftar ke database sebelumnya. Setelah Kembali mengisi input, maka berhasil mendaftarkan nama, username, dan password ke	Sistem menampilkan pesan “username sudah digunakan“ sebagai tanda bahwa kolom username berisikan input yang telah terdaftar, kemudian saat	Valid

kemudian pengguna menekan “Buat akun”	database, lalu pengguna diarahkan ke halaman login.	berhasil, akan diarahkan ke halaman login.

Pada tabel test case diatas yang dibuat berdasarkan *independent path* yang telah dibuat, hasil keempat pengujian fitur sign up yang dilakukan valid sehingga tidak ditemukan error. Fitur yang diuji kedua adalah fitur masuk, kode program untuk pengujian fitur masuk dapat dilihat pada Gambar 11.

Gambar 11. Kode Program Fitur Masuk

Flowchart dan flowgraph fitur daftar dapat dilihat pada Gambar 12.



Gambar 12. Flowchart dan Flowgraph Fitur Masuk

Cyclomatic Complexity:

N = 7
E = 8
V(G) = (E−N)+2
= 8−7+2 = 3

Keterangan:
V(G) = Jumlah Region
E = Jumlah edge yang ditentukan dengan gambar panah
N = Jumlah node dengan gambar lingkaran

independent path:
Jalur 1 = 1-2-3-4-5-6-7
Jalur 2 = 1-2-3-4-2-3-4-5-6-7
Jalur 3 = 1-2-3-4-5-6-2-3-4-5-6-7

Tabel pengujian path fitur masuk dapat dilihat pada Tabel 13.

Tabel 13. Pengujian Path Fitur Masuk

Path	**1**
Jalur	1-2-3-4-5-6-7
Skenario	1. Halaman login 2. Masukkan username dan password 3. Tekan masuk 4. Validasi input (valid) 5. Terima respone ke API 6. Validasi response (valid) 7. Berhasil, diarahkan ke Home
Hasil pengujian	Berhasil
Path	**2**
Jalur	1-2-3-4-2-3-4-5-6-7
Skenario	1. Halaman login 2. Masukkan username dan password 3. Tekan masuk 4. Validasi input (tidak valid) 2. Masukkan username dan password 3. Tekan masuk 4. Validasi input (valid) 5. Terima respone ke api 6. Validasi response (valid) 7. Berhasil, diarahkan ke home
Hasil pengujian	Berhasil
Path	**3**
Jalur	1-2-3-4-5-6-2-3-4-5-6-7
Skenario	1. Halaman login 2. Masukkan username dan password 3. Tekan masuk 4. Validasi input (valid)

	5. Terima respone ke api 6. Validasi response (tidak valid) 2. Masukkan username dan password 3. Tekan masuk 4. Validasi input (valid) 5. Terima respone ke api 6. Validasi response (valid) 7. Berhasil, diarahkan ke home
Hasil pengujian	Berhasil

Setelah diketahui jumlah *independent path*, maka akan dilakukan perbandingan menggunakan tabel hubungan antara *cyclomatic complexity* dan risiko pada tabel 14.

Tabel 14. Hubungan antara Cyclomatic Complexity Fitur Masuk

V(G)	Deskripsi	Risiko
1 – 4	Prosedur sederhana	Rendah
5 – 10	Prosedur terstruktur & stabil	Rendah
11 – 20	Prosedur agak kompleks	Menengah
21 – 50	Prosedur kompleks & kritis	Menengah
> 50	Sangat rentan kesalahan	Sangat Tinggi

Berdasarkan tabel di atas, fitur Login masuk ke kategori prosedur sederhana karena Tingkat risikonya rendah karena jumlah *independent path* adalah 3. Langkah selanjutnya adalah membuat *Test Case* berdasarkan *independent path*. *Test case* untuk fitur ini dapat dilihat pada Tabel 15.

Tabel 15. Test Case Fitur Masuk

No	Nama Skenario	Kegiatan	Hasil yang diharapkan	Hasil	Keterangan
1	UL1	Pengguna masuk ke halaman login dan memasukkan username dan password yang telah didaftar sebelumnya, kemudian pengguna menekan “Masuk”	Proses masuk berhasil dan pengguna diarahkan ke Home.	Sistem menampilkan pesan “Login Berhasil “ sebagai tanda bahwa proses masuk berhasil, kemudian diarahkan ke halaman Home	Valid
2	UL2	Pengguna masuk ke halaman login dan memasukkan username dan password yang tidak sesuai, kemudian	Muncul pesan error bahwa input untuk password tidak sesuai dengan database. Setelah Kembali mengisi input,	Sistem menampilkan pesan “Password salah“ sebagai tanda bahwa password tidak	Valid

		pengguna menekan "Masuk"	maka proses masuk berhasil, lalu pengguna diarahkan ke halaman home.	sesuai, kemudian saat berhasil, akan diarahkan ke halaman home.	
3	UL3	Pengguna masuk ke halaman sign up dan memasukkan username dan password yang belum pernah didaftar sebelumnya, kemudian pengguna menekan "Masuk"	Muncul pesan error karena terdapat isi dari kolom username belum terdaftar di database. Setelah Kembali mengisi input, maka proses masuk berhasil, lalu pengguna diarahkan ke halaman home.	Sistem menampilkan pesan "Nama pengguna tidak ditemukan" sebagai tanda bahwa username belum terdaftar, kemudian saat berhasil, akan diarahkan ke halaman home.	Valid

Pada Tabel 15 berdasarkan *independent path* yang telah dibuat, hasil ketiga pengujian fitur login yang dilakukan valid sehingga tidak ditemukan error.

4. Pengujian ISO/EIC 25010

Di tahap ini kami melakukan pengujian menggunakan metode ISO/EIC 25010, Untuk menguji aspek fungsional , Reliability,Usability dan performance pada aplikasi pomify.

1. Functional Suitability

Evaluasi terhadap aspek *Functional Suitability* dilakukan untuk menilai kualitas aplikasi Pomify dalam hal kesesuaian fitur-fitur yang tersedia dengan kebutuhan pengguna. Evaluasi ini juga bertujuan untuk memastikan bahwa setiap fungsi yang disediakan dapat berjalan dengan baik dan mendukung pengguna dalam menyelesaikan tugas secara efektif. Proses pengujian dilakukan melalui penyebaran kuesioner yang terdiri dari lima pernyataan, dan diisi oleh 30 responden sebagai sampel pengguna aplikasi.Rangkuman penilaian bisa dilihat pada Tabel 16.

