



Prototyping Aplikasi Penjadwalan Konsultasi Dokter Menggunakan Framework Model View Controller

Putra Agus Saripudin¹, Azifa Habiba¹

¹Program Studi Teknik Informatika, Universitas Esa Unggul, Bekasi, Indonesia

Artikel Info

Kata Kunci:

Janji Temu Dokter;
Klinik dr. Yudi Dimyati;
Laravel;
Model View Controller;
Prototyping.

Keywords:

Doctor Appointment;
Clinic dr. Yudi Dimyati;
Laravel;
Model View Controller;
Prototyping.

Riwayat Artikel:

Submitted: 04 September 2025

Accepted: 25 November 2025

Published: 28 November 2025

Abstrak: Kemajuan teknologi informasi mendorong digitalisasi layanan kesehatan, namun banyak fasilitas kesehatan primer masih mengandalkan prosedur manual yang berpotensi menyebabkan inefisiensi, waktu tunggu yang panjang, serta ketidakteraturan jadwal praktik dokter. Penelitian ini bertujuan merancang dan mengembangkan *prototype* aplikasi penjadwalan konsultasi dokter berbasis web menggunakan *framework Laravel* dan arsitektur *Model View Controller* (MVC) untuk meningkatkan efisiensi pelayanan di Klinik dr. Yudi Dimyati. Metode pengembangan menggunakan pendekatan *prototyping* melalui tahapan pengumpulan kebutuhan, perancangan antarmuka, evaluasi iteratif, dan pengembangan sistem. Pengujian dilakukan menggunakan *black-box testing* untuk memverifikasi kesesuaian fungsi. Hasil menunjukkan bahwa pada sisi pengguna, seluruh 15 skenario pengujian berhasil dijalankan (100%), sedangkan pada sisi administrator 29 dari 30 skenario berhasil (96,67%), sehingga tingkat keberhasilan keseluruhan mencapai 97,8%. Temuan tersebut menunjukkan bahwa *prototype* telah memenuhi spesifikasi fungsional dan mampu mendukung alur pendaftaran, pemesanan janji, pengelolaan jadwal, dan pembaruan status pembayaran secara otomatis. Secara praktis, sistem ini berpotensi meningkatkan efisiensi operasional klinik, sementara secara teoretis penelitian ini memperkuat bukti empiris mengenai efektivitas arsitektur MVC dan pendekatan *prototyping* dalam pengembangan sistem informasi kesehatan.

Abstract: Advances in information technology have accelerated the digital transformation of healthcare services; however, many primary healthcare facilities continue to rely on manual procedures that generate inefficiencies, prolonged waiting times, and inconsistencies in physicians' scheduling. This study aims to design and develop a web-based doctor appointment scheduling prototype using the Laravel framework and the Model View Controller (MVC) architecture to enhance service efficiency at Klinik dr. Yudi Dimyati. The system was developed using a prototyping approach involving requirements elicitation, iterative interface design, user-driven refinement, and system implementation. Functional verification was conducted through black-box testing. The results show that all 15 user-side test scenarios were successfully completed (100%), while 29 of 30 administrator-side scenarios were executed correctly (96.67%), yielding an overall success rate of 97.8%. These findings indicate that the prototype meets the defined functional specifications and effectively supports key processes, including patient registration, appointment booking, schedule management, and automated payment status updates. Practically, the system demonstrates potential for improving operational efficiency in clinical settings, while theoretically, the study contributes empirical evidence supporting the effectiveness of the MVC architecture and the prototyping method in developing healthcare information systems.

Corresponding Author:

Putra Agus Saripudin

Email: putraagus9888@gmail.com

PENDAHULUAN

Kemajuan teknologi informasi telah membawa perubahan besar dalam berbagai aspek kehidupan, termasuk layanan kesehatan yang beralih dari proses manual menuju alur digital yang lebih terstruktur, efisien, dan terukur (Heryawan et al., 2025; Ramadani et al., 2023; Cha, 2023). Digitalisasi sistem informasi kesehatan menjadi kebutuhan mendesak untuk meningkatkan efisiensi sekaligus kualitas layanan. Pemanfaatan teknologi memungkinkan pasien melakukan reservasi secara daring dan memperoleh jadwal konsultasi secara lebih praktis, sehingga efektivitas waktu pelayanan dapat dicapai (Putri et al., 2025). Salah satu bentuk implementasi yang krusial adalah sistem penjadwalan janji temu dokter berbasis *website*, yang mampu mengotomasi proses administrasi, memperbaiki alur pasien, serta memfasilitasi monitoring kinerja layanan (Ramdoni & Herdiansyah, 2023). Transformasi ini hanya dapat diwujudkan apabila proses rekayasa perangkat lunak mengikuti siklus pengembangan sistem kesehatan melalui tahapan *prototyping*, validasi pengguna, dan integrasi antarsistem agar aplikasi dapat diadopsi secara optimal (Betancor et al., 2025).

Salah satu komponen penting dalam fasilitas pelayanan kesehatan adalah jadwal praktik dokter. Tanpa jadwal praktik yang terorganisir, pasien akan kesulitan mengetahui jadwal dokter yang tersedia, terutama bagi pasien baru. Jadwal praktik dokter meliputi informasi jam kerja dokter, waktu pemeriksaan pasien, kapasitas pasien yang dapat dilayani, serta informasi mengenai cuti dokter. Jadwal praktik yang tersusun dengan baik tidak hanya membantu mengatur layanan medis agar lebih efisien, tetapi juga memungkinkan pasien untuk membantu janji temu sesuai dengan waktu yang tersedia (Hendiawati dkk., 2023). Dengan demikian, dapat dihindari penumpukan pasien atau keterlambatan pelayanan di rumah sakit. Hal ini juga memudahkan pasien dalam memilih dokter sesuai kebutuhan medis mereka sekaligus mengelola waktu secara lebih efektif.

Beberapa penelitian menemukan bahwa rata-rata waktu tunggu di unit rawat jalan kerap melampaui standar layanan nasional yang menetapkan batas maksimal 60 menit. Kondisi tersebut berimplikasi pada menurunnya kepuasan pasien sekaligus berisiko menghambat akses terhadap layanan esensial (Santalia et al., 2023; Azmi et al., 2024). Realitas operasional ini memperkuat urgensi penerapan intervensi digital yang mampu menyederhanakan proses pendaftaran, menata aliran pasien, dan meningkatkan transparansi jadwal praktik dokter. Hasil pra-survei yang dilakukan di Klinik dr. Yudi Dimyati mengonfirmasi masalah serupa, di mana proses pendaftaran pasien masih dilakukan secara manual menggunakan kertas, informasi jadwal dokter hanya dapat diperoleh dengan datang langsung ke klinik, serta pencatatan data pasien belum terintegrasi dengan sistem digital. Kondisi ini menyebabkan pelayanan berjalan kurang efisien dan menyulitkan pasien dalam merencanakan konsultasi medis.

Berbagai studi sebelumnya menunjukkan bahwa sistem penjadwalan janji temu daring *dengan fitur self-scheduling* dapat meningkatkan efisiensi operasional dan menurunkan angka ketidakhadiran pasien. Penjadwalan digital yang dilengkapi pengingat otomatis dan alur konfirmasi mampu mengurangi beban administratif tenaga kesehatan, memperpendek waktu antara pendaftaran dan konsultasi, serta meningkatkan kepuasan pasien (Habibi et al., 2024; Greenup & Best, 2025). Namun, sebagian besar studi tersebut berfokus pada aspek fungsional tertentu, seperti efektivitas pemesanan daring atau pengurangan beban administratif tanpa mengkaji secara mendalam pengembangan sistem berbasis *prototyping* yang terintegrasi dengan alur operasional klinik. Dengan demikian, terdapat kebutuhan untuk mengembangkan *prototype* aplikasi penjadwalan janji temu yang tidak hanya mendukung pemesanan, tetapi juga menyesuaikan struktur arsitektural dengan kebutuhan operasional fasilitas kesehatan.

