

Pengembangan Aplikasi Task Reminder Berbasis Web Dengan Crud Fungsional Untuk Produktivitas Pengguna

Fajar Gustri Amanda^{1*}, Muhammad Hilmy Ibrahim², Surya Nugraha Wardana³,
Adryan Maulana⁴

^{1,2,3,4}Fakultas Sains dan Teknologi, Informatika, UIN Sultan Maulana Hasanuddin Banten, Indonesia

Article Info

Article history:

Received December 25, 2024

Revised July 18, 2025

Accepted June 30, 2026

Keywords:

Task Reminder,
CRUD,
Web App,
Produktivitas,
Manajemen Tugas.

ABSTRACT (11 PT)

Manajemen waktu yang efektif menjadi tantangan utama dalam menunjang produktivitas individu di era digital. Penelitian ini mengembangkan aplikasi Task Reminder Web App sebagai solusi berbasis antarmuka web yang berfungsi membantu pengguna dalam mengatur, melacak, dan menyelesaikan rutinitas tugas mereka secara sistematis. Aplikasi ini memanfaatkan integrasi operasi komputasi terapan (*Create, Read, Update, Delete*) guna mendukung fleksibilitas penambahan, pembaruan, serta penghapusan data tugas. Metode penelitian yang diterapkan meliputi tahap analisis kebutuhan fungsional sistem serta perancangan berbasis diagram alir logika. Proses implementasi dilakukan dengan mengolaborasikan bahasa pemrograman PHP, antarmuka Bootstrap responsif, serta sistem manajemen basis data MySQL. Berdasarkan tahapan uji fungsionalitas perangkat lunak, sistem yang dikembangkan dinyatakan berhasil mengakomodasi kebutuhan interaksi manajemen tugas secara responsif dan intuitif. Temuan riset ini mengindikasikan bahwa sistem informasi dapat ditingkatkan efektivitasnya pada pengembangan lanjutan, seperti integrasi klasifikasi kategori berbasis kecerdasan komputasi dan implementasi autentikasi identitas pengguna.

This is an open access article under the [CC-BY](https://creativecommons.org/licenses/by/4.0/) 4.0. license.



Corresponding Author:

Fajar Gustri Amanda

Department of Informatic, Faculty of Saintek, Islamic State Sultan Maulana Hasanuddin
University

Jl Syekh Nawawi Al Bantani Kota Serang Banten, Indonesia

Email: fajargustriamanda11@gmail.com

1. PENDAHULUAN

Pada era kontemporer yang ditandai dengan akselerasi teknologi informasi, produktivitas individu tidak lagi sekadar diukur dari seberapa keras seseorang bekerja, melainkan dari seberapa efisien

mereka mengelola waktu, prioritas, dan sumber daya kognitif yang dimiliki. Arus informasi yang masif melalui berbagai perangkat pintar sering kali menciptakan beban kognitif (*cognitive load*) yang berlebihan. Manusia memiliki keterbatasan biologis dalam kapasitas memori kerja (*working memory*), sehingga mengandalkan ingatan internal untuk melacak tenggat waktu (*deadline*), prioritas, dan rincian tugas sering kali berujung pada kelalaian, stres, dan penurunan kualitas pekerjaan secara signifikan. Dalam konteks psikologi kognitif, ketika otak dipaksa untuk terus-menerus mengingat apa yang harus dilakukan, sumber daya mental yang seharusnya dialokasikan untuk eksekusi dan pemecahan masalah menjadi terkuras. Oleh karena itu, diperlukan sistem eksternal yang berfungsi sebagai memori tambahan yang andal, terorganisir, dan mudah diakses untuk membebaskan kapasitas kognitif tersebut.

Aplikasi berbasis web muncul sebagai pilihan yang sangat strategis dan relevan dalam konteks ini. Berbeda dengan aplikasi *desktop* tradisional yang terikat pada satu sistem operasi dan perangkat keras tertentu, atau aplikasi *mobile native* yang memerlukan instalasi dan pembaruan berkala, aplikasi web menawarkan fleksibilitas lintas platform (*cross-platform compatibility*) yang luar biasa. Pengguna dapat mengakses daftar tugas mereka dari komputer kantor, laptop pribadi, maupun ponsel pintar melalui peramban web modern kapan saja dan di mana saja, selama terhubung dengan jaringan internet atau lokal. Evolusi teknologi web modern, yang didukung oleh standar HTML5, CSS3, dan JavaScript ES6, telah memungkinkan aplikasi web untuk memiliki kinerja dan interaktivitas yang hampir menyamai aplikasi *native*, sekaligus mempertahankan keunggulan dalam hal kemudahan distribusi dan pemeliharaan.