Tabel 16. Penilaian *Functional Suitability*

No	Pernyataan	Jumlah "Ya"	Jumlah"Tidak"	Persentase
1	Fitur masuk dan daftar dapat berjalan dengan baik	30	0	100%
2	Timer pomodoro berjalan dengan baik	28	2	93.3%
3	Fitur Menambahkan tugas dapat berfungsi dengan baik	29	1	96.7%
4	Fitur menyesuaikan waktu setiap sesi berfungsi dengan baik	28	2	93.3%
5	Fitur report menampilkan statistik akumulasi durasi pomodoro yang di lakukan dan Tugas yang telah selesai	30	0	100%

	dalam rentang Hari ini,Minggu ini, Bulan ini

Berdasarkan hasil rekapitulasi yang disajikan pada Tabel [nomor tabel], dapat diketahui bahwa seluruh fitur yang direncanakan dalam aplikasi Pomify telah berfungsi sebagaimana mestinya menurut persepsi pengguna. Penilaian ini dilakukan dengan menggunakan rumus rata-rata kelengkapan fitur sebagai berikut:

$$x = \frac{I}{C} \times 100\%$$

Keterangan :

I = Jumlah fitur yang berfungsi (diterima oleh pengguna)

C = Jumlah fitur yang direncanakan

Lalu subtitusi nilai ke dalam rumus :

$$x = \frac{5}{5} \times 100\% = 100\%$$

Hasil ini menunjukkan bahwa seluruh fitur utama yang telah dirancang dan diimplementasikan berhasil berfungsi dengan baik dan sesuai dengan ekspektasi pengguna. Nilai kelengkapan fitur sebesar 100% ini mencerminkan bahwa aspek Functional Suitability aplikasi Pomify berada dalam kategori sangat baik. Dengan demikian, dapat disimpulkan bahwa seluruh fungsi yang tersedia pada aplikasi telah memenuhi kebutuhan pengguna dan mendukung tujuan utama aplikasi, yaitu membantu manajemen waktu dan mengurangi prokrastinasi.

2. Usability

Pengujian *Usability* dilakukan dengan menyebarkan kuisioner kepada 30 responden dengan 5 pernyataan. Hasil analisis bisa dilihat pada Tabel 17.

Tabel 17. Hasil Analisis Usability

No	persentase	skor	skor maksimal	jumlah
1	84%	21	25	3
2	88%	22	25	7
3	92%	23	25	6
4	96%	24	25	11
5	100%	25	25	3
Rata Rata		705	750	91.2%

Berdasarkan hasil analisis pada Tabel 17, diperoleh total akumulasi skor usabilitas sebesar 705 dari skor maksimum 750, dengan persentase capaian rata-rata sebesar 91,2%. Nilai tersebut menunjukkan bahwa sistem telah memenuhi kriteria usabilitas secara optimal berdasarkan parameter-parameter evaluasi yang meliputi usefulness (kemanfaatan), ease of use (kemudahan penggunaan), ease of learning (kemudahan pembelajaran), dan satisfaction (kepuasan pengguna).

3. Reliability

Pengujiian Reliability pada aplikasi, dilakukan untuk mengukur sejauh mana aplikasi dapat berfungsi secara konsisten dan stabil dalam kondisi nyata saat digunakan oleh pengguna. Dalam

rangka mengumpulkan data mengenai keandalan aplikasi, sebuah kuesioner dengan 5 pernyataan sesusai dengan aspek reliabilitas disebarkan kepada 30 pengguna aplikasi. Kuesioner ini dirancang untuk menilai aspek maturity, fault tolerance, recoverability, availability, dan eror rate. Hasil dari penilaiannya bisa dilihat pada Tabel 18.

Tabel 18. Hasil Penilaian *Reliability*

No	Pernyataan	Sangat tidak setuju	Tidak setuju	Netral	Setuju	Sangat setuju
1	Aplikasi pomify berjalan stabil tanpa sering error	0	0	0	15	15
2	Aplikasi dapat diakses kapan pun saat ingin digunakan	0	0	0	12	18
3	Aplikasi memberikan notifikasi saat menginput field kosong	0	0	0	15	15
4	Data tugas tetap tersedia walau terjadi force close	0	0	0	10	20
5	Setelah error, aplikasi dapat kembali berfungsi normal dengan cepat.	0	0	0	19	11

Berdasarkan hasil penilaian pada Tabel 18, kita dapat memperoleh nilai persentase reliability dengan membandingkan jumlah total jawaban positif (kategori Setuju dan Sangat Setuju) terhadap jumlah total maksimum skor yang mungkin dicapai, kemudian dikalikan dengan 100%. Rumus yang digunakan adalah sebagai berikut:

$$Persentase = \frac{(Jumlah\ jawaban\ setuju + Sangat\ setuju)}{(Jumlah\ responden\ x\ jumlah\ pernyataan)} \times 100\%$$

$$Persentase = \frac{(71 + 79)}{(30 \times 5)} \times 100\% = 98.67\%$$

Perhitungan ini dilakukan untuk mengevaluasi sejauh mana aspek reliability pada aplikasi Pomify dirasakan oleh pengguna. Evaluasi mencakup sejumlah indikator utama seperti maturity (tingkat kematangan sistem), fault tolerance (kemampuan sistem dalam menghadapi kesalahan), recoverability (kemampuan pemulihan setelah terjadi gangguan), availability (ketersediaan sistem), dan error rate (tingkat kesalahan). Hasil evaluasi menunjukkan nilai sebesar 98,67%, yang mengindikasikan bahwa aplikasi Pomify telah menunjukkan performa yang sangat baik dalam memenuhi ekspektasi pengguna dari sisi keandalan sistem.

4. Performance Eficiency

Berdasarkan pengujian yang dilakukan menggunakan Android Profiler untuk menganalisis penggunaan CPU, konsumsi RAM, dan waktu pemuatan aplikasi, diperoleh nilai rata-rata yang mencerminkan kinerja efisiensi aplikasi Pomify. Setelah melakukan pengujian selama 6 menit dengan menggunakan aplikasi Pomify, dengan fitur timer aktif dan menambahkan tugas, diperoleh data sebagai berikut:

a. CPU Usage

Penggunaan CPU tertinggi yang tercatat selama pengujian adalah 8%, seperti pada Gambar 13.