Dari perspektif rekayasa perangkat lunak, pemilihan pola arsitektur yang memisahkan tanggung jawab seperti *Model View Controller* (MVC) memberikan keuntungan dalam hal pemeliharaan,

pengujian, dan kemudahan pengembangan berulang. Pendekatan *prototyping* yang dikombinasikan dengan *framework* modern seperti *Laravel* memungkinkan percepatan siklus pengembangan, memudahkan pengujian antarmuka, serta mendukung pengorganisasian kode agar lebih terstruktur. Literatur terbaru menunjukkan bahwa penerapan arsitektur MVC pada aplikasi kesehatan membantu memastikan fleksibilitas sistem ketika dilakukan proses validasi oleh pengguna (Paolone et al., 2021; Faisal & Nakayama, 2024). Namun, studi-studi tersebut umumnya tidak mengkaji keterpaduan penerapan MVC dengan pengembangan *prototype* untuk sistem penjadwalan dokter yang terhubung dengan kebutuhan operasional klinik primer di Indonesia. Selain itu, integrasi dengan *payment gateway* Midtrans melalui API memungkinkan proses pembayaran berlangsung secara otomatis, serta memperbarui status janji temu secara *real-time* tanpa intervensi manual dari staf administrasi.

Berdasarkan kajian literatur, terdapat *research gap* yang mendasari pentingnya penelitian ini. Studi terdahulu umumnya hanya mengembangkan sebagian komponen dari sistem penjadwalan, seperti fitur pemesanan daring, pengingat otomatis, atau pengelolaan antrian, tanpa menghubungkan seluruh proses layanan dalam satu alur operasional yang utuh. Selain itu, proses pemesanan janji temu pada penelitian sebelumnya belum terhubung secara langsung dengan mekanisme pengelolaan jadwal dokter maupun sistem pembayaran digital yang mampu memperbarui status janji temu secara otomatis. Pendekatan *prototyping* yang memungkinkan validasi iteratif berbasis umpan balik pengguna juga jarang diterapkan, sehingga rancangan sistem yang dihasilkan belum sepenuhnya mencerminkan kebutuhan operasional fasilitas kesehatan. Dengan demikian, diperlukan penelitian yang mengembangkan *prototype* aplikasi penjadwalan yang memadukan alur pemesanan, manajemen jadwal dokter, automasi pembayaran, serta validasi langsung pada lingkungan klinik primer.

Sejalan dengan uraian tersebut, penelitian ini merumuskan pertanyaan penelitian sebagai berikut: "Bagaimana cara merancang dan membangun *prototype* aplikasi penjadwalan konsultasi dokter berbasis *framework Laravel* dengan pola *Model View Controller* menggunakan pendekatan *prototyping* dalam konteks operasional Klinik dr. Yudi Dimyati?" Untuk menjawab pertanyaan tersebut, penelitian ini mengembangkan aplikasi berbasis web "Janji Temu Dokter" yang dirancang untuk memfasilitasi proses pemesanan janji temu, pengelolaan jadwal praktik dokter, serta pencatatan data pasien secara lebih terstruktur dan *real-time*.

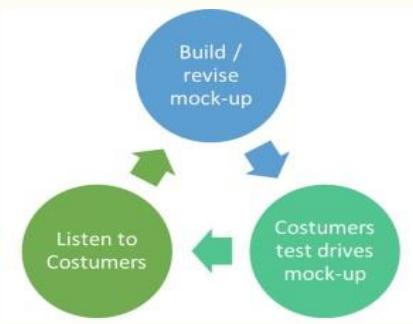
Selain itu, *prototype* ini mengimplementasikan integrasi *payment gateway* Midtrans melalui API untuk memfasilitasi pembayaran daring, di mana sistem menerima notifikasi otomatis atas status transaksi sehingga status janji temu dapat diperbarui tanpa intervensi manual. Berdasarkan tujuan penelitian yang telah ditetapkan, aplikasi ini dikembangkan untuk: (1) menyediakan sistem pencatatan antrian pasien secara *real-time* guna meminimalkan kesalahan pencatatan serta mencegah tumpang tindih jadwal; (2) memungkinkan pasien melakukan *reschedule* janji temu pada hari yang sama apabila terjadi perubahan mendadak; dan (3) menyediakan antarmuka yang memungkinkan pihak klinik mengakses dan menampilkan data pasien maupun dokter secara mudah dan efisien. Sistem yang dihasilkan selanjutnya diuji menggunakan *black-box testing* dan divalidasi oleh pengguna di lingkungan Klinik dr. Yudi Dimyati untuk memastikan kesesuaian *prototype* dengan kebutuhan operasional layanan.

METODE

Penelitian ini berfokus pada perancangan dan pengembangan *prototype* aplikasi "Janji Temu Dokter" yang diterapkan pada Klinik dr. Yudi Dimyati untuk meningkatkan efisiensi pendaftaran pasien, pengelolaan jadwal praktik dokter, serta pemantauan riwayat konsultasi secara lebih terstruktur. Proses pengembangan sistem menggunakan metodologi *Agile* dengan pendekatan *prototyping*, karena model ini memungkinkan penyesuaian berulang berdasarkan kebutuhan operasional klinik.

Tahapan *prototyping* dimulai dari *Listen to Customer* yang bertujuan untuk mengumpulkan kebutuhan melalui observasi proses pelayanan klinik dan wawancara dengan tiga sampel terdiri dari masing-masing satu orang dokter, staf administrasi dan pasien untuk memperoleh gambaran

operasional yang aktual. Tahap Kedua yaitu *Build/Revise Mock-up*. Proses pengembangan desain awal sistem dalam bentuk *prototype* sederhana. *Prototype* ini mencerminkan tampilan dan alur kerja sistem yang diusulkan. Pengguna diberikan kesempatan untuk mencoba *prototype* tersebut, kemudian memberikan masukan terkait elemen yang sesuai ataupun tidak sesuai dengan kebutuhan mereka. Tahap Ketiga yaitu *Customer Test-Drivers Mock-Up*. Tahapan evaluasi dan revisi *prototype* berdasarkan *feedback* pengguna. Jika terdapat kekurangan, *prototype* akan diperbaiki dan diuji ulang oleh pengguna. Proses ini berlangsung secara iteratif hingga pengguna merasa puas dengan hasil akhir dan *prototype* dianggap layak untuk dikembangkan menjadi sistem akhir.

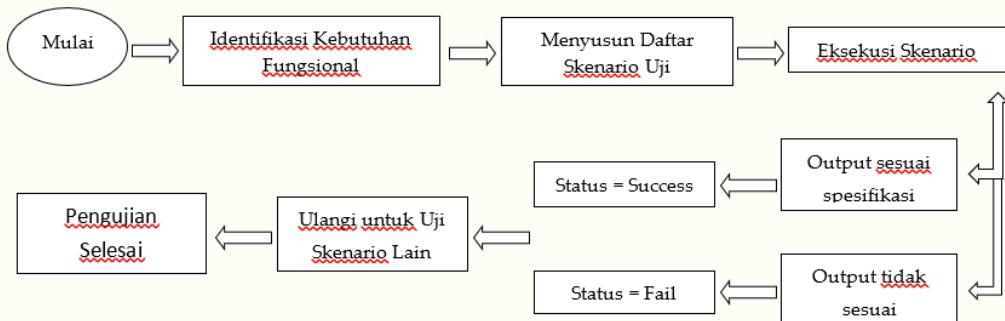


Gambar 1. Model *Prototyping*

Sumber: (Aprilianto, 2021)

Rancangan sistem menggunakan arsitektur *Model View Controller* (MVC) dengan *framework* *Laravel* untuk memastikan keteraturan struktur kode, kemudahan pemeliharaan, dan fleksibilitas pengembangan. Arsitektur sistem dibagi menjadi *model* yang menangani struktur data dan logika bisnis, *view* yang menyajikan antarmuka pengguna, serta *controller* yang mengatur alur proses pendaftaran, pemesanan janji temu, reschedule, dan pembaruan status transaksi. Alur proses kerja sistem mencakup tahapan: pasien memilih dokter dan jadwal yang tersedia, sistem menghasilkan permintaan transaksi ke *payment gateway* Midtrans, pengguna menyelesaikan pembayaran, dan sistem menerima notifikasi otomatis dari Midtrans sehingga status transaksi dan status janji temu diperbarui tanpa intervensi manual. Alur operasional ini mengacu pada diagram alir proses yang menggambarkan keseluruhan urutan aktivitas dalam sistem.