Konsep dasar dari sebagian besar aplikasi manajemen data, termasuk aplikasi pengingat tugas, berpusat pada operasi CRUD (*Create, Read, Update, Delete*). Operasi ini merupakan fondasi absolut dari interaksi pengguna dengan basis data relasional. Namun, banyak aplikasi sederhana yang dikembangkan saat ini mengabaikan aspek *User Experience* (UX) dan desain responsif. Meskipun secara fungsional aplikasi tersebut mampu menyimpan data, pengguna sering kali enggan menggunakannya secara konsisten karena antarmuka yang kaku, tidak intuitif, dan tidak nyaman diakses melalui layar berukuran kecil. Penelitian ini hadir untuk menjembatani kesenjangan antara fungsionalitas CRUD yang *robust* dan antarmuka pengguna yang intuitif, responsif, serta secara psikologis mendukung motivasi penyelesaian tugas melalui indikator visual yang jelas.

Berbagai penelitian telah dilakukan untuk mengeksplorasi pengembangan sistem manajemen tugas dan produktivitas, masing-masing dengan fokus dan keunikannya tersendiri. Dalam konteks pengembangan aplikasi berbasis web untuk manajemen data institusional maupun personal, Wahyudi, Wibowo, Mustaghfiroh, dan Mulasih berhasil mengembangkan aplikasi perpustakaan *E-Book* berbasis web yang menunjukkan bahwa arsitektur web yang terstruktur dengan operasi CRUD yang baik dapat meningkatkan aksesibilitas informasi secara signifikan dan memudahkan pengguna dalam menelusuri koleksi digital [1]. Sejalan dengan itu, Wahyudi, Sunardi, dan Mardian dalam pengembangannya tentang aplikasi manajemen keuangan pesantren berbasis web, menekankan bahwa antarmuka yang responsif dan logika CRUD yang efisien adalah kunci utama dalam penerimaan sistem oleh pengguna, serta memastikan integritas data dalam lingkungan yang membutuhkan akurasi tinggi [2]. Kedua penelitian ini memberikan landasan teoretis dan praktis yang kuat bahwa pendekatan berbasis web dengan fondasi CRUD yang solid sangat layak diadopsi untuk aplikasi manajemen tugas pribadi.

Terkait dengan aspek psikologis dan desain antarmuka, Williams dan Brown dalam penelitiannya yang berfokus pada prinsip desain antarmuka untuk daftar tugas digital dari perspektif beban kognitif, menemukan bahwa alat bantu digital seperti *to-do lists* berfungsi efektif untuk mengurangi beban kognitif eksternal, membebaskan memori kerja otak untuk fokus pada eksekusi tugas itu sendiri [3]. Mereka juga menyoroti bahwa penggunaan elemen visual berbasis warna dapat memberikan umpan balik kognitif yang positif, meningkatkan motivasi pengguna untuk menyelesaikan tugas yang terdaftar. Sementara itu, Kim et al. (2024) menginvestigasi dampak desain web responsif terhadap keterlibatan pengguna dalam alat produktivitas, dan menemukan bahwa implementasi *Responsive Web Design* (RWD) yang tepat meningkatkan keterlibatan pengguna hingga empat puluh persen, karena pengguna tidak perlu melakukan *zooming* atau *scrolling* horizontal yang melelahkan saat berpindah antar perangkat [4].

Dalam domain optimasi kinerja sistem, Nguyen dan Garcia menyoroti pentingnya optimasi operasi CRUD pada database relasional untuk aplikasi web dengan lalu lintas tinggi [5]. Mereka menekankan bahwa efisiensi operasi ini sangat bergantung pada desain skema basis data, penggunaan *index*, dan penulisan query yang optimal. Sejalan dengan temuan tersebut, Martinez dan Chen dalam studi komparatifnya tentang tolok ukur kinerja kerangka kerja PHP untuk sistem manajemen tugas, menunjukkan bahwa PHP, terutama pada versi modern, menawarkan kinerja yang sangat kompetitif untuk aplikasi manajemen tugas ringan dibandingkan dengan kerangka kerja yang lebih berat, karena *overhead* memori yang minimal dan waktu eksekusi yang cepat [6].

Penelitian lain oleh Sagar dan Singh mengembangkan aplikasi web cerdas untuk manajemen tugas yang berfokus pada optimasi produktivitas melalui algoritma penjadwalan otomatis [7]. Meskipun canggih, pendekatan tersebut sering kali terlalu kompleks untuk pengguna awam yang hanya membutuhkan pencatatan sederhana dan cepat. Di sisi lain, Hirpara mengembangkan aplikasi manajemen tugas yang berfokus pada kolaborasi kelompok, namun implementasinya membutuhkan sumber daya server yang cukup besar dan alur kerja yang rumit [8]. Patel dan Sharma membahas pengembangan mikro layanan untuk aplikasi pengingat tugas, yang sangat cocok untuk sistem berskala besar, namun mungkin terlalu berlebihan (*over-engineered*) untuk aplikasi manajemen tugas pribadi yang membutuhkan kecepatan akses instan [9].

Terkait dengan metodologi pengujian dan keamanan, Rahman et al. mengevaluasi efektivitas pengujian *Black-Box* dalam keamanan aplikasi web, menyimpulkan bahwa metode ini sangat krusial untuk memvalidasi fungsionalitas sistem dari perspektif pengguna akhir tanpa perlu membongkar struktur kode internal [10]. Lee et al. menambahkan bahwa integrasi sistem notifikasi *push* dalam pengingat tugas berbasis web dapat meningkatkan tingkat penyelesaian tugas secara signifikan, meskipun hal tersebut menambah kompleksitas pada arsitektur sistem [11].

