

Implementasi Aplikasi *Reminder* Pasien Berbasis Web Sebagai *Software as a Service* Menggunakan *Waterfall Model*

IMPLEMENTATION OF WEB-BASED PATIENT *REMINDER* APPLICATION AS *SOFTWARE AS A SERVICE* USING THE *WATERFALL MODEL*

I Putu Mahendra Adi Wardana

Universitas Hindu Indonesia: Jl. Sangalangit, Penatih, Denpasar, 80238, Indonesia
Program Studi Sistem Informasi, Fakultas Teknologi dan Sains, Universitas Hindu Indonesia
e-mail: mahendrawardana@unhi.ac.id

Abstrak

Klinik adalah fasilitas kesehatan yang sangat dikenal oleh masyarakat. Dalam menjalankan operasinya, klinik mulai menggunakan teknologi untuk membantu pekerjaan mereka. Beberapa halangan masih ditemui dalam operasional klinik seperti keterbatasan tenaga kerja untuk memantau pasien. Kondisi ini membutuhkan inovasi dalam teknologi untuk memberikan layanan yang lebih baik bagi pasien. Melalui pendekatan *software as a service*, peneliti membuat sebuah aplikasi *reminder* pasien berbasis web untuk mengatasi permasalahan operasional yang ditemui di klinik. Aplikasi ini dibangun dengan menggunakan *waterfall model* yang terdiri dari tahap analisis kebutuhan, desain, implementasi, pengujian, dan pemeliharaan. Hasilnya adalah aplikasi *reminder* pasien berbasis web yang telah diuji dengan metode black box testing dan sudah digunakan oleh operator klinik. Dengan memperhatikan langkah-langkah dalam *waterfall model*, dapat dicapai aplikasi *reminder* pasien berbasis web yang telah diimplementasikan dengan baik.

Kata kunci — *Reminder*, Pasien, Web, Waterfall Model, Klinik

Abstract

Clinics are health facilities that are well known to the public. In carrying out their operations, clinics are starting to use technology to help their work. Some obstacles are still encountered in clinical operations, such as limited manpower to monitor patients. This condition requires innovation in technology to provide better services for patients. Using a software-as-a-service approach, researchers created a web-based patient reminder application to address operational problems encountered in clinics. This application is built using the Waterfall Model, which consists of the stages of needs analysis, design, implementation, testing, and maintenance. The result is a web-based patient reminder application that has been tested using the black box testing method and is being used by clinic operators. By paying attention to the steps in the Waterfall Model, a well-implemented web-based patient reminder application can be achieved.

Kata kunci — *Reminder*, Patient, Web, Waterfall Model, Clinic

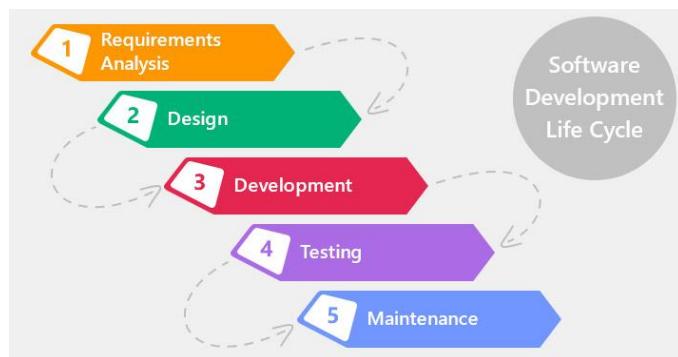
1. PENDAHULUAN

Pada era teknologi saat ini, kebutuhan informasi sangat penting dan merupakan bagian yang tak terpisahkan dalam kehidupan masyarakat [1]. Masyarakat sudah mulai menggunakan fasilitas kesehatan dengan rutin, salah satunya klinik. Dalam hal ini, manajemen operasional klinik sudah mulai menggunakan teknologi untuk mempermudah pekerjaan mereka [2]. Namun, masih ada kendala yang sering terjadi dalam klinik, yaitu kurangnya tenaga kerja yang berisiko mempengaruhi citra klinik di masyarakat [3]. Untuk mengatasi kendala tersebut, pemanfaatan teknologi menjadi solusi yang tepat. Hal ini membuat adanya keperluan dalam membuat suatu *software as a service* yang mempermudah operasional klinik. Sebelumnya, operasional klinik dilakukan secara konvensional, namun saat ini dilakukan secara digital dan dapat diakses dimanapun, kapanpun dan siapapun yang terhubung di jaringan internet. *Software as a service* yang dibuat dalam hal ini meliputi sistemasi operasional yang terintegrasi pada WhatsApp dan Email. Sistemasi operasional ini meliputi pembuatan *reminder* dan manajemen data *appointment*, *consultation*, tindakan/perawatan/*operation*, hingga *control & follow up*. Dengan demikian, klinik kesehatan dapat menjangkau pasien mereka dengan lebih baik dan