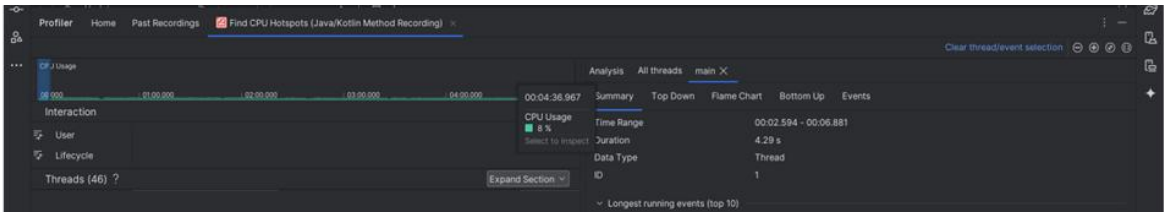


Figure 13. Penggunaan CPU Tertinggi

Penggunaan CPU terendah selama pengujian tercatat pada 4%, yang dapat dilihat pada Gambar 14

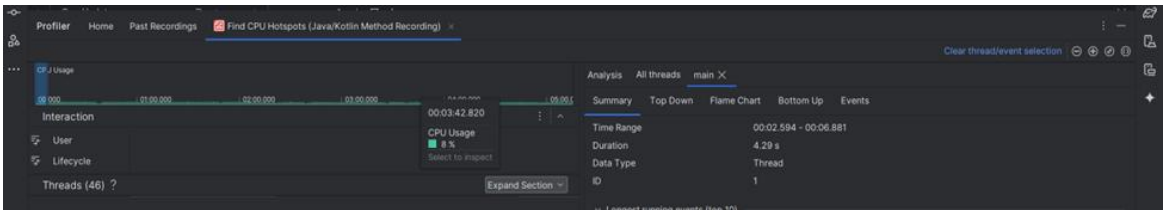


Figure 14. Penggunaan CPU Terendah

Secara keseluruhan, rata-rata penggunaan CPU dalam mode idle dengan timer aktif adalah 8% selama periode penggunaan aplikasi yang berlangsung selama 6 menit.

b. RAM Consumption

Konsumsi RAM saat aplikasi pertama kali dibuka dan berada dalam kondisi idle tercatat sebesar 65MB, sebagaimana ditunjukkan pada Gambar 15.

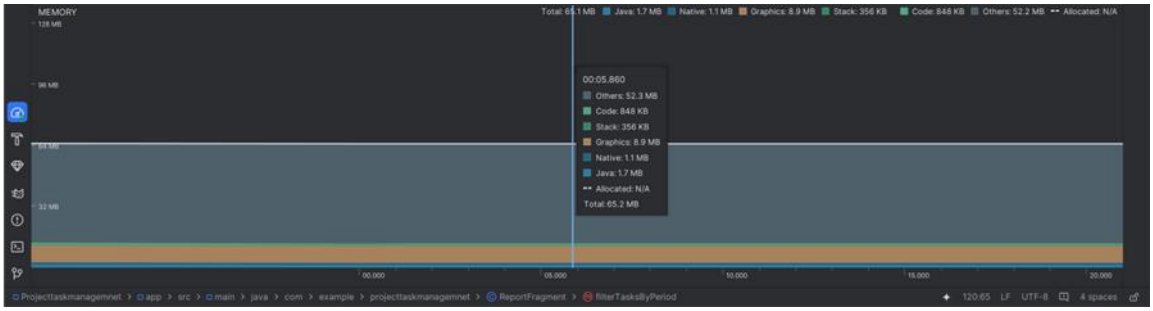


Figure 15. Konsumsi RAM saat Aplikasi Pertama Kali Dibuka

Pada saat pengguna melakukan proses penambahan tugas ke database, terjadi lonjakan konsumsi RAM yang signifikan. Nilainya meningkat dari sebelumnya 65MB menjadi 117MB, seperti terlihat pada Gambar 16.

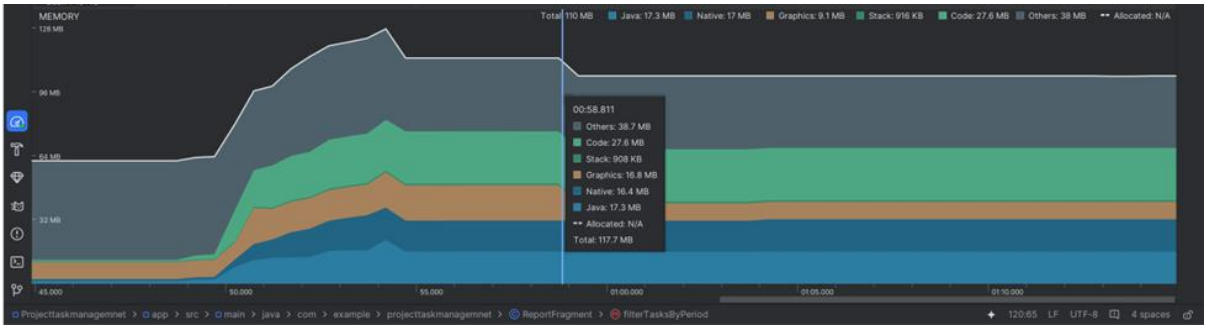
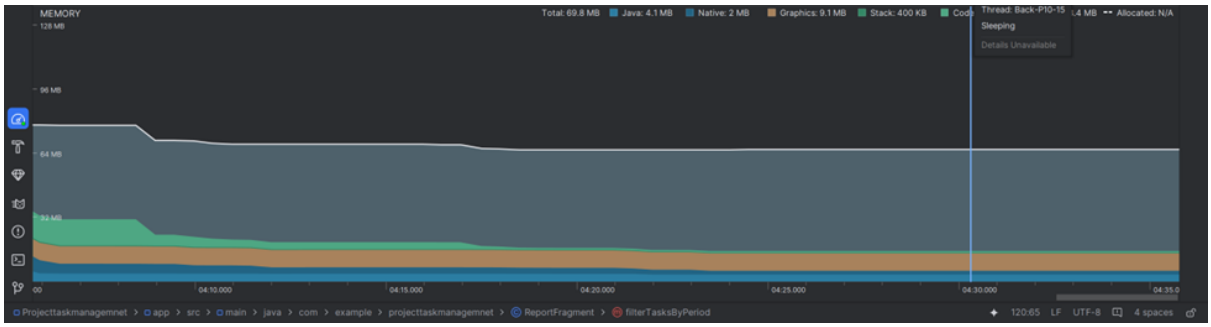


Figure 16. Konsumsi RAM saat Penambahan Tugas

Bisa dilihat di Gambar 17 Lonjakan tersebut bersifat sementara, hanya berlangsung selama proses penambahan tugas berlangsung. Setelah proses tersebut selesai, konsumsi RAM menurun kembali menjadi 69MB dan tetap stabil pada nilai tersebut, sebagaimana ditampilkan pada Gambar 17.