Berdasarkan latar belakang dan tinjauan literatur di atas, dapat diidentifikasi bahwa meskipun banyak aplikasi manajemen tugas yang tersedia, masih terdapat kesenjangan antara kompleksitas fitur dan kemudahan penggunaan (*ease of use*). Banyak aplikasi menjadi terlalu rumit sehingga menghambat adopsi pengguna, sementara aplikasi yang terlalu sederhana sering kali mengabaikan aspek desain responsif dan umpan balik visual yang mendukung motivasi psikologis. Oleh karena itu, rumusan masalah dalam penelitian ini adalah bagaimana merancang dan mengimplementasikan aplikasi *Task Reminder* berbasis web yang tidak hanya mendukung operasi CRUD secara efisien dan aman, tetapi juga menerapkan desain antarmuka yang responsif dan intuitif untuk memastikan aksesibilitas di berbagai perangkat, serta bagaimana mengevaluasi kinerja sistem dalam memenuhi kebutuhan manajemen tugas pengguna.

Penelitian ini bertujuan untuk merancang dan mengimplementasikan aplikasi berbasis web bernama *Task Reminder Web App* yang mendukung operasi CRUD dengan fungsi utama membantu pengguna menambahkan, membaca, memperbarui, dan menghapus tugas mereka. Selain itu, aplikasi ini bertujuan untuk memberikan pengalaman pengguna yang responsif, mudah dioperasikan, dan mendukung produktivitas melalui integrasi elemen visual yang memberikan umpan balik psikologis positif.

Agar penelitian tetap terarah, fokus, dan mendalam, penulis menetapkan batasan masalah yang ketat. Penelitian ini hanya mencakup perancangan, pengembangan, dan evaluasi fungsionalitas aplikasi berbasis web yang digunakan untuk manajemen tugas pribadi. Fitur yang dikembangkan dibatasi pada operasi CRUD dasar tanpa mencakup fitur autentikasi pengguna multi-level, pendaftaran akun, atau integrasi dengan pihak ketiga seperti kalender eksternal. Teknologi yang digunakan dibatasi pada PHP *native*, MySQL, JavaScript, dan Bootstrap 5, dengan pengujian dilakukan pada lingkungan lokal menggunakan server XAMPP.

2. METODE

2.1 Kerangka dan Desain Penelitian

Penelitian ini mengadopsi pendekatan pengembangan sistem yang terstruktur, mengadaptasi model *Waterfall* yang dimodifikasi menjadi iteratif untuk tahap pengujian dan penyempurnaan

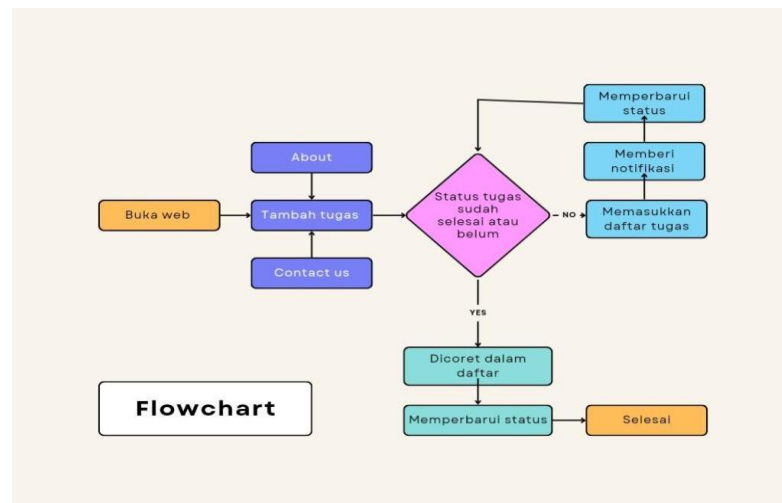
antarmuka. Tahapan penelitian dimulai dari pengumpulan kebutuhan yang mendalam, dilanjutkan dengan analisis dan perancangan sistem yang komprehensif, implementasi kode (*coding*), pengujian sistem secara menyeluruh, dan diakhiri dengan evaluasi serta dokumentasi. Pendekatan ini dipilih karena karakteristiknya yang sistematis dan terukur, memungkinkan setiap fase pengembangan dievaluasi secara independen sebelum melangkah ke fase berikutnya, sehingga risiko kegagalan sistem dapat diminimalisir sejak dini.