mengurangi risiko terjadinya kendala dalam follow up pasien. Pembuatan *software as a service* ini menggunakan *Waterfall Model* yang memastikan bahwa proses pengembangan *software* dilakukan secara bertahap dan sistematis [4]. Model ini membantu dalam memahami proses dan memastikan bahwa setiap tahap dalam pengembangan *software* dilakukan dengan benar dan memenuhi spesifikasi yang dibutuhkan [5]. Dengan adanya *software as a service* yang dibuat menggunakan *waterfall model*, klinik kesehatan dapat mempermudah operasional mereka dan menjangkau pasien dengan lebih baik. Hal ini akan berdampak positif bagi klinik kesehatan dan masyarakat dalam memperoleh informasi dan layanan kesehatan yang lebih baik.

2. METODE PENELITIAN

Para pelaku bisnis dalam dunia kesehatan memiliki banyak hal yang harus dipikirkan setiap hari, dan memastikan bahwa pasien memiliki jadwal kontrol yang tepat adalah salah satu dari banyak tugas mereka. Oleh karena itu, implementasi aplikasi *reminder* pasien berbasis web sebagai *software as a service* menggunakan *waterfall model* adalah hal yang sangat penting untuk dilakukan.



Gambar 1. *Waterfall Model*

Langkah pertama dalam *waterfall model* adalah fase perencanaan [6]. Dalam fase ini, peneliti sebagai pengembang sistem (*developer*) akan mengumpulkan informasi dan membuat spesifikasi sistem yang diinginkan [7]. Mereka akan menentukan kebutuhan pasien dan dokter serta bagaimana aplikasi harus berfungsi untuk memenuhi kebutuhan tersebut. Langkah kedua adalah fase analisis [8]. Dalam fase ini, *developer* akan menganalisis informasi yang dikumpulkan dalam fase perencanaan dan membuat desain aplikasi [9]. Mereka akan membuat diagram sistem, alur kerja, dan dokumentasi yang diperlukan untuk menjelaskan bagaimana aplikasi akan berfungsi [10]. Langkah ketiga adalah fase pengembangan [11]. Dalam fase ini, *developer* akan membuat kode aplikasi dan melakukan pengujian [12]. Mereka akan menguji setiap fitur dan memastikan bahwa aplikasi berfungsi sesuai dengan spesifikasi yang dibuat dalam fase perencanaan dan analisis. Langkah keempat adalah fase pengujian [13]. Dalam fase ini, *developer* akan melakukan pengujian ulang untuk memastikan bahwa aplikasi bekerja dengan baik dan memenuhi kebutuhan pasien dan dokter. Mereka akan melakukan pengujian beta untuk memastikan bahwa aplikasi bekerja dengan baik dan siap untuk diluncurkan [14]. Langkah terakhir adalah fase peluncuran dan pemeliharaan. Dalam fase ini, aplikasi *reminder* pasien berbasis web sebagai *software as a service* akan diluncurkan dan diterima oleh pasien dan dokter. *Developer* akan memantau dan memelihara aplikasi agar tetap bekerja dengan baik dan memenuhi kebutuhan pasien dan dokter. Dengan mengikuti *waterfall model*, implementasi aplikasi *reminder* pasien berbasis web sebagai *software as a service* akan berjalan dengan lancar dan memenuhi kebutuhan pasien dan dokter. *Developer* akan memastikan bahwa aplikasi berjalan dengan baik.