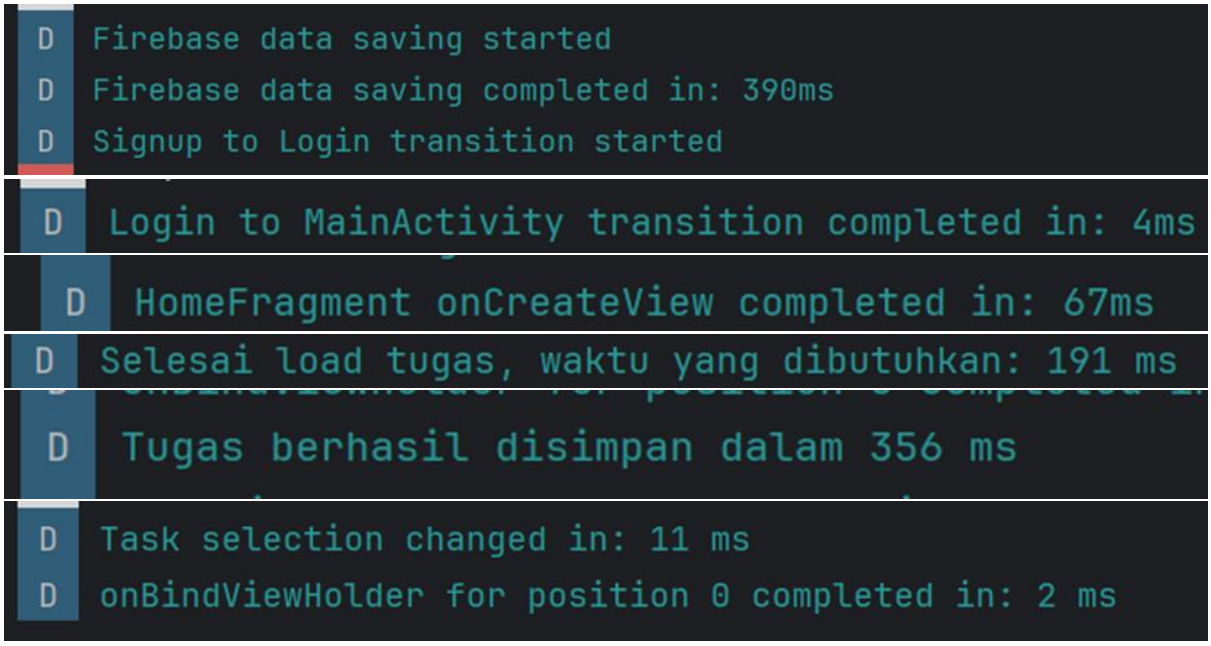
Gambar 17. Konsumsi RAM setelah Penambahan Tugas

Dari hasil pengamatan konsumsi RAM, dapat ditunjukkan bahwa aplikasi Pomify memiliki pengelolaan memori yang cukup efisien. Meskipun terjadi lonjakan saat eksekusi fungsi tertentu seperti penambahan tugas ke database, sistem mampu menstabilkan penggunaan RAM dengan cepat setelah proses selesai.

c. Load Time

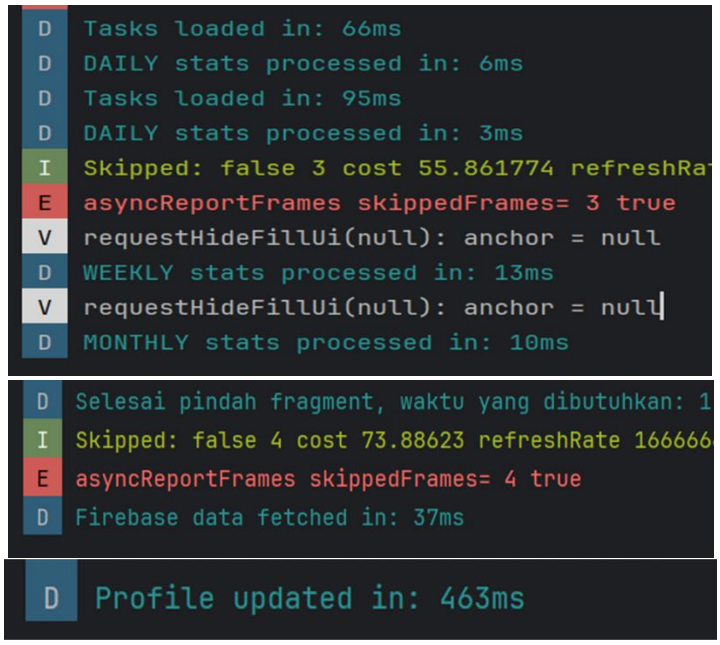
Waktu pemuatan dari halaman Login ke Sign Up tercatat sebesar 20 milidetik (ms). Waktu yang dibutuhkan untuk menyimpan data berupa Nama Pengguna, Nama, dan Kata Sandi ke dalam database adalah 390 ms. Sementara itu, waktu pemuatan dari halaman Login ke MainActivity tercatat lebih cepat, yaitu hanya 4 ms. Waktu untuk halaman HomeFragment memuat seluruh elemen UI adalah 67 ms. Waktu yang diperlukan untuk memuat tugas yang belum diselesaikan dari database ke UI adalah 191 ms. Proses penambahan tugas baru membutuhkan waktu 356 ms untuk menyimpan data ke database. Untuk mengatur tugas menjadi aktif, waktu yang dibutuhkan adalah 11 ms. Terlihat pada Gambar 18.

```
Login to Signup transition completed in: 20ms
```



Gambar 18. Load Time

Selanjutnya di bagian fragment report Waktu untuk memuat Data report hari ini dari database memakan waktu 6 ms untuk pecobaan pertama dan 6 ms untuk percobaan kedua,lalu di bagian minggu ini memakan waktu yang lebih lama dari report hari ini yaitu 13ms,lalu di bagian Bulan ini memakan waktu 10ms untuk dimuat. Di bagian fragment Profile, waktu untuk mengambil data Nama pengguna dari database dan menampilkannya adalah 37 ms. Sedangkan waktu untuk mengubah nama pengguna di profil, yang mencakup proses penyimpanan dan penampilan ulang, tercatat sebesar 463 ms. Terlihat pada Gambar 19.