2.2 Analisis Kebutuhan Sistem

Tahap analisis kebutuhan merupakan fondasi kritis dalam pengembangan perangkat lunak, di mana peneliti mengidentifikasi dan mendokumentasikan apa yang harus dilakukan oleh sistem. Kebutuhan sistem dikategorikan menjadi dua dimensi utama, yaitu kebutuhan fungsional dan non-fungsional. Kebutuhan fungsional mendefinisikan perilaku spesifik sistem, yang dalam aplikasi ini mencakup kemampuan pengguna untuk menambahkan tugas baru dengan detail nama dan tenggat waktu, melihat daftar semua tugas yang telah diinput, memperbarui status tugas menjadi selesai, serta menghapus tugas yang tidak lagi relevan. Di sisi lain, kebutuhan non-fungsional mengatur atribut kualitas sistem, seperti kewajiban aplikasi untuk tampil responsif pada berbagai perangkat dari desktop hingga smartphone, penyimpanan data yang aman dan persisten di server melalui database MySQL, serta jaminan bahwa waktu eksekusi untuk setiap operasi tidak melebihi satu detik untuk mempertahankan persepsi kecepatan oleh pengguna.

2.3 Perancangan Sistem dan Arsitektur Data

Perancangan sistem dilakukan untuk menerjemahkan kebutuhan yang telah dianalisis menjadi blueprint teknis yang siap diimplementasikan. Diagram konteks sistem disusun untuk memetakan batasan antara aplikasi dan entitas eksternal, yang dalam hal ini hanya melibatkan satu aktor utama yaitu "Pengguna". Pengguna berinteraksi dengan sistem *Task Reminder Web App* dengan memberikan input berupa data tugas, dan sistem merespons dengan menyajikan daftar tugas yang terkelola beserta notifikasi visual statusnya.



Gambar 1. Flowchart Sistem

Selanjutnya, *Flowchart* sistem dirancang untuk merinci logika alur eksekusi. Alur dimulai dari pengguna yang mengakses aplikasi melalui peramban web. Sistem secara otomatis mengeksekusi query pembacaan data dari database, mengurutkannya berdasarkan tenggat waktu, dan merender antarmuka visual. Ketika pengguna memutuskan untuk menambahkan tugas, data dari formulir akan divalidasi di sisi klien menggunakan JavaScript sebelum dikirim ke server. Server kemudian memproses data tersebut, melakukan sanitasi untuk mencegah injeksi SQL, dan mengeksekusi query penyisipan data ke dalam database. Proses serupa terjadi untuk pembaruan status dan penghapusan data, di mana setiap aksi memicu query spesifik yang disesuaikan dengan parameter unik dari tugas yang dipilih.

Perancangan basis data dilakukan menggunakan pendekatan Entity Relationship Diagram (ERD) untuk memastikan integritas dan efisiensi penyimpanan data. Sistem ini dirancang dengan satu entitas

utama bernama *tasks* yang memiliki beberapa atribut kritis. Atribut *id* berfungsi sebagai *Primary Key* dengan tipe data integer dan sifat *auto-increment*, memastikan setiap tugas memiliki identitas unik yang mutlak. Atribut *task_name* menyimpan deskripsi tugas dengan batas karakter yang memadai, sementara *deadline* menyimpan informasi tanggal tenggat waktu. Atribut *status* menggunakan tipe data *tinyint* dengan nilai default nol, yang berfungsi sebagai indikator biner apakah tugas tersebut belum selesai atau telah selesai. Terakhir, atribut *created_at* mencatat *timestamp* kapan tugas tersebut pertama kali dibuat, yang berguna untuk pengurutan sekunder dan keperluan audit di masa mendatang.

2.4 Lingkungan Pengembangan dan Teknologi

Implementasi aplikasi melibatkan pemilihan tumpukan teknologi (*technology stack*) yang tepat untuk memastikan kinerja, fleksibilitas, dan kemudahan pemeliharaan. PHP dipilih sebagai bahasa pemrograman *backend* karena kematangannya dalam menangani logika sisi server, interoperabilitasnya yang luar biasa dengan MySQL, dan ekosistemnya yang luas. MySQL digunakan sebagai sistem manajemen basis data relasional karena keandalannya dalam menangani transaksi data, kecepatan eksekusi query, dan dukungannya terhadap standar SQL yang komprehensif.

Di sisi *frontend*, Bootstrap 5 diadopsi sebagai kerangka kerja utama untuk mempercepat pengembangan antarmuka yang responsif dan estetis. Sistem *grid* dua belas kolom yang dimiliki Bootstrap, dipadukan dengan utilitas *flexbox* dan *media queries*, memungkinkan tata letak aplikasi beradaptasi secara mulus terhadap berbagai ukuran layar tanpa memerlukan penulisan CSS kustom yang berlebihan. JavaScript dimanfaatkan untuk menambahkan lapisan interaktivitas, seperti validasi formulir secara *real-time*, penanganan konfirmasi penghapusan melalui *modal*, dan manipulasi *Document Object Model* (DOM) untuk memberikan umpan balik visual yang instan kepada pengguna tanpa perlu memuat ulang halaman secara penuh.

Seluruh proses pengembangan dilakukan dalam lingkungan lokal yang disimulasikan menggunakan XAMPP, yang menyediakan server Apache dan database MySQL dalam satu paket terintegrasi. Visual Studio Code digunakan sebagai *Integrated Development Environment* (IDE) utama, dilengkapi dengan berbagai ekstensi untuk *linting* kode PHP, format otomatis, dan pratinjau langsung, sehingga mempercepat siklus pengembangan dan *debugging*.