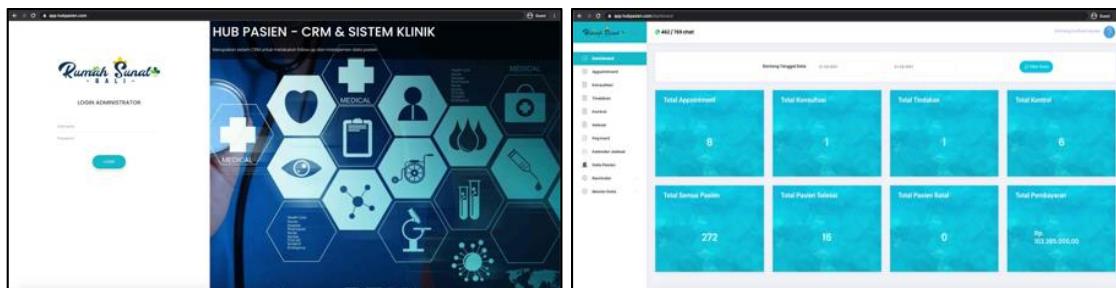
3. HASIL DAN PEMBAHASAN

Fase perencanaan yang telah dilaksanakan, solusi dari permasalahan yang dihadapi klinik terkait sistemasi operasional secara digital tersebut adalah dengan mengembangkan perangkat lunak yang terhubung dengan jaringan internet agar bisa diakses dimanapun dan kapanpun.

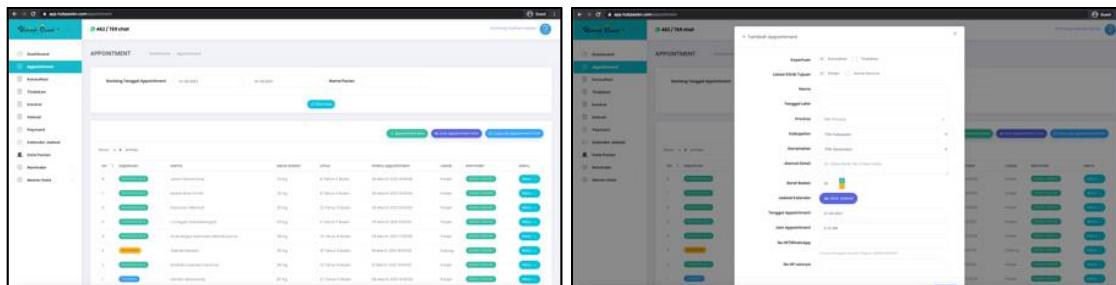
Konsep SaaS akan menjadi kunci dalam penggunaan Perangkat Lunak beserta konsep *Design Responsive* mempermudah *user* untuk menggunakan aplikasi ini disegala perangkat yang mendukung *browser*. Fitur aplikasi yang menjadi solusi dalam permasalahan klinik adalah *create an appointment by patient & admin klinik*, manajemen janji/pertemuan pasien dengan klinik, manajemen konsultasi, manajemen tindakan, manajemen kontrol, *follow up reminder*, integrasi WhatsApp dan e-mail, manajemen laporan transaksi keuangan, manajemen kalender perjanjian antara pasien dengan klinik hingga *application setting*.

Fitur aplikasi tersebut dirancang dan dikembangkan dengan seksama agar dapat terintegrasi dengan baik dan tepat. Dilengkapi dengan *Reminder* yang terintegrasi dengan WhatsApp dan Email menjadi nilai tambah secara fungsional. *Software as a service* menjadi salah satu kebutuhan yang memenuhi solusi dari permasalahan di klinik. Pengembangan aplikasi dimulai dari pembuatan *flowchart*, pembuatan *wireframe*, konfirmasi desain kasar kepada pemilik klinik, menganalisa *user experience*, pembuatan *user interface/front-end*, pengembangan *back-end*, integrasi *front-end* dengan *back-end*, instalasi *software* pada *server* yang sudah memiliki domain, melakukan testing aplikasi, *problem solving* aplikasi hingga *implementation & maintenance* pada aplikasi.