Gambar 19. Load Time Fragment Report dan Profil

Pengujian load time pada aplikasi Pomify menunjukkan kinerja yang efisien dan responsif. Waktu pemuatan antar halaman seperti dari Login ke Sign Up (20 ms) dan Login ke MainActivity

(4 ms) menunjukkan transisi yang sangat cepat. Pemuatan elemen antarmuka seperti pada HomeFragment (67 ms) serta pengambilan data tugas dari database (191 ms) juga berlangsung dalam durasi yang wajar. Proses yang melibatkan interaksi langsung dengan database, seperti penyimpanan data pengguna (390 ms), penambahan tugas (356 ms), dan pengubahan nama profil (463 ms), meskipun memiliki waktu eksekusi yang lebih tinggi, masih berada dalam batas toleransi standar aplikasi mobile.

Secara keseluruhan, sebagian besar aktivitas dalam aplikasi Pomify memiliki waktu respons di bawah 200 milidetik, yang menunjukkan bahwa aplikasi ini telah memenuhi standar efisiensi kinerja, baik dalam pemrosesan antarmuka pengguna (UI) maupun komunikasi dengan database. Mengacu pada Tabel 6, di mana waktu pemuatan (load time) dikategorikan baik jika berada di bawah 1 detik, hasil pengujian menunjukkan bahwa waktu pemuatan terlama yang dicapai oleh aplikasi hanya sebesar 463 milidetik. Hal ini mengindikasikan bahwa performa load time aplikasi Pomify tergolong sangat baik dan responsif.

5. Ringkasan Evaluasi *ISO/IEC 25010*

Ringkasan evaluasi kualitas aplikasi Pomify dilakukan berdasarkan empat aspek utama: functional suitability (100%), usability (91.2%), reliability (98.67%), dan performance efficiency (95%). Keempat aspek ini divisualisasikan dalam bentuk diagram dan pie chart berikut untuk memberikan gambaran proporsional tingkat pencapaian kualitas aplikasi.

a. Ringkasan hasil *suitability*

Dalam hasil pengujian suitability, diperoleh nilai 100% pada aspek *Functional Suitability*. Penilaian ini didasarkan pada lima pernyataan yang kami ajukan kepada pengguna. Hasilnya dapat dilihat pada Gambar 20, 21, 22, dan 23, yang telah kami visualisasikan dalam bentuk diagram dan pie chart.

Gambar 20 menunjukkan bahwa mayoritas responden menyatakan fitur masuk dan daftar pada aplikasi Pomify berfungsi dengan baik. Hal ini menunjukkan bahwa proses autentikasi pengguna telah dirancang dan diimplementasikan dengan benar tanpa kendala berarti.

Gambar 20. Pie chart pernyataan pertama

Gambar 21 menggambarkan bahwa fitur timer Pomodoro berjalan sesuai harapan. Pengguna dapat menggunakan timer dengan lancar untuk mengatur waktu kerja dan istirahat mereka, menandakan implementasi fungsi inti aplikasi berhasil.

Gambar 21. Pie chart pernyataan kedua

Gambar 22 memperlihatkan bahwa sebagian besar pengguna dapat menambahkan tugas ke dalam aplikasi dengan mudah. Ini menandakan bahwa alur kerja penambahan tugas cukup jelas dan mendukung kebutuhan pengguna dalam perencanaan aktivitas.

Gambar 22. Pie chart pernyataan ketiga

Gambar 23 menunjukkan bahwa pengguna dapat menyesuaikan durasi sesi Pomodoro sesuai kebutuhan mereka. Hal ini mencerminkan fleksibilitas fungsi yang memungkinkan personalisasi manajemen waktu bagi tiap individu.

Gambar 23. Pie chart pernyataan ketiga

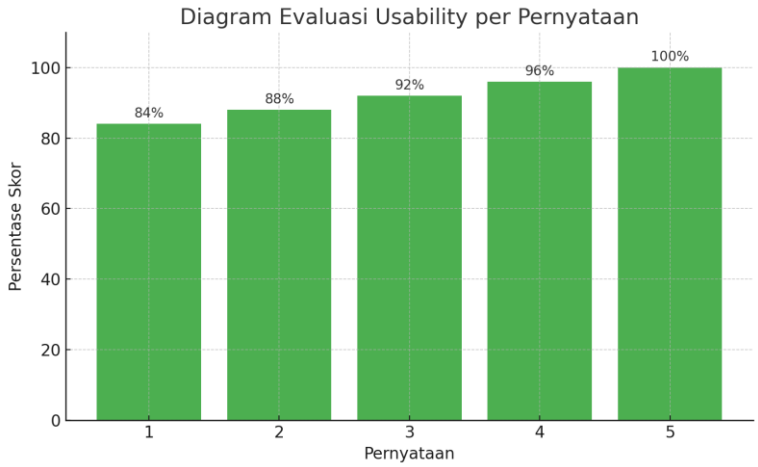
Gambar 24 menggambarkan bahwa fitur laporan berhasil menampilkan statistik akumulasi durasi Pomodoro dan tugas yang telah diselesaikan dalam rentang waktu harian, mingguan, hingga bulanan. Ini menunjukkan bahwa aplikasi tidak hanya

Gambar 24. Pie chart pernyataan kelima

Hasil dari test Suitability menunjukkan hasil yang memuaskan dengan nilai kelengkapan fitur mencapai 100%.

b. Ringkasan hasil usability

Visualisasi pada Gambar 25 dalam bentuk diagram batang menampilkan tingkat pencapaian dari masing-masing indikator usability secara jelas, sehingga memudahkan dalam melihat aspek mana yang paling menonjol dan mana yang masih dapat ditingkatkan. Dengan hasil ini, Pomify dinilai sangat layak dalam hal usability untuk digunakan secara luas oleh pengguna.