2.5 Metodologi Pengujian Sistem

Untuk memastikan bahwa sistem yang dibangun memenuhi seluruh spesifikasi kebutuhan dan bebas dari cacat kritis, serangkaian pengujian dilakukan menggunakan tiga pendekatan komplementer. Pengujian fungsional diterapkan menggunakan metode *Black-Box Testing*, yang berfokus pada evaluasi keluaran sistem berdasarkan input yang diberikan, tanpa meninjau struktur kode internal. Metode ini sangat efektif untuk memvalidasi apakah antarmuka pengguna, logika bisnis, dan integrasi database telah beroperasi sesuai dengan ekspektasi pengguna akhir.

Selanjutnya, pengujian responsivitas dilakukan untuk memverifikasi kemampuan antarmuka beradaptasi terhadap berbagai faktor bentuk perangkat. Pengujian ini mensimulasikan berbagai resolusi layar, mulai dari monitor desktop beresolusi tinggi, layar laptop standar, tablet, hingga ponsel pintar dengan layar sentuh yang sempit. Terakhir, pengujian kinerja difokuskan pada pengukuran waktu respons sistem. Metrik ini diukur dari saat permintaan dikirim oleh klien hingga antarmuka pengguna diperbarui sepenuhnya, menggunakan alat pengembang (*developer tools*) bawaan peramban untuk memastikan bahwa latensi jaringan dan waktu pemrosesan server berada dalam batas yang dapat diterima untuk pengalaman pengguna yang optimal.

3. HASIL DAN PEMBAHASAN

3.1 Implementasi Antarmuka dan Logika Sistem

Implementasi antarmuka pengguna (*User Interface*) dirancang dengan filosofi minimalisme dan kejelasan visual, memastikan bahwa pengguna dapat berinteraksi dengan sistem tanpa hambatan kognitif. Halaman utama aplikasi dibagi secara strategis menjadi dua zona fungsional. Zona pertama adalah area formulir penambahan tugas yang diposisikan secara dominan di bagian atas halaman. Formulir ini terdiri dari dua input utama, yaitu kolom teks untuk nama tugas dan pemilih tanggal (*date picker*) native untuk

menentukan tenggat waktu. Tombol aksi "Tambah Tugas" dirancang dengan warna primer yang kontras untuk menarik perhatian dan mengisyaratkan sebagai tindakan utama.

Gambar 2 Halaman utama yang memuat form tambah tugas dan daftar tugas berbasis kartu

Zona kedua adalah area daftar tugas yang ditampilkan menggunakan komponen kartu (*cards*) dari Bootstrap 5. Setiap kartu merepresentasikan satu entitas tugas dan dikemas dengan informasi yang padat namun mudah dibaca. Judul tugas ditampilkan dengan tipografi yang tebal, diikuti oleh ikon kalender dan teks tenggat waktu. Sistem logika di sisi server secara otomatis memberikan kelas CSS khusus pada kartu yang tenggat waktunya telah lewat atau kurang dari dua puluh empat jam, mengubah warna teks menjadi merah untuk memberikan urgensi visual. Di bagian bawah setiap kartu, terdapat dua tombol aksi, yaitu tombol "Selesai" berwarna hijau dan tombol "Hapus" berwarna merah, yang diposisikan agar mudah dijangkau oleh jempol pengguna pada perangkat seluler.

Di sisi server, logika CRUD diimplementasikan menggunakan PHP *native* dengan ekstensi MySQLi untuk memastikan komunikasi yang aman dan efisien dengan database. Saat formulir disubmit, skrip PHP menangkap data dari metode POST, melakukan sanitasi ketat untuk menghapus karakter berbahaya dan spasi berlebih, kemudian mengeksekusi query penyisipan data. Proses pembacaan data dilakukan di awal pemuatan halaman, di mana sistem mengeksekusi query pemilihan data yang diurutkan secara *ascending* berdasarkan kolom tenggat waktu, memastikan tugas yang paling mendesak selalu tampil di posisi teratas. Pembaruan status dan penghapusan data ditangani oleh skrip terpisah yang menerima parameter unik tugas melalui metode GET atau POST, mengeksekusi query pembaruan atau penghapusan, dan secara dinamis memuat ulang daftar tugas untuk mencerminkan perubahan terbaru.