Evaluasi sistem dilakukan menggunakan metode *black-box testing* untuk menguji kesesuaian fungsi-fungsi utama dengan spesifikasi yang ditetapkan, seperti pencatatan pasien, pemesanan janji temu, perubahan jadwal, integrasi API Midtrans, serta pembaruan status transaksi secara otomatis. Pemilihan *black-box testing* didasarkan pada sifat penelitian yang masih berada pada fase pengembangan *prototype* sehingga fokus utama adalah memastikan berfungsinya komponen-komponen inti sebelum masuk ke tahap implementasi penuh. Pada tahap ini, *User Acceptance Testing* (UAT) tidak dilakukan karena *prototype* belum berada pada tingkat kesiapan implementasi sistem yang membutuhkan verifikasi formal oleh pihak manajemen. Selain itu, validasi pengguna tetap dilakukan melalui uji coba *mock-up* pada tahap *customer test-drives* untuk memastikan *prototype* sesuai dengan kebutuhan operasional.



Gambar 2. *Flowchart Black-Box Testing*

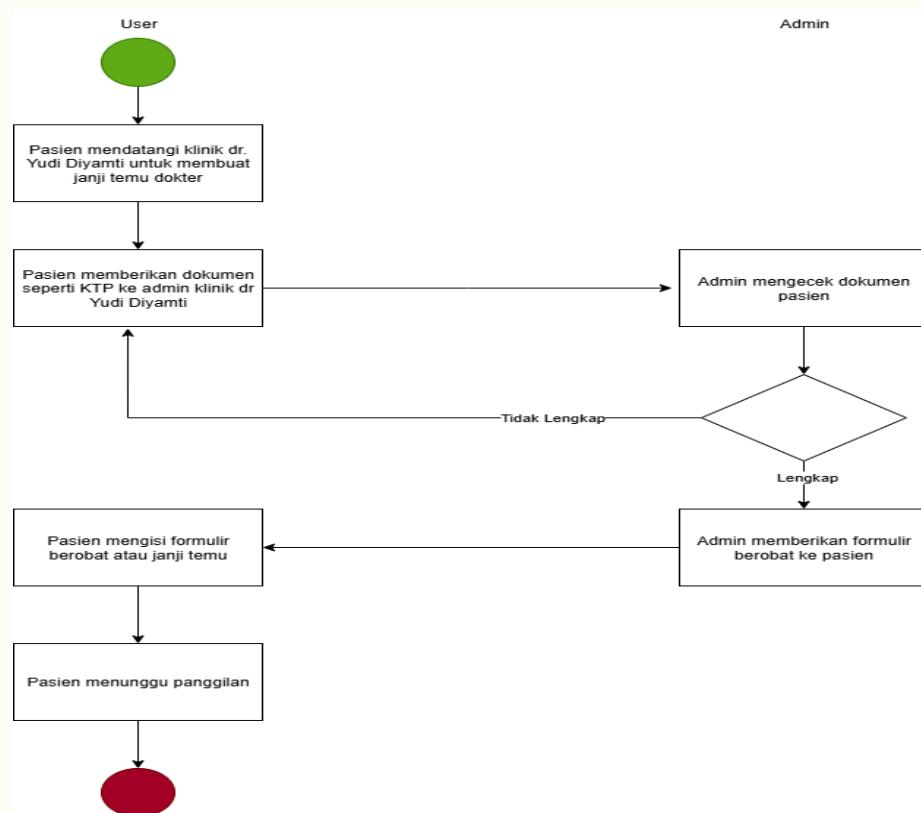
Penilaian tingkat keberhasilan dalam penelitian ini menggunakan pendekatan *task success rate* sebagaimana dikemukakan oleh Nielsen & Budiu(2001). Metrik ini menilai efektivitas sistem berdasarkan proporsi tugas yang berhasil diselesaikan dibandingkan total tugas yang diujikan. Setiap tugas diklasifikasikan ke dalam tiga kategori, yaitu *success*, *partial success*, dan *fail*, di mana *partial success* dihitung sebagai keberhasilan parsial dengan bobot 0,5. Pendekatan ini dipilih karena mampu memberikan gambaran yang lebih komprehensif mengenai efektivitas *prototype* dalam mendukung pengguna menyelesaikan tugas secara tepat dan konsisten pada tahap pengujian fungsional. Perhitungan tingkat keberhasilan dilakukan menggunakan rumus berikut:

$$\text{Success Rates} = \frac{\text{Success} + (\text{Partial Success} \times 0,5)}{\text{Total Task}} \times 100\%$$

Pada penelitian ini tidak ditemukan *partial success*, sehingga nilai *partial success* bernilai nol dan rumus tersebut setara dengan perhitungan berbasis proporsi keberhasilan penuh. Nilai *success rate* yang dihasilkan kemudian digunakan sebagai indikator kuantitatif untuk mengevaluasi stabilitas fungsi inti sistem serta kesesuaian rancangan *prototype* dengan kebutuhan operasional klinik pada tahap pengujian.

HASIL DAN PEMBAHASAN

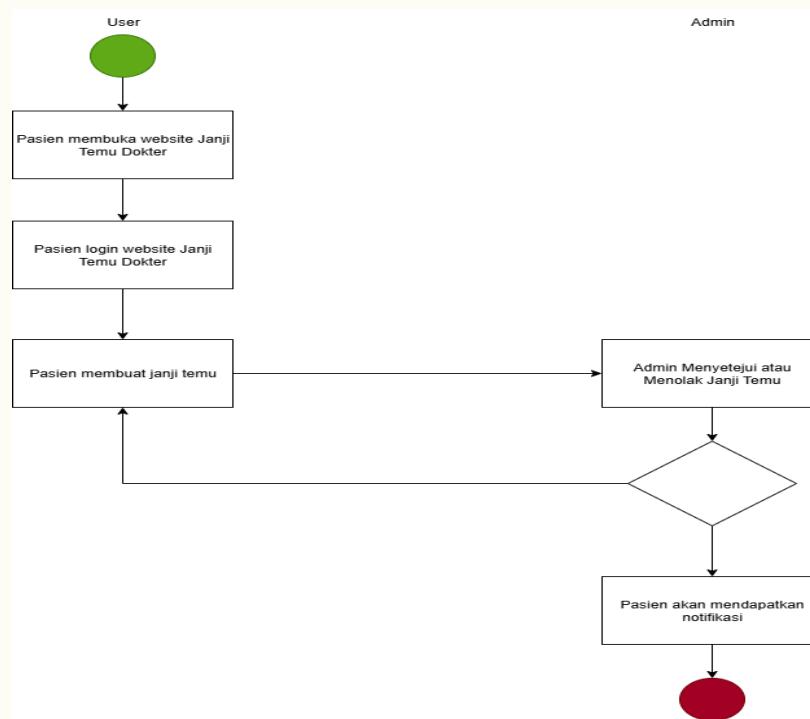
Sistem yang Sedang Berjalan



Gambar 2. Sistem Yang Sedang Berjalan

Proses bisnis yang sedang berjalan di klinik dr. Yudi Dimyati dapat kita lihat pada tabel diatas, ketika pasien ingin berobat atau melakukan janji temu dengan dokter, mereka diharuskan untuk datang langsung ke klinik. apabila klinik sedang ramai dengan pasien yang ingin berobat, maka pasien harus mengantre. Setelah mendapatkan giliran, pasien akan diminta untuk mengisi formulir pendaftaran, dan selanjutnya harus menunggu panggilan untuk menjalani pemeriksaan.