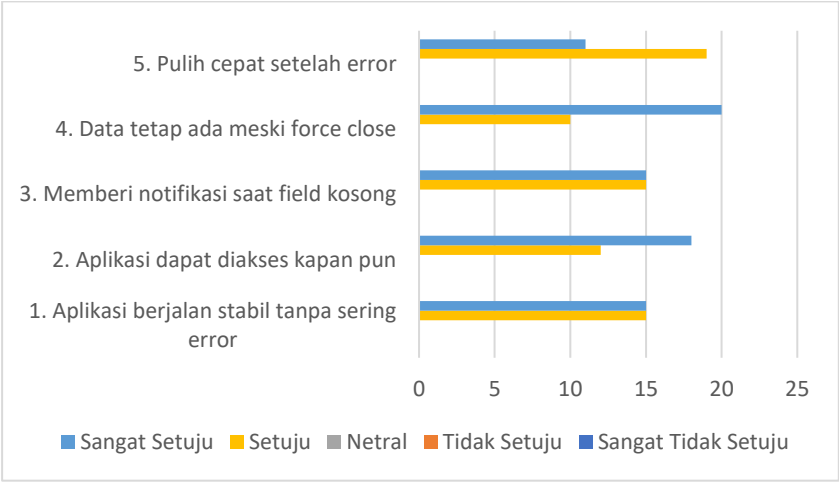


Gambar 25. Diagram batang skor pernytaan hasil *usability*

Evaluasi usability dilakukan berdasarkan lima pernyataan yang mencerminkan kemudahan penggunaan aplikasi Pomify. Hasil menunjukkan skor rata-rata sebesar 91,20%, dengan skor tiap pernyataan berada di kisaran 84% hingga 100%.Visualisasi hasil ditampilkan pada Gambar 25 dalam bentuk diagram batang, yang menunjukkan tingkat pencapaian usability dari masing-masing pernyataan.

c. ringkasan hasil reliability

Hasil pengujian reliability aplikasi Pomify diperoleh melalui survei terhadap 30 responden. Mereka diminta untuk memberikan penilaian terhadap lima pernyataan terkait kestabilan, notifikasi, dan pemulihan aplikasi setelah terjadi error.Setiap responden memilih salah satu dari lima pilihan skala Likert: Sangat Tidak Setuju, Tidak Setuju, Netral, Setuju, dan Sangat Setuju.Data hasil penilaian tersebut kemudian ditampilkan dalam bentuk diagram batang kelompok horizontal (clustered bar chart) di gambar untuk memberikan visualisasi yang lebih jelas mengenai distribusi jawaban dari masing-masing pernyataan.

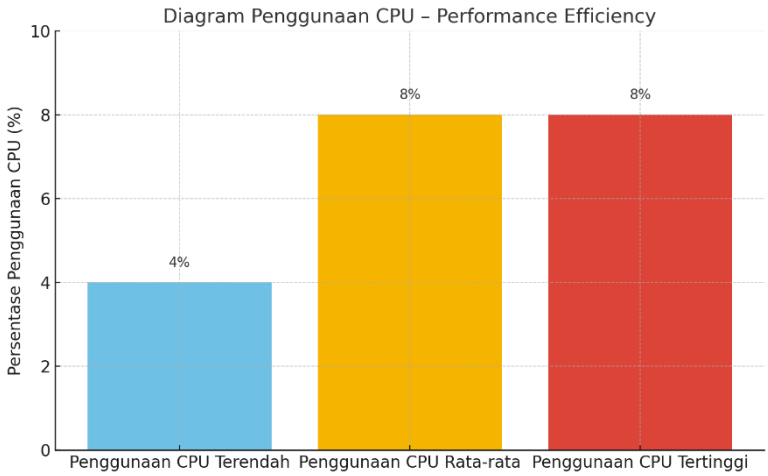


Gambar 26. Diagram batang kelompok dari hasil reliability

Hasil pengujian reliabilitas dari gambar di atas menunjukkan bahwa seluruh responden memberikan penilaian positif terhadap lima aspek yang diuji. Mayoritas responden menyatakan setuju dan sangat setuju bahwa aplikasi berjalan stabil, dapat diakses kapan saja, memberikan notifikasi saat input tidak lengkap, mampu menyimpan data meskipun terjadi force close, dan dapat pulih dengan cepat setelah error. Tidak ada responden yang memberikan penilaian netral atau negatif, yang mengindikasikan bahwa aplikasi Pomify memiliki tingkat reliabilitas yang sangat baik.

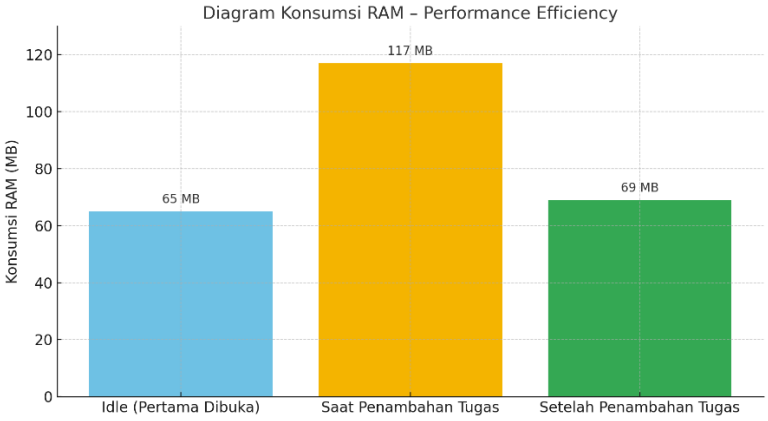
d. d.ringkasan hasil performance efficiency

Hasil pengujian performance efficiency aplikasi Pomify menunjukkan kinerja yang efisien dan stabil di berbagai aspek teknis. Pada aspek CPU usage, penggunaan CPU rata-rata berada di angka 8% dalam kondisi idle dengan timer aktif, dengan nilai tertinggi tidak melebihi 8%, yang menunjukkan bahwa aplikasi tidak membebani sistem secara berlebihan.Dan hasilnya bisa dilihat pada gambar 27.



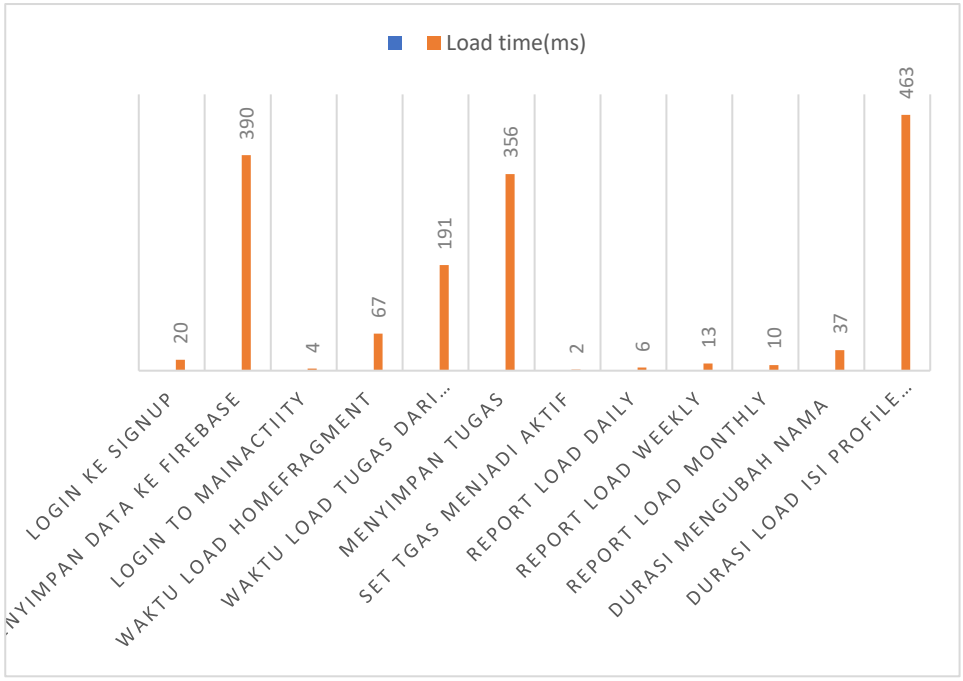
Gambar 27. Diagram batang CPU *efficiency*