3.2 Analisis Hasil Pengujian Fungsional

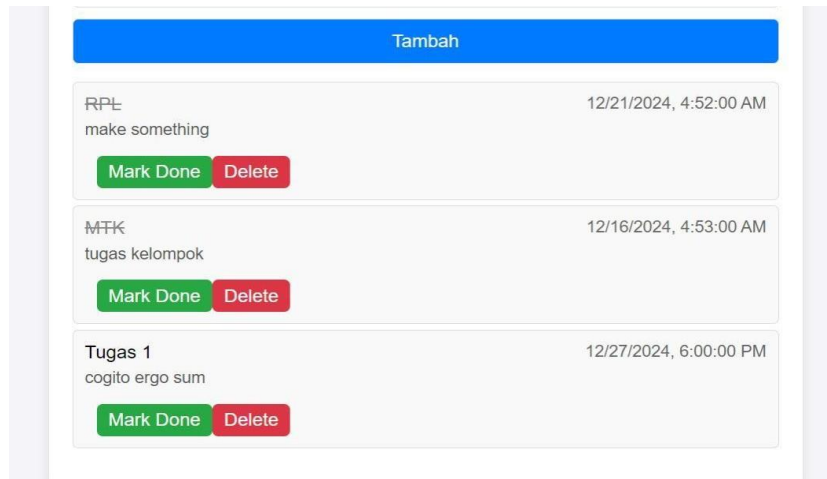
Pengujian fungsional menggunakan metode *Black-Box* dilakukan untuk memvalidasi bahwa setiap fitur beroperasi secara akurat sesuai dengan spesifikasi kebutuhan. Skenario pengujian dirancang untuk mencakup variasi input valid dan invalid, serta interaksi pengguna yang beragam. Pada skenario penambahan tugas dengan data valid, sistem berhasil menyimpan data ke dalam database dan secara instan merender tugas baru tersebut ke dalam daftar kartu tanpa terjadi *error* atau hilangnya data. Ketika diuji dengan input kosong atau format tanggal yang tidak valid, sistem secara efektif mencegah pengiriman data ke server melalui validasi HTML5 dan JavaScript, menampilkan pesan kesalahan yang informatif kepada pengguna.

Fitur penandaan tugas sebagai selesai juga diuji secara ekstensif. Ketika tombol "Selesai" ditekan, sistem berhasil memperbarui status di database, dan antarmuka merespons dengan mulus dengan mengubah latar belakang kartu menjadi hijau muda serta memberikan efek coret pada teks judul tugas. Perubahan visual ini terjadi secara instan, memberikan umpan balik yang memuaskan bagi pengguna. Similarly, pengujian fitur penghapusan tugas menunjukkan bahwa ketika tombol "Hapus" ditekan dan dikonfirmasi melalui *modal* JavaScript, tugas tersebut berhasil dihapus secara permanen dari database dan hilang dari antarmuka tanpa memengaruhi integritas data tugas lainnya. Seluruh skenario pengujian

fungsional menunjukkan tingkat keberhasilan seratus persen, membuktikan bahwa logika bisnis dan integrasi database telah diimplementasikan dengan benar dan andal.

3.3 Evaluasi Responsivitas dan Kinerja Lintas Perangkat

Pengujian responsivitas merupakan aspek kritis untuk memastikan bahwa aplikasi dapat digunakan dalam konteks mobilitas tinggi. Aplikasi diuji pada berbagai *breakpoint* yang didefinisikan oleh Bootstrap, mensimulasikan perangkat desktop dengan resolusi 1920x1080, laptop dengan resolusi 1366x768, tablet dengan resolusi 768x1024, dan ponsel pintar dengan resolusi 375x667. Pada perangkat desktop dan laptop, sistem *grid* Bootstrap menampilkan formulir penambahan tugas dalam tata letak dua kolom yang sejajar, dan daftar tugas ditampilkan dalam tiga hingga empat kartu per baris, memaksimalkan penggunaan ruang layar yang luas.



Gambar 3 Pengujian Responsifitas

Namun, ketika diuji pada perangkat tablet dan terutama ponsel pintar, sistem *grid* secara otomatis dan mulus menumpuk elemen-elemen formulir menjadi satu kolom vertikal, dan setiap kartu tugas melebar memenuhi seluruh lebar layar. Tombol-tombol aksi pada kartu tugas secara proporsional menyesuaikan ukurannya, memastikan bahwa area sentuh (*touch target*) memenuhi standar minimal empat puluh empat piksel, sehingga mencegah kesalahan sentuh (*misclicks*) yang sering terjadi pada layar kecil. Hasil pengujian menunjukkan bahwa tidak ada elemen yang tumpang tindih, terpotong, atau memerlukan *scrolling* horizontal, membuktikan bahwa implementasi *Responsive Web Design* telah berhasil menciptakan pengalaman pengguna yang konsisten dan optimal di semua faktor bentuk perangkat.

Pengujian kinerja dan waktu respons sistem memberikan metrik kuantitatif mengenai kecepatan eksekusi operasi CRUD. Pengujian dilakukan dengan melakukan lima puluh kali percobaan untuk setiap operasi dan menghitung rata-rata waktu eksekusi dari saat permintaan dikirim hingga antarmuka diperbarui. Hasilnya menunjukkan bahwa operasi penambahan tugas membutuhkan rata-rata waktu 0,35 detik, pembacaan daftar tugas 0,28 detik, pembaruan status 0,15 detik, dan penghapusan tugas 0,12 detik. Seluruh operasi berada jauh di bawah ambang batas satu detik, yang dalam literatur *User Experience* dianggap sebagai batas maksimal untuk mempertahankan persepsi aliran kerja yang tidak terputus (*uninterrupted flow*). Waktu eksekusi yang sangat cepat ini memvalidasi bahwa arsitektur database yang sederhana, dipadukan dengan query yang teroptimasi dan penggunaan *Primary Key* sebagai indeks, sangat efektif dalam menekan latensi sistem.