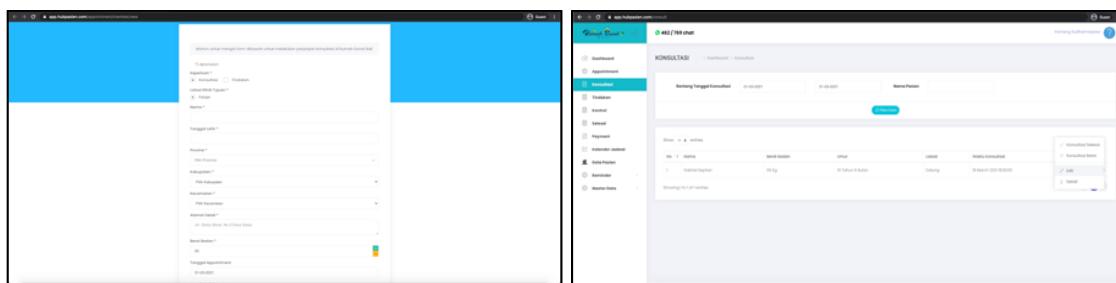
Fitur-fitur yang telah dirumuskan sebelumnya, diimplementasikan dalam sebuah antar muka/*front-end* yang dapat dilihat dan digunakan oleh *user* / klinik. *User interface* dari aplikasi ditampilkan pada gambar berikut.



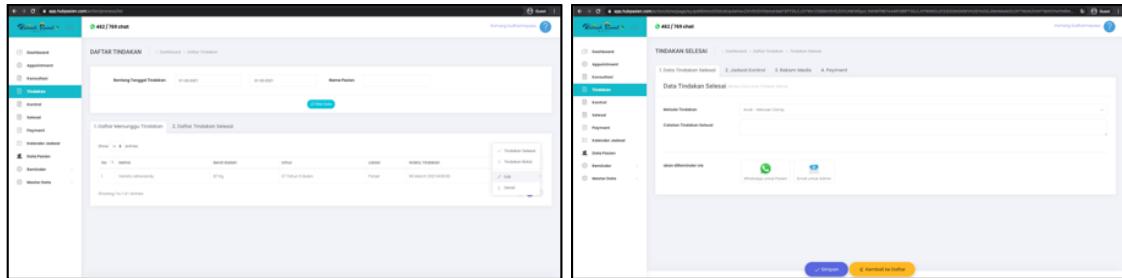
Gambar 2. Implementasi Halaman Login dan Halaman Dashboard



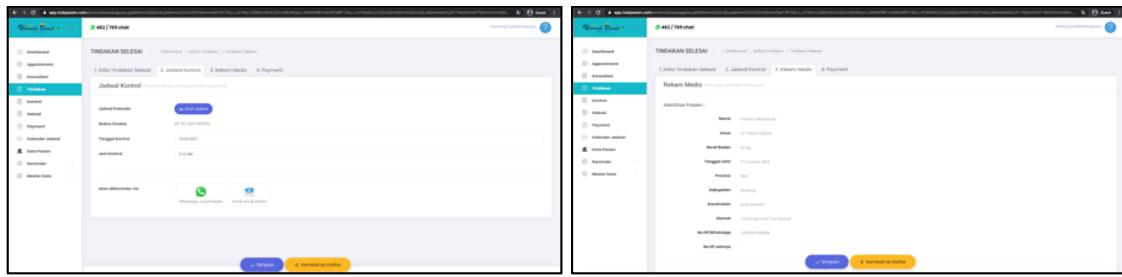
Gambar 3. Implementasi Halaman Manajemen Data Appointment dan Halaman Create Appointment oleh Admin Klinik



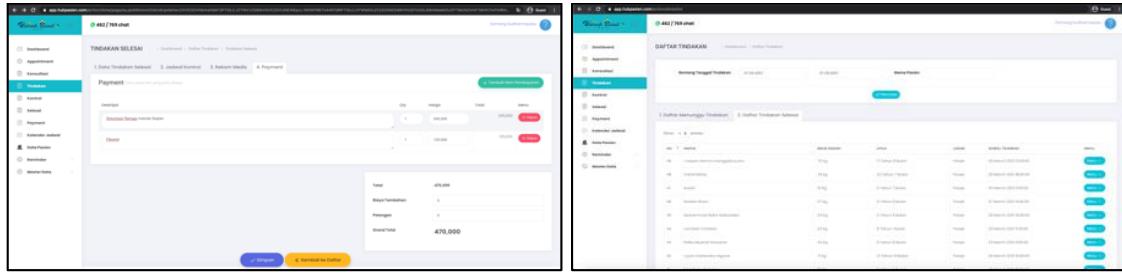
Gambar 4. Implementasi Halaman Create Appointment oleh Calon Pelanggan Klinik dan Halaman Manajemen Konsultasi