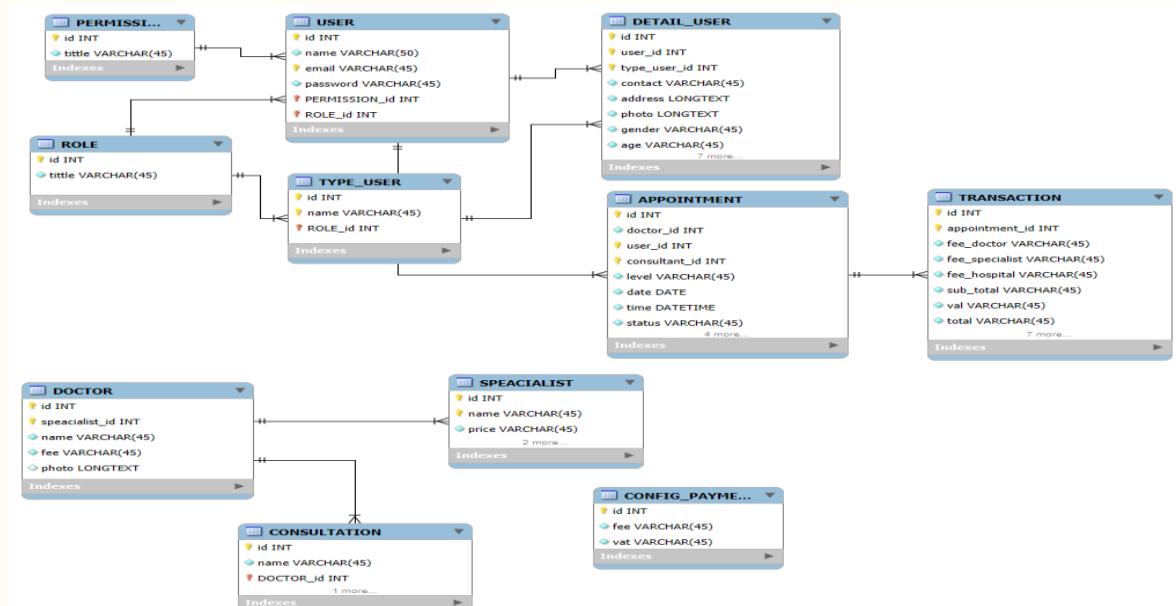
Sistem Yang Diusulkan



Gambar 3. Sistem Yang Diusulkan

Perancangan aplikasi janji temu dokter berbasis *website* ini dikembangkan sebagai platform digital untuk meningkatkan kualitas pelayanan klinik dengan sistem yang terstruktur dan terukur. Aplikasi yang dikembangkan akan terintegrasi antara *user* dan *admin*. *User* disini sebagai pasien yang akan melakukan janji temu dengan dokter. Dimana *user* harus *login* terlebih dahulu di *website* janji temu dokter, memilih dokter dari halaman list dokter, selanjutnya *user* akan diberikan formulir pendaftaran janji temu dokter. Sedangkan *admin* bertugas untuk menyetujui dan menolak pendaftaran akun *user* serta mengontrol riwayat konsultasi pasien. Melalui sistem ini diharapkan tidak ada penumpukan pada saat pasien melakukan pendaftaran.

Implementasi Basis Data

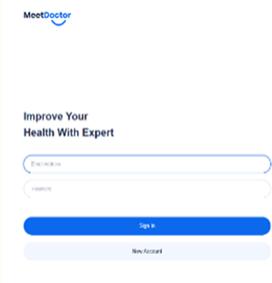


Gambar 4. ERD

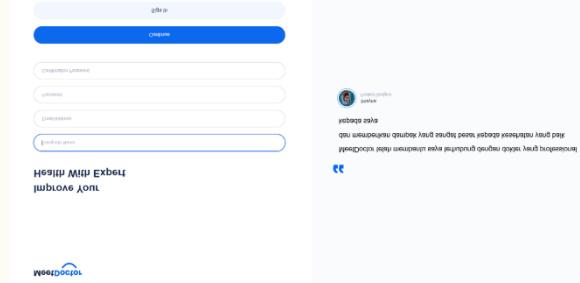
Entity Relationship Diagram (ERD) digunakan untuk merancang struktur basis data dengan menunjukkan hubungan antar entitas secara sistematis dan terorganisir. ERD mendukung efektivitas pengelolaan data layanan, jadwal, dan transaksi dalam sistem janji temu dokter. Perancangan ini memungkinkan identifikasi entitas seperti *Users*, *Roles*, *Appointment Transactions*, dan *Notifications*, beserta relasinya, sehingga sistem dapat mengelola proses pemesanan janji temu secara otomatis, meningkatkan pengalaman pengguna, serta memudahkan admin dalam pengelolaan data secara digital.

Implementasi Sistem

Bagian antarmuka menunjukkan hasil implementasi sistem berdasarkan desain yang telah dirancang, yang mencakup tampilan untuk pengguna yaitu pasien maupun administrator. Pada portal *user*, pasien dapat mengakses berbagai fitur yang disediakan sesuai kebutuhan, seperti pendaftaran, informasi layanan, maupun interaksi dengan sistem secara langsung.



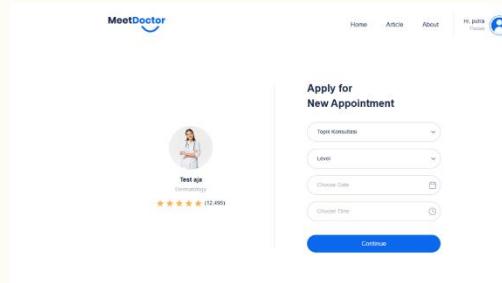
Gambar 5. Halaman *Login User*



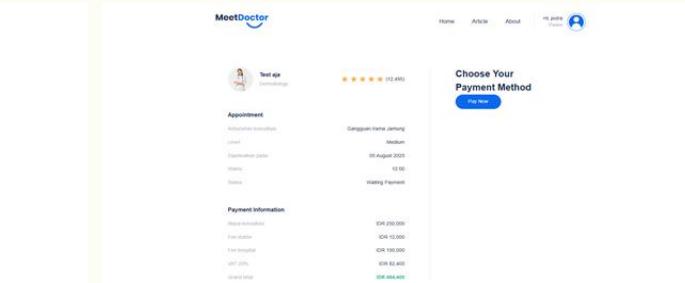
Gambar 6. Halaman *Register User*

Gambar 5. Halaman *Login User* memungkinkan pasien melakukan autentikasi menggunakan email dan password yang telah terdaftar sebelumnya. Proses *login* yang berhasil akan mengarahkan pasien ke halaman utama portal untuk melakukan pemesanan janji temu.

Gambar 6. Pada halaman *Register User*, pasien dapat melakukan pendaftaran akun baru dengan mengisi formulir yang berisi nama lengkap, email dan password. Validasi input dilakukan untuk menghindari pendaftaran dengan data yang tidak lengkap atau duplikat.



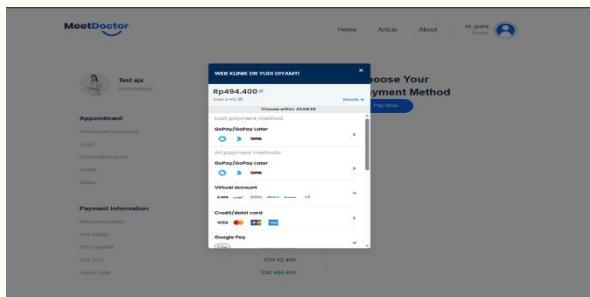
Gambar 7. Halaman Janji Temu Dokter



Gambar 8. Halaman Pembayaran *User*

Gambar 7. Halaman Pemesanan Janji Temu memungkinkan pasien untuk menjadwalkan konsultasi dengan dokter yang dipilih. Sebelum proses penjadwalan, pasien diwajibkan mengisi formulir yang memuat informasi terkait jadwal konsultasi termasuk pemilihan tanggal dan waktu.

Gambar 8. Setelah pasien sudah mengisi formulir janji temu, sistem akan mengarahkan ke halaman pembayaran. Untuk menyelesaikan proses pemesanan janji temu, pasien diwajibkan melakukan pembayaran terlebih dahulu.



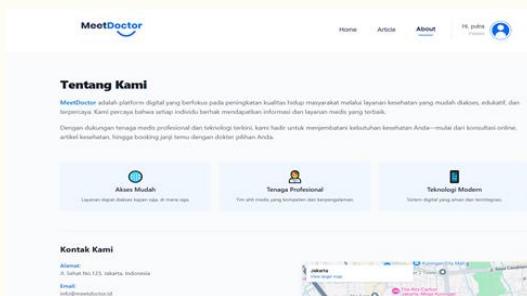
Gambar 9. Halaman Pembayaran Midtrans

Gambar 9. Halaman pembayaran terintegrasi dengan Midtrans, yang memfasilitasi pasien dalam melakukan transaksi melalui berbagai metode pembayaran yang tersedia.