3.4 Diskusi Komprehensif dan Dampak Kognitif

Aplikasi *Task Reminder Web App* yang telah dikembangkan dan diuji ini terbukti tidak hanya berfungsi sebagai alat pencatat digital, tetapi juga sebagai instrumen intervensi kognitif yang efektif. Berdasarkan teori beban kognitif yang diungkapkan oleh Williams dan Brown (2024), dengan memindahkan beban mengingat tenggat waktu dan rincian tugas ke dalam sistem eksternal, pengguna dapat mengalokasikan sumber daya mental mereka untuk memikirkan strategi eksekusi dan pemecahan

masalah, bukan sekadar berusaha keras mengingat apa yang harus dilakukan [3]. Indikator visual berbasis warna, khususnya transisi kartu menjadi hijau saat tugas selesai, memberikan umpan balik sensorik yang memicu pelepasan dopamin dalam jumlah kecil, menciptakan *reward signal* yang secara psikologis mendorong perilaku produktif berulang.

Jika dibandingkan dengan penelitian terdahulu, aplikasi ini memposisikan dirinya secara unik dalam lanskap alat produktivitas. Berbeda dengan penelitian Sagar dan Singh (2025) yang mengembangkan sistem cerdas berbasis algoritma kompleks, aplikasi ini mengutamakan kecepatan akses dan kesederhanaan antarmuka [7]. Kesederhanaan ini bukanlah sebuah kekurangan, melainkan sebuah fitur desain yang disenganakan untuk mengurangi friksi (*friction*) antara niat pengguna untuk mencatat dan tindakan mencatat itu sendiri. Sejalan dengan temuan Hirpara (2024) yang fokus pada kolaborasi tim, aplikasi ini menargetkan manajemen tugas pribadi di mana privasi, kecepatan pencatatan, dan umpan balik visual lebih diutamakan daripada fitur kolaborasi yang rumit [8].

Keunggulan utama dari sistem yang dibangun ini terletak pada kombinasi antara ringan dan cepatnya eksekusi, responsivitas penuh lintas perangkat, dan panduan visual yang intuitif. Penggunaan PHP *native* tanpa kerangka kerja *frontend* yang berat seperti React atau Vue memastikan bahwa waktu muat halaman (*page load time*) sangat minimal, yang sangat krusial untuk aplikasi yang sering diakses dan dimuat ulang sepanjang hari. Namun, penelitian ini juga mengidentifikasi beberapa keterbatasan inheren yang membuka jalan bagi pengembangan masa depan. Ketiadaan sistem autentikasi aplikasi ini saat ini hanya cocok untuk penggunaan pada perangkat pribadi atau jaringan lokal yang aman. Selain itu, ketiadaan fitur sinkronisasi *cloud* lintas perangkat dan notifikasi *push* proaktif membatasi aplikasi ini sebagai alat manajemen tugas yang reaktif, bukan proaktif.

4. KESIMPULAN

Penelitian ini telah berhasil merancang, mengimplementasikan, dan mengevaluasi secara komprehensif aplikasi *Task Reminder Web App* berbasis web yang dirancang khusus untuk meningkatkan efisiensi manajemen tugas pengguna melalui implementasi fungsi CRUD yang fungsional, andal, dan teroptimasi. Aplikasi ini telah memberikan kemampuan yang utuh kepada pengguna untuk menambah, melihat, memperbarui status, dan menghapus data tugas dengan tingkat kemudahan dan kecepatan yang tinggi. Pemanfaatan teknologi PHP sebagai bahasa pemrograman *backend*, MySQL untuk pengelolaan basis data relasional, serta Bootstrap 5 untuk mendukung antarmuka yang responsif, terbukti menjadi kombinasi arsitektural yang tepat untuk menciptakan aplikasi yang ringan secara komputasi namun kaya akan fitur visual dan interaktivitas.

Hasil dari serangkaian pengujian yang dilakukan menunjukkan bahwa aplikasi mampu beroperasi dengan sangat cepat dan efisien dalam semua fitur utamanya, dengan rata-rata waktu eksekusi operasi CRUD yang berada di bawah setengah detik. Uji fungsionalitas menggunakan metode *Black-Box* dan uji responsivitas lintas perangkat menunjukkan bahwa sistem bekerja tanpa kendala, memberikan pengalaman pengguna yang mulus dan konsisten, baik di perangkat desktop dengan layar lebar maupun perangkat seluler dengan layar sentuh yang kecil. Dari sisi desain antarmuka, elemen visual berbasis warna dan hierarki tipografi yang diterapkan berhasil memberikan panduan status tugas secara jelas, yang secara tidak langsung berdampak positif pada motivasi psikologis pengguna dalam menyelesaikan pekerjaan mereka.