Hasil ini mencerminkan efisiensi CPU yang sangat baik, di mana aplikasi Pomify mampu mempertahankan konsumsi daya komputasi yang rendah meskipun dalam keadaan aktif secara fungsional (dengan timer berjalan). Hal ini menandakan bahwa proses latar belakang dan pengelolaan event dalam aplikasi dioptimalkan dengan baik, sehingga tidak terjadi lonjakan penggunaan sumber daya yang tidak perlu. Dengan rata-rata pemakaian CPU yang konsisten di bawah 10%, Pomify menunjukkan bahwa aplikasi ini dirancang dengan mempertimbangkan efisiensi dan kenyamanan pengguna.Dan untuk RAM comsumption nya bisa dilihat pada gambar 28.



Gambar 28. Diagram batang RAM comsumption

Secara keseluruhan, pola konsumsi RAM yang ditunjukkan selama pengujian mengindikasikan bahwa aplikasi Pomify dirancang dengan manajemen memori yang baik. Lonjakan memori yang terjadi saat penambahan tugas merupakan respons normal terhadap aktivitas pemrosesan data, dan yang terpenting, aplikasi mampu mengembalikan penggunaan RAM ke kondisi stabil dalam waktu singkat. Hal ini menunjukkan bahwa tidak terdapat kebocoran memori atau penggunaan sumber daya yang berlebihan setelah proses selesai.



Gambar 29. Load Time

Berdasarkan gambar di atas, waktu muat (load time) aplikasi Pomify menunjukkan bahwa aplikasi ini telah memenuhi standar efisiensi kinerja, karena seluruh durasi waktu muat berada di bawah 1 detik.

KESIMPULAN DAN SARAN

Berdasarkan hasil dan diskusi yang telah dilakukan, aplikasi Pomify yang dikembangkan menggunakan teknik Pomodoro berhasil memenuhi kebutuhan fungsional pengguna dalam manajemen waktu dan tugas. Aplikasi ini dirancang berdasarkan hasil pengumpulan kebutuhan melalui wawancara terstruktur, yang kemudian diterjemahkan ke dalam fitur seperti pengaturan tugas, timer Pomodoro, laporan aktivitas, registrasi/login, dan navigasi yang intuitif. Proses perencanaan dan pemodelan menggunakan diagram UML (Use Case, Activity, Sequence, dan Class) memberikan dasar yang kuat dalam memahami alur kerja sistem, sementara tahap konstruksi menggunakan Java dan Android Studio serta backend Firebase memastikan skalabilitas dan keamanan data. Pengujian fungsional dan kompatibilitas membuktikan bahwa aplikasi berjalan lancar di berbagai perangkat Android, dan pengujian whitebox menunjukkan bahwa logika program berjalan baik dengan tingkat kompleksitas yang rendah. Evaluasi berdasarkan standar ISO/IEC 25010 menunjukkan hasil positif pada aspek functional suitability (100% kelengkapan fitur), usability (skor rata-rata 91,2%), reliability (98,67%), dan performance efficiency (penggunaan sumber daya efisien).

Untuk pengembangan selanjutnya, disarankan agar Pomify memperluas pengujian kompatibilitas pada berbagai perangkat dan versi Android, serta mengoptimalkan konsumsi RAM saat penambahan tugas. Integrasi dengan kalender digital atau aplikasi produktivitas juga dapat memperluas fungsionalitas aplikasi. Studi ini memiliki keterbatasan pada pengujian yang hanya dilakukan pada dua perangkat dan belum mencakup uji ketahanan jangka panjang. Selain itu, aspek penting dalam standar ISO/IEC 25010 seperti portability, security, maintainability, dan compatibility belum diuji, padahal hal ini penting untuk keberlanjutan dan kesiapan aplikasi dalam skala yang lebih luas. Dengan pencapaian dan potensi pengembangannya, Pomify memiliki prospek menjanjikan sebagai aplikasi manajemen waktu yang efektif dalam meningkatkan produktivitas dan mengurangi prokrastinasi.

REFERENSI

[1] M. E. Lianto, C. H. Primasari, E. Marsella, Y. P. Wibisono, and M. Cininta, “Evaluasi functional suitability, performance efficiency, usability, dan portability berdasarkan ISO 25010 pada aplikasi vr gamelan slenthem,” KONSTELASI: Konvergensi Teknologi dan Sistem Informasi, vol. 3, no. 1, pp. 24–36, 2023, doi: 10.24002/konstelasi.v3i1.6620.

[2] M. Karengke, H. Surasa, and B. Zaman, “Evaluasi penggunaan website renovaction menggunakan metode usability testing,” JTRISTE, vol. 9, no. 1, pp. 83–97, 2022, doi: 10.55645/jtriste.v9i1.366.

[3] A. Bela, S. Thohiroh, Y. R. Efendi, and S. Rahman, “Prokrastinasi akademik dan manajemen waktu terhadap stres akademik pada mahasiswa di masa pandemi: review literatur,” Jurnal Psikologi Wijaya Putra (Psikowipa), vol. 4, no. 1, pp. 37–48, 2023.

[4] J. Hartati, W. Achadi, S. Syarnubi, and M. M. Naufa, “Hubungan prokrastinasi dan dukungan sosial teman sebaya pada mahasiswa pendidikan agama islam fitk uin raden patah palembang,” Al-Mada: Jurnal Agama, Sosial, dan Budaya, vol. 5, no. 4, pp. 608–618, 2022.

[5] A. M. Dharma, “Prokrastinasi akademik di kalangan mahasiswa program studi dharma acarya,” Jurnal Pendidikan, Sains Sosial, dan Agama, vol. 6, no. 1, pp. 64–78, 2020.