Gambar 10. Halaman artikel menyajikan konten edukasi kesehatan yang dapat diakses pengguna. Artikel – artikel ini bertujuan memberikan informasi kesehatan yang relevan untuk menambah wawasan pasien dan mendukung gaya hidup sehat.



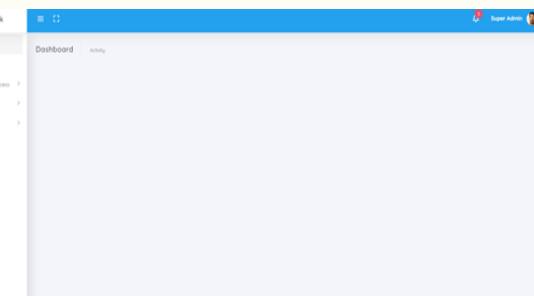
Gambar 10. Halaman Artikel



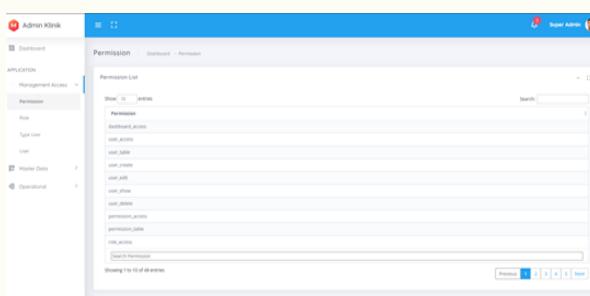
Gambar 11. Halaman Tentang Kami

Gambar 11. Halaman ini berisi informasi mengenai profil klinik, visi-misi, sejarah singkat, serta tujuan pengembangan aplikasi janji temu dokter. Halaman ini memberikan gambaran pengguna mengenai identitas dan nilai-nilai dari layanan yang disediakan.

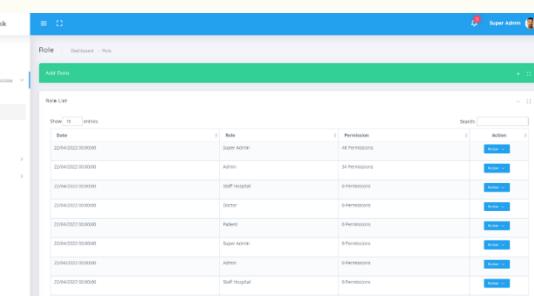
Gambar 12. Halaman *Dashboard* merupakan halaman utama untuk admin setelah berhasil *login*, menampilkan rangkuman data penting seperti jumlah pengguna, janji temu, dokter dan transaksi.



Gambar 12. Halaman *Dashboard Admin*



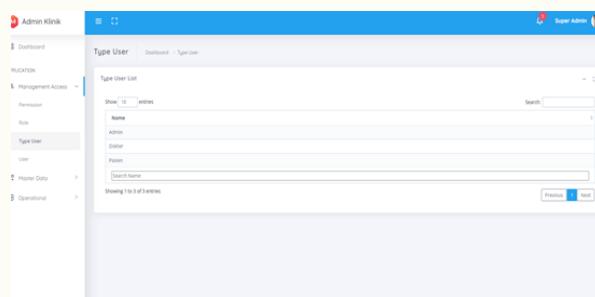
Gambar 13. Halaman *Permission Admin*



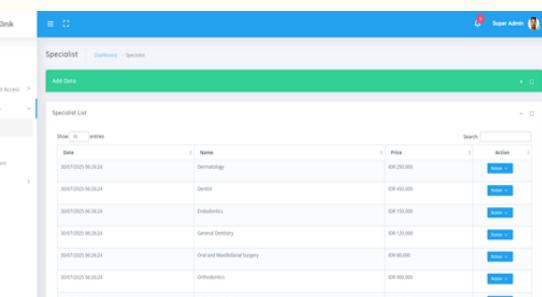
Gambar 14. Halaman *Role Admin*

Gambar 13. Halaman ini menyediakan fitur untuk mengelola hak akses dan izin pengguna dalam sistem. Admin dapat melihat seluruh daftar *permission* yang telah terdaftar.

Gambar 14. Halaman *Role* berfungsi untuk mengelola peran atau jabatan yang diberikan kepada pengguna dalam sistem, seperti pengguna biasa, dokter, dan admin. Melalui halaman ini, admin dapat menetapkan, mengubah, atau menghapus peran sesuai kebutuhan, sehingga struktur otorisasi dalam sistem dapat diatur secara terperinci. Pengelolaan peran ini bertujuan untuk memastikan bahwa setiap pengguna memiliki hak akses yang sesuai dengan tugas dan tanggung jawabnya, sekaligus menjaga keamanan serta keteraturan operasional sistem.



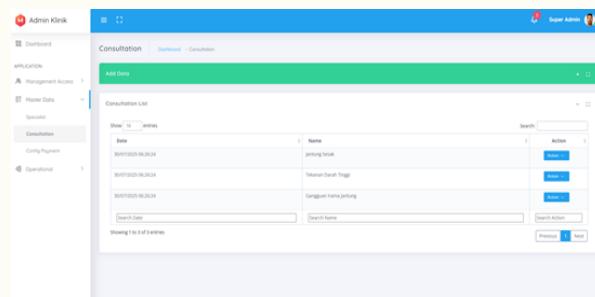
Gambar 15. Halaman Type User Admin



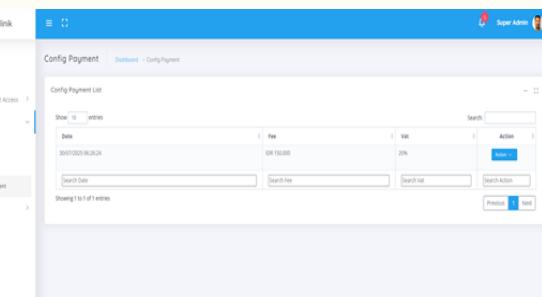
Gambar 16. Halaman Specialist Admin

Gambar 15. Halaman ini digunakan untuk mengelola tipe-tipe pengguna dalam sistem, seperti pasien, dokter, dan admin, sehingga memudahkan administrasi klasifikasi pengguna dan pengaturan hak akses yang sesuai.

Gambar 16. Halaman khusus untuk mengelola data spesialisasi dokter, termasuk penambahan, pengubahan, dan penghapusan spesialisasi. Fitur ini memudahkan pencocokan dokter dengan kebutuhan pasien berdasarkan bidang keahlian.



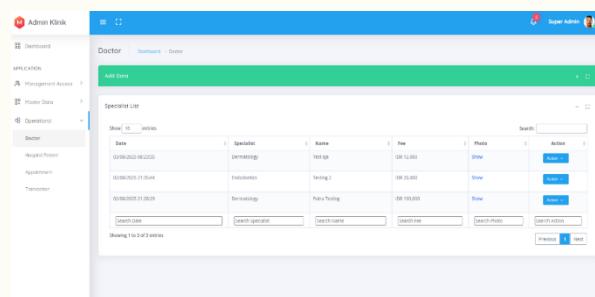
Gambar 17. Halaman Consultation Admin



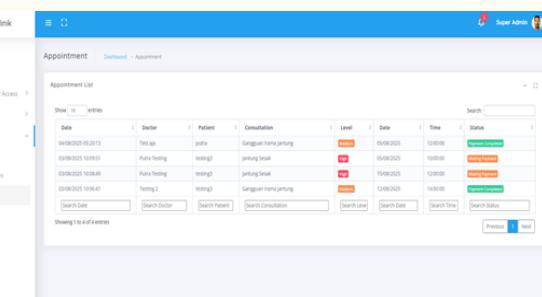
Gambar 18. Halaman Config Payment Admin

Gambar 17. Halaman untuk mengelola data konsultasi pasien, termasuk melihat riwayat konsultasi, menambah catatan medis, dan memantau status janji temu yang sedang berlangsung atau telah selesai.