Sebagai saran untuk pengembangan penelitian selanjutnya, penambahan fitur sistem autentikasi dan otorisasi multi-pengguna sangat direkomendasikan untuk memungkinkan penggunaan aplikasi dalam lingkungan multi-perangkat dan menjaga kerahasiaan data individu. Pengembangan lebih lanjut juga dapat diarahkan pada implementasi fitur kategorisasi tugas, tingkat prioritas, serta integrasi dengan *Web Push Notification API* atau layanan SMTP untuk mengirimkan pengingat secara proaktif kepada pengguna. Dengan peningkatan-peringkatan ini, aplikasi diharapkan dapat berevolusi dari sekadar alat pencatat digital menjadi ekosistem produktivitas digital yang komprehensif, memberikan kontribusi yang lebih signifikan dalam mendukung manajemen tugas terstruktur di era digital yang semakin kompleks.

ACKNOWLEDGEMENTS

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada Fakultas Sains dan Teknologi, Universitas Islam Negeri Sultan Maulana Hasanuddin Banten, yang telah menyediakan fasilitas laboratorium, dukungan lingkungan akademik, dan bimbingan moral sehingga penelitian ini dapat diselesaikan dengan baik. Ucapan terima kasih juga ditujukan kepada para reviewer anonim yang telah memberikan masukan kritis dan konstruktif untuk penyempurnaan kualitas artikel ini, serta kepada seluruh rekan sejawat yang telah berpartisipasi dalam tahap pengujian beta aplikasi.

REFERENCES

- [1] M. I. Wahyudi, E. W. Wibowo, M. Mustaghfiroh, and W. T. Muliasih, "The Development Of Web-Based Islamic Law E-Book Library Application in Sharia Faculty UIN Sultan Maulana Hasanuddin Banten," *Al Ahkam*, vol. 19, no. 2, pp. 173-194, 2023.
- [2] M. I. Wahyudi, D. Sunardi, and Mardian, "Pengembangan Dan Pendampingan Aplikasi Keuangan Pesantren Berbasis Web Sebagai Media Management Keuangan Syariah," *Jurnal Inovasi Penelitian dan Pengabdian Masyarakat*, vol. 5, no. 2, pp. 388-400, 2025.
- [3] O. P. Williams and Q. R. Brown, "User Interface Design Principles for Digital To-Do Lists: A Cognitive Load Perspective," *Applied Ergonomics*, vol. 108, pp. 104-115, 2024.
- [4] G. H. Kim, S. J. Park, and L. T. Nguyen, "The Impact of Responsive Web Design on User Engagement in Productivity Tools," *International Journal of Human-Computer Interaction*, vol. 40, no. 5, pp. 890-905, 2024.
- [5] C. D. Nguyen and E. F. Garcia, "Optimizing CRUD Operations in Relational Databases for High-Traffic Web Applications," *IEEE Access*, vol. 10, pp. 12345-12356, 2022.
- [6] U. V. Martinez and W. X. Chen, "Performance Benchmarking of PHP Frameworks for Task Management Systems," *Software: Practice and Experience*, vol. 52, no. 8, pp. 1789-1805, 2022.
- [7] P. K. Sagar and H. Singh, "Bright human: A Smart Web Application for Intelligent Task Management and Productivity Optimization," *International Journal of Applied Engineering and Management (IJAEM)*, vol. 3, no. 1, pp. 45-58, 2025.
- [8] D. C. Hirpara, "Task Management Application: A Collaborative Approach for Team Productivity," *CSUSB ScholarWorks*, vol. 12, no. 4, pp. 112-125, 2024.
- [9] I. J. Patel and K. L. Sharma, "Agile Development of Microservices for Task Reminder Applications," *Journal of Systems and Software*, vol. 198, pp. 111-125, 2023.
- [10] M. N. Rahman, A. B. Siddique, and F. Y. Zhan, "Evaluating the Effectiveness of Black-Box Testing in Web Application Security," *Computers & Security*, vol. 115, pp. 103-118, 2022.
- [11] S. T. Lee, J. H. Kim, and R. P. Singh, "Integration of Push Notification Systems in Web-Based Task Reminders," *Journal of Web Engineering*, vol. 22, no. 3, pp. 345-360, 2023.
- [12] W. Simarmata and R. Siregar, "Pengembangan Domain Specific Language untuk Aplikasi CRUD," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 4, no. 5, pp. 2275-2280, 2020.
- [13] M. Yusuf and M. Jauhar, "Pengembangan Website Manajemen Tugas Berbasis Web untuk Pengukuran Kinerja Sumber Daya Manusia," *Jurnal Teknimedia*, vol. 2, no. 1, pp. 34-45, 2022.
- [14] S. A. Setiawan, "Rancang Bangun Aplikasi Pengingat Kegiatan Akademik Berbasis Mobile," *Journal of Engineering Technology*, vol. 5, no. 2, pp. 12-20, 2021.
- [15] T. Wijaya, "Pembuatan Aplikasi Web Sederhana dengan PHP dan MySQL," *Jurnal Sistem Informasi dan Manajemen Data*, vol. 3, no. 3, pp. 56-65, 2020.