[6] Y. Kartika and A. A. Azhar, "Analisis self control penggunaan gadget pada perilaku prokrastinasi: studi kasus mahasiswa ilmu komunikasi UIN Sumatera Utara," Jurnal Indonesia: Manajemen Informatika dan Komunikasi, vol. 5, no. 2, pp. 1799–1808, 2024.

[7] M. H. Pradika, A. Sutja, and H. Wahyuni, "Penyebab prokrastinasi mahasiswa pada penyelesaian skripsi Fakultas Peternakan Universitas Jambi," Jurnal Pendidikan Tambusai, vol. 7, no. 1, pp. 329–334, 2023, doi: 10.31004/jptam.v7i1.5298.

[8] S. Yuniar, E. D. Wahyuni, and R. Permatasari, "Rancang bangun sistem informasi pengerjaan tugas berbasis teknik pomodoro menggunakan metode waterfall," Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi, vol. 12, no. 3, pp. 1590–1600, 2023.

[9] M. Z. Z. B. Nasution, M. I. P. Nasution, and S. S. A. Sundari, "Penerapan teknik pomodoro dalam upaya meningkatkan efektifitas belajar mahasiswa pada masa pandemi Covid-19 di kelas sistem informasi-3," Jurnal Inovasi Penelitian, vol. 3, no. 5, pp. 6035–6040, 2022.

[10] I. N. Iman, P. Purwantoro, and A. Suharso, "Rancang bangun sistem manajemen waktu dengan teknik pomodoro menggunakan arsitektur MVVM," JATI (Jurnal Mahasiswa Teknik Informatika), vol. 7, no. 3, pp. 1700–1706, 2023.

[11] W. A. Kartikasari, M. Marjohan, and R. Hariko, "Hubungan self regulated learning dan dukungan orangtua terhadap perilaku prokrastinasi akademik," JRTI (Jurnal Riset Tindakan Indonesia), vol. 7, no. 3, pp. 388–394, 2022.

[12] S. Dimastuti, N. Gutji, and D. Rahmayanty, "Identifikasi faktor-faktor penyebab prokrastinasi akademik siswa: sebuah studi deskriptif pada siswa kelas VIII sekolah menengah pertama," Al-Isyraq: Jurnal Bimbingan, Penyuluhan, dan Konseling Islam, vol. 7, no. 1, pp. 211–220, 2024.

[13] D. I. Aminta, D. B. Santoso, and E. Flurentin, "Prokrastinasi akademik mahasiswa bimbingan dan konseling Universitas Negeri Malang," Jurnal Pembelajaran, Bimbingan, dan Pengelolaan Pendidikan, vol. 3, no. 3, pp. 215–221, 2023.

[14] A. D. Permadi, D. Deslianti, R. Toyib, and P. Pahrizal, "Penerapan model incremental dalam pembangunan sistem informasi penentuan jenis usaha," JUSIBI (Jurnal Sistem Informasi dan E-Bisnis), vol. 5, no. 1, pp. 16–29, 2023, doi: 10.54650/jusibi.v5i1.463.

[15] W. P. Sari, E. Yakub, and K. Khadijah, "Pengaruh bimbingan kelompok dengan teknik self management untuk mengurangi prokrastinasi akademik siswa di MTs," Educational Guidance and Counseling Development Journal, vol. 6, no. 1, pp. 29–36, 2023.

[16] F. N. Ramadha, E. D. Wahyuni, and D. D. Vannes, "SDLC big bang dan waterfall: perbandingan pendekatan dalam pengembangan perangkat lunak," Nuansa Informatika, vol. 18, no. 2, pp. 41–45, 2024.

[17] N. Apriyanti, S. F. A. Wati, and A. R. E. Najaf, "Pemanfaatan metodologi PXP dan pengujian user acceptance testing (UAT) dalam pengembangan website e-kavling," JATI (Jurnal Mahasiswa Teknik Informatika), vol. 8, no. 3, pp. 3678–3686, 2024.

[18] E. Reswara, S. Suakanto, and E. N. Alam, "Pengembangan backend aplikasi presensi karyawan berbasis mobile pada KSPPS Karya Usaha Mandiri dengan metode iterative incremental," Syntax Literate: Jurnal Ilmiah Indonesia, vol. 7, no. 11, pp. 15647–15666, 2022.

[19] R. F. Valentina, N. H. Purnomo, and S. Pramanasari, "Implementasi teknik belajar pomodoro untuk meningkatkan fokus belajar pada mata pelajaran IPS," Pendas: Jurnal Ilmiah Pendidikan Dasar, vol. 9, no. 4, pp. 1676–1685, 2024.

[20] M. Rafliyanto and F. Mukhlis, "Optimizing learning: applying the pomodoro technique in Islamic education," el-Tarbawi, vol. 16, no. 1, pp. 153–180, 2023.

[21] D. A. Wasesha, "Perancangan aplikasi layanan salon hewan secara daring menggunakan model proses incremental," INTI Nusa Mandiri, vol. 16, no. 1, pp. 39–48, 2021.

[22] F. D. Zebua et al., "Perancangan aplikasi Unika Market di lingkungan Universitas Katolik Santo Thomas Medan berbasis web menggunakan metode incremental," in Seminar Nasional Inovasi Sains Teknologi Informasi Komputer, Oct. 2023, pp. 145–150.

[23] M. H. S. Fedianto, F. P. Aditiawan, and M. M. Al Haromainy, "Pengujian sistem jaringan dokumentasi dan informasi menggunakan black box testing dan white box testing," Jurnal Publikasi Sistem Informasi dan Manajemen Bisnis, vol. 3, no. 1, pp. 213–221, 2024.

[24] R. Rafli, F. Fauziah, and R. T. Aldisa, "Aplikasi pengolahan data penjualan pembangkit listrik tenaga surya menggunakan model view controler berbasis framework CodeIgniter dan white box testing," J-SAKTI (Jurnal Sains Komputer dan Informatika), vol. 5, no. 2, pp. 677–686, 2021.

[25] R. Dewi, D. H. Satyareni, and E. Kurniawan, "Implementasi ISO-IEC 25010 untuk analisis kualitas sistem informasi manajemen kerja praktik (SIM-KP)," METHOMIKA: Jurnal Manajemen Informatika & Komputerisasi Akuntansi, vol. 9, no. 1, pp. 76–85, 2025.

[26] V. Bong and M. D. Firmansyah, "Analisa Duolingo terhadap prestasi Bahasa Inggris siswa SMP Batam dengan metode TAM," Jurnal Informasi dan Teknologi, pp. 122–130, 2023.