Gambar 18. Admin dapat mengatur konfigurasi terkait metode dan sistem pembayaran yang digunakan dalam aplikasi, termasuk integrasi dengan layanan pembayaran *online* untuk proses transaksi yang aman dan efisien.



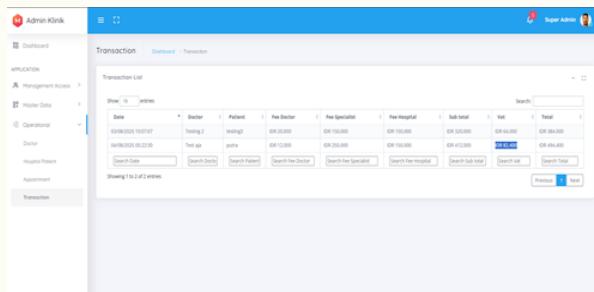
Gambar 19. Halaman Doctor Admin



Gambar 20. Halaman Appointment Admin

Gambar 19. Halaman ini menampilkan daftar seluruh dokter yang terdaftar dalam sistem, lengkap dengan informasi profil dan spesialisasinya. Melalui halaman ini, admin memiliki wewenang untuk melakukan pengelolaan data dokter, yang mencakup penambahan data dokter baru, pembaruan informasi dokter yang sudah ada, serta penghapusan data dokter yang tidak lagi aktif. Fitur ini dirancang untuk memudahkan pengelolaan sumber daya manusia di klinik, memastikan bahwa informasi yang tersedia selalu akurat dan terkini, sehingga mendukung kelancaran operasional layanan kesehatan.

Gambar 20. Halaman *Appointment* pada portal admin menampilkan daftar janji temu pasien dengan dokter, mencakup informasi jadwal, jenis konsultasi, tingkat urgensi, dan status pembayaran, serta dilengkapi fitur pencarian untuk memudahkan pengelolaan data janji temu secara efisien.



Gambar 21. Halaman Transaction Admin

Halaman ini menampilkan riwayat transaksi pembayaran yang dilakukan oleh pasien melalui aplikasi. Admin dapat memantau status pembayaran serta mengelola data transaksi untuk keperluan laporan dan audit.

Pengujian Blackbox

1. Blackbox User

Tabel 1. Blackbox User

Test Case ID	Test Case	Hasil Yang Diharapkan	Status
USR001	User bisa login ke halaman website janji temu dokter	User bisa login dan redirect ke homepage	Passed
USR002	Catatan: user harus menggunakan email dan password yang benar User tidak bisa login ke halaman website janji temu dokter	User tidak bisa login dan user akan mendapatkan message error	Passed
USR003	Catatan: user menggunakan email dan password yang belum terdaftar User berhasil registrasi di website janji temu dokter	User berhasil registrasi dan redirect ke homepage	Passed
USR004	User tidak berhasil registrasi di website janji temu dokter	User tidak berhasil registrasi dan akan mendapatkan message error	Passed
USR005	Precondition: user tidak melengkapi form registrasi User bisa akses halaman janji temu dokter	User bisa akses halaman appointment	Passed
USR006	User bisa memilih jadwal janji temu dokter di halaman janji temu	User bisa memilih jadwal janji temu dokter	Passed
USR007	User bisa akses halaman pembayaran di website janji temu dokter	User bisa akses halaman payment	Passed
USR008	User tidak bisa akses halaman pembayaran	User tidak redirect ke halaman payment dan akan mendapatkan message error	Passed
	Precondition: user belum memilih jadwal janji temu		

USR009	<i>User berhasil melakukan pembayaran di website janji temu dokter</i>	<i>User berhasil melakukan pembayaran melalui midtrans</i>	<i>Passed</i>
USR010	<i>User tidak mendapatkan pop up booking success</i>	<i>User tidak akan mendapatkan pop up booking success</i>	<i>Passed</i>
	<i>precondition: user tidak menyelesaikan payment</i>		
USR011	<i>User akan mendapatkan pop up booking success</i> <i>Catatan: user sudah melakukan payment</i>	<i>User akan mendapatkan pop up booking success</i>	<i>Passed</i>
PaUSR012	<i>User akan mendapatkan pop up registrasi success</i>	<i>User akan mendapatkan pop up registrasi success</i>	<i>Passed</i>
USR013	<i>User bisa akses halaman artikel</i>	<i>User berhasil redirect ke halaman artikel kesehatan</i>	<i>Passed</i>
USR014	<i>User bisa akses halaman contact dan about us</i>	<i>User berhasil redirect ke halaman about us dan contact</i>	<i>Passed</i>
USR015	<i>User tidak akan bisa registrasi dengan menggunakan email yang sudah terdaftar</i>	<i>User tidak berhasil registrasi</i>	<i>Passed</i>

Hasil pengujian blackbox pada sisi *user* menunjukkan bahwa seluruh skenario fungsionalitas utama sistem berhasil dijalankan sesuai dengan spesifikasi yang diharapkan. Fitur-fitur seperti *login*, registrasi, akses halaman janji temu, pemilihan jadwal konsultasi, hingga proses pembayaran melalui Midtrans dapat diakses dengan baik dan memberikan keluaran yang benar. Validasi input juga berjalan efektif, ditunjukkan dengan munculnya pesan kesalahan saat pengguna mencoba login dengan akun yang belum terdaftar atau melakukan registrasi dengan data yang tidak lengkap maupun email ganda. Selain itu, sistem juga memastikan alur proses pemesanan terkontrol, misalnya pengguna tidak dapat mengakses halaman pembayaran sebelum memilih jadwal, dan notifikasi *pop-up* muncul sesuai kondisi pembayaran atau registrasi. Dengan demikian, pengujian ini membuktikan bahwa modul pengguna telah berfungsi stabil dan memenuhi kebutuhan pasien sebagai *end-user*.

2. Blackbox Admin

Tabel 2. Blackbox Admin

Test Case ID	Test Case	Hasil Yang Diharapkan	Status
ADM001	<i>Admin berhasil login di halaman janji temu dokter</i> <i>notes: menggunakan email yang sudah terdaftar</i>	<i>Admin berhasil login dan redirect ke homepage</i>	<i>Passed</i>
ADM002	<i>Admin tidak berhasil login di halaman janji temu dokter</i> <i>notes: menggunakan email yang belum terdaftar</i>	<i>Admin tidak berhasil login dan akan mendapatkan message error</i>	<i>Passed</i>
ADM003	<i>Admin berhasil redirect ke halaman dashboard admin</i>	<i>Admin berhasil redirect ke halaman dashboard</i>	<i>Passed</i>
ADM004	<i>Admin dapat melihat data permission di dashboard management access</i>	<i>Admin dapat melihat data permission di halaman permission list</i>	<i>Passed</i>
ADM005	<i>Admin dapat melihat data role di dashboard Management access</i>	<i>Admin dapat melihat semua data role di halaman role list</i>	<i>Passed</i>

ADM006	Admin dapat melihat data Type User di <i>dashboard Management Access</i>	Admin dapat melihat semua data type <i>user</i> di halaman type <i>user list</i>	Passed
ADM007	Admin dapat melihat data User di <i>dashboard Management Access</i>	Admin dapat melihat semua data <i>user</i> di halaman <i>user list</i>	Failed
ADM008	Admin dapat melihat data <i>Specialist</i> di <i>dashboard Master Data</i>	Admin dapat melihat semua data <i>specialist</i> di halaman <i>specialist list</i>	Passed
ADM009	Admin dapat melihat data <i>Consultation</i> di <i>dashboard Master Data</i>	Admin dapat melihat semua data <i>consultation</i> di halaman <i>consultation list</i>	Passed
ADM010	Admin dapat melihat data <i>Config Payment</i> di <i>dashboard Master Data</i>	Admin dapat melihat semua data <i>config payment</i> di halaman <i>config payment list</i>	Passed
ADM011	Admin dapat melihat data <i>Doctor</i> di <i>dashboard Operasional</i>	Admin dapat melihat semua data <i>doctor</i> di halaman <i>doctor list</i>	Passed
ADM012	Admin dapat melihat data <i>Appointment</i> di <i>dashboard Operational</i>	Admin dapat melihat semua data <i>Appointment</i> di halaman <i>Appointment list</i>	Passed
ADM013	Admin dapat melihat data <i>Trasanction</i> di <i>dashboard Operasional</i>	Admin dapat melihat semua data <i>Transaction</i> di halaman <i>Transaction list</i>	Passed
ADM014	Admin dapat melihat notifikasi di <i>dashboard</i>	Admin dapat melihat semua Notifikasi	Passed
ADM015	Admin berhasil logout dari dasboard admin	Admin berhasil <i>logout</i> dan akan <i>redirect</i> ke <i>landing page</i>	Passed
ADM016	Admin dapat menambahkan data role di <i>dashboard admin</i>	Admin berhasil menambahkan data <i>role</i>	Passed
ADM017	Admin dapat melihat data <i>role</i>	Admin melihat data <i>role</i> secara detail	Passed
ADM018	Admin dapat mengubah data di <i>dashboard role</i>	Admin berhasil mengubah data <i>role</i>	Passed
ADM019	Admin dapat menambahkan data <i>specialist</i> di halaman <i>specialist</i>	Admin berhasil menambahkan data <i>specialist</i>	Passed
ADM020	Admin dapat mengubah data <i>specialist</i> di halaman <i>specialist list</i>	Admin berhasil mengubah data <i>specialist</i>	Passed
ADM021	Admin dapat menghapus data <i>specialist</i> di halaman <i>specialist list</i>	Admin berhasil menghapus data <i>specialist</i>	Passed
ADM022	Admin dapat menambahkan data <i>consultation</i> di halaman <i>consultation list</i>	Admin dapat menambahkan data <i>consultation</i>	Passed
ADM023	Admin dapat melihat data <i>consultation</i> di halaman <i>consultation list</i> <i>Precondition:</i> data = terbaru	Admin dapat melihat data yang terbaru	Passed
ADM024	Admin dapat mengubah data <i>consultation</i> di halaman <i>consultation list</i>	Admin berhasil mengubah data <i>consultation</i>	Passed
ADM025	Admin dapat menghapus data <i>consultation</i> di halaman <i>consultation list</i>	Admin berhasil menghapus data <i>consultation</i>	Passed
ADM026	Admin dapat mengubah data <i>config payment</i> di halaman <i>config payment list</i>	Admin berhasil mengubah data <i>config payment</i>	Passed

ADM027	Admin dapat menambahkan <i>doctor</i> di halaman <i>doctor list</i>	Admin berhasil menambahkan <i>doctor</i>	Passed
ADM028	Admin dapat melihat data <i>doctor</i> di halaman <i>doctor list</i> <i>Precondition:</i> data = terbaru	Admin dapat melihat data <i>doctor</i> terbaru	Passed
ADM029	Admin dapat mengubah data <i>doctor</i> di halaman <i>doctor list</i>	Admin berhasil mengubah data <i>doctor</i>	Passed
ADM030	Admin dapat menghapus <i>doctor</i> di halaman <i>doctor list</i>	Admin berhasil menghapus <i>doctor</i>	Passed

Pengujian *blackbox* pada sisi admin memperlihatkan bahwa hampir seluruh fungsi pengelolaan data dan kontrol sistem dapat dijalankan dengan baik, mulai dari autentikasi login, akses dashboard, pengelolaan permission, role, tipe user, spesialisasi dokter, hingga manajemen data konsultasi, pembayaran, dokter, janji temu, transaksi, dan notifikasi. Admin juga dapat melakukan operasi tambah, ubah, dan hapus pada berbagai entitas data sesuai hak aksesnya, yang menegaskan bahwa sistem mendukung fleksibilitas dan kelengkapan dalam administrasi klinik. Dari total skenario, hanya satu kasus yaitu ADM007 yang mengalami kegagalan karena validasi data dan rute fitur yang tidak sesuai. Namun, kegagalan ini tidak bersifat kritis dan tidak mengganggu operasional utama, sehingga sistem tetap dapat digunakan secara efektif. Secara keseluruhan, hasil pengujian membuktikan bahwa modul admin memiliki tingkat reliabilitas tinggi dalam mengelola layanan kesehatan berbasis digital.

Hasil pengujian fungsional menggunakan metode *black-box* menunjukkan bahwa *prototype* aplikasi berhasil menjalankan hampir seluruh skenario pengujian baik pada sisi *user* maupun administrator. Penilaian tingkat keberhasilan pada tahap ini dihitung menggunakan pendekatan *task success rate* sebagaimana dikemukakan oleh Nielsen & Budiu (2001), yaitu menilai efektivitas sistem berdasarkan proporsi tugas yang berhasil diselesaikan dibandingkan total tugas yang diujikan. Setiap skenario diklasifikasikan ke dalam kategori *success*, *partial success*, dan *fail*, namun pada penelitian ini tidak ditemukan *partial success*, sehingga perhitungan didasarkan sepenuhnya pada jumlah tugas yang berhasil diselesaikan secara penuh.

Dari total 15 skenario pengujian untuk *user*, seluruhnya berhasil dijalankan dengan tingkat keberhasilan 100%. Sementara itu, pada sisi administrator, 29 dari 30 skenario berhasil dengan tingkat keberhasilan 96,67%. Jika digabungkan, tingkat keberhasilan keseluruhan mencapai 97,8%, yang menunjukkan bahwa sistem mampu menjalankan fungsi inti secara stabil dan sesuai spesifikasi. Hasil pengujian yang menunjukkan tingkat keberhasilan keseluruhan sebesar 97,8% berada pada kategori sangat tinggi apabila dibandingkan dengan standar efektivitas *prototype* menurut Nielsen & Budiu (2001), yang menetapkan bahwa nilai di atas 80% telah mencerminkan tingkat performa yang baik (Oliveira et al., 2021). Dengan demikian, tingkat keberhasilan yang dicapai mengindikasikan bahwa rancangan alur interaksi, struktur antarmuka, serta stabilitas fungsional aplikasi telah memenuhi ekspektasi pengguna dan selaras dengan parameter efektivitas yang direkomendasikan dalam literatur *usability*.

Secara teoretis, capaian tersebut dapat dijelaskan melalui kerangka *Technology Acceptance Model* (TAM), khususnya pada aspek *perceived usefulness* dan *perceived ease of use*. Tingginya tingkat keberhasilan skenario pengujian menunjukkan bahwa pengguna mampu menjalankan proses inti tanpa hambatan, sehingga persepsi kegunaan dan kemudahan penggunaan sistem meningkat secara simultan. Model TAM menjelaskan bahwa ketika fungsi utama seperti pemesanan janji, pengelolaan jadwal dokter, *reschedule*, dan pembayaran digital berjalan dengan lancar, maka tingkat penerimaan teknologi cenderung meningkat (Widyassari & Sirojunafis, 2023; Syahputra et al., 2025; Yuniarti et al., 2025). Dalam konteks penelitian ini, keberhasilan proses pembayaran otomatis melalui notifikasi Midtrans, pembaruan status janji temu secara *real-time*, serta kemudahan navigasi antarmuka mendukung terbentuknya pengalaman pengguna yang positif, sebagaimana dijelaskan dalam kerangka *user experience model* bahwa konsistensi, efisiensi alur, dan keandalan sistem memengaruhi kepuasan dan keinginan pengguna untuk terus menggunakan sistem.

Dari perspektif rekayasa perangkat lunak, pemilihan arsitektur *Model View Controller* (MVC) dan penerapan metode *prototyping* mendukung penelitian Paolone et al. (2021) serta Faisal & Nakayama (2024). Kedua studi tersebut menegaskan bahwa modularitas, fleksibilitas, dan kemudahan pemeliharaan merupakan karakteristik penting yang harus dimiliki aplikasi kesehatan modern. *Prototype* yang dikembangkan dalam penelitian ini menunjukkan bahwa kombinasi MVC dan prototyping mampu menghasilkan sistem yang adaptif, mudah diuji, serta sesuai dengan kebutuhan operasional yang berubah secara dinamis di lingkungan klinik primer. Dengan demikian, penelitian ini memberikan kontribusi empiris yang memperkaya literatur, menunjukkan bahwa integrasi alur pemesanan, pengelolaan jadwal dokter, dan automasi pembayaran dalam satu sistem terpadu dapat meningkatkan efisiensi, akurasi, dan kepuasan layanan kesehatan.

KESIMPULAN

Penelitian ini berhasil merancang dan mengembangkan *prototype* aplikasi penjadwalan konsultasi dokter berbasis web menggunakan framework Laravel dan arsitektur *Model View Controller* melalui pendekatan *prototyping*. Sistem yang dikembangkan mampu mendukung alur pendaftaran pasien, pemesanan janji temu, pengelolaan jadwal praktik dokter, serta pembaruan status pembayaran secara otomatis melalui integrasi API Midtrans. Hasil pengujian fungsional menggunakan metode *black-box testing* menunjukkan bahwa *prototype* telah bekerja sesuai spesifikasi, dengan tingkat keberhasilan 100% pada sisi pengguna dan 96,67% pada sisi administrator, sehingga tingkat keberhasilan keseluruhan mencapai 97,8%. Temuan ini menunjukkan bahwa *prototype* telah memenuhi kebutuhan operasional klinik primer serta menjawab tujuan penelitian yang menargetkan peningkatan efisiensi layanan, keakuratan pencatatan, dan kemudahan interaksi pengguna.

Secara teoretis, penelitian ini berkontribusi dengan memperkuat bukti empiris mengenai efektivitas arsitektur MVC dalam sistem informasi kesehatan serta menegaskan relevansi pendekatan *prototyping* sebagai metode pengembangan yang responsif terhadap kebutuhan pengguna. Secara praktis, *prototype* ini menawarkan solusi terstruktur bagi digitalisasi layanan di klinik primer melalui automasi antrian, fasilitas *reschedule*, dan pengelolaan data pasien maupun dokter yang lebih sistematis. Meskipun demikian, penelitian ini memiliki keterbatasan, antara lain belum dilakukannya pengujian *User Acceptance Testing* (UAT) secara formal serta belum mencakup evaluasi *usability* yang komprehensif. Penelitian lanjutan direkomendasikan untuk memperluas cakupan pengujian, termasuk analisis performa sistem dan evaluasi pengalaman pengguna berbasis instrumen kuantitatif, guna memperkuat kesiapan implementasi pada skala operasional yang lebih luas.

DAFTAR PUSTAKA

- Azmi, Z. U., Mailintina, Y., & Efkelin, R. (2024). Faktor-Faktor yang Berhubungan dengan Lama Waktu Tunggu Rawat Jalan Pasien BPJS Puskesmas Kecamatan Penjaringan. *MAHESA : Mahayati Health Student Journal*, 4(11), 5179–5191. <https://doi.org/10.33024/mahesa.v4i11.17166>
- Betancor, P. K, Boehringer, D., Jordan, J., Lüchtenberg, C., Lambeck, M., Ketterer, M. C., Reinhard, T., & Reich, M. (2025). Efficient Patient Care In The Digital Age: Impact Of Online Appointment Scheduling In A Medical Practice And A University Hospital On The “No-Show”-Rate. *Frontiers in Digital Health*, 7, 1–10. <https://doi.org/10.3389/fdgh.2025.1567397>
- Cha, D. (2023). Digital healthcare: the new frontier of holistic and efficient care. *Clinical and Experimental Emergency Medicine*, 10, 235–237. <https://doi.org/10.15441/ceem.23.054>
- Faisal, H., & Nakayama, M. (2024). Implementation of the World Health Organization Minimum Dataset for Emergency Medical Teams to Create Disaster Profiles for the Indonesian SATUSEHAT Platform Using Fast Healthcare Interoperability Resources: Development and Validation Study. *JMIR Medical Nformatics*, 12, 1–13. <https://doi.org/10.2196/59651>

- Greenup, E. P., & Best, D. (2025). Systematic Review And Meta-Analysis Of No Show Or Non-Attendance Rates Among Telehealth And In-Person Models Of Care. *BMC Health Serv Res* 25, 25(663), 1–7. <https://doi.org/10.1186/s12913-025-12826-2>
- Heryawan, L., Mori, Y., Yamamoto, G., Kume, N., Lazuardi, L., Fuad, A., & Kuroda, T. (2025). Fast Healthcare Interoperability Resources (FHIR)-Based Interoperability Design in Indonesia: Content Analysis of Developer Hub's Social Networking Service. *JMIR Formative Research*, 9, 1-12. <https://doi.org/10.2196/51270>
- Habibi, M. R. M., Abadi, F. M., Tabesh, H., Vakili-Arki, H., Abu-Hanna, A., Ghaddaripour, K., & Eslami, S. (2024). Evaluation Of No-Show Rate In Outpatient Clinics With Open Access Scheduling System: A Systematic Review. *Health Science Reports*, 7(7), e2160.
- Nielsen, J., & Budiu, R. (2001). *Success Rate: The Simplest Usability Metric*. Nngroup.
- Oliveira, E., Branco, A. C., Carvalho, D., Sacramento, E., Tymoshchuk, O., Pedro, L., Antunes, M. J., Almeida, A. M., & Ramos, F. (2021). End-user Evaluation of a Mobile Application Prototype for Territorial Innovation. *Proceedings of the 23rd International Conference on Enterprise Information Systems (ICEIS 2021)*, 2, 495–504. <https://doi.org/10.5220/0010479104950504>
- Paolone, G., Paesani, R., Marinelli, M., & Di Felice, P. (2021). Empirical Assessment of the Quality of MVC Web Applications Returned by xGenerator. *Computers*, 10(20), 1-18. <https://doi.org/10.3390/computers10020020>
- Putri, S. N. E., Firdaus, F., Mpuhaji, M. D. A., Gunawan, I. M. A. O., Indrawan, G., & Fitriati, I. (2025). Optimisasi Implementasi Sistem Informasi Reminder Treatmentpada Pasien Berbasis SMS Gateway. *DECODE: Jurnal Pendidikan Teknologi Informasi*, 5(1), 1–11. <http://dx.doi.org/10.51454/decode.v5i1.832>
- Ramadani, N., Fauziah, R., & Efendi, Z. (2023). Sistem Pakar Diagnosis Penyakit Epilepsi pada Anak di RSUD dengan Forward Chaining Method. *DECODE: Jurnal Pendidikan Teknologi Informasi*, 3(2), 184–192. <http://dx.doi.org/10.51454/decode.v3i2.164>
- Ramdoni, M. M., & Herdiansyah, M. I. (2023). Pengembangan Sistem Informasi Konsultasi Dokter Menggunakan Framework Laravel. *Journal of Information System Research (JOSH)*, 4(3), 831–839. <https://doi.org/10.47065/josh.v4i3.3276>
- Santalia, G., Arif, Y., Maisa, E. A., & Helfiyani. (2023). Waktu Tunggu Pasien Rawat Jalan di RSUD Dr. Adnan WD Kota Payakumbuh. *Jurnal Keperawatan Jiwa (JKJ): Persatuan Perawat Nasional Indonesia*, 11(3), 723–730. <http://dx.doi.org/10.26714/jkj.11.3.2023.723-730>
- Syahputra, K. W., Haris, M. S., & Pradini, R. S. (2025). Analisis Penerimaan Sistem Pendaftaran Online Pasien Rawat Jalan Di Rs Radjiman Wediodiningrat Menggunakan Teknology Acceptance Model. *JATI: Jurnal Mahasiswa Teknik Informatika*, 9(4), 5860–5866. <https://doi.org/10.36040/jati.v9i4.13917>
- Widyassari, A. P., & Sirojunnafis, A. (2023). Pengembangan Sistem Informasi Antrian Pasien Menggunakan Technology Acceptance Model. *JSITIK*, 2(1), 54–67. <https://doi.org/10.53624/jsitik.v2i1.319>
- Yuniarti, S., Indriyani, Y., & Setyawan, A. (2025). Technology Acceptance Model (TAM) sebagai Alat Ukur untuk Mengetahui Pengaruh Penerapan SIMRS Terhadap Efektivitas Pelayanan di RSUD Ir. Soekarno Kabupaten Sukoharjo. *Jurnal Ners*, 9(4), 5570–5585. <https://doi.org/10.31004/jn.v9i4.47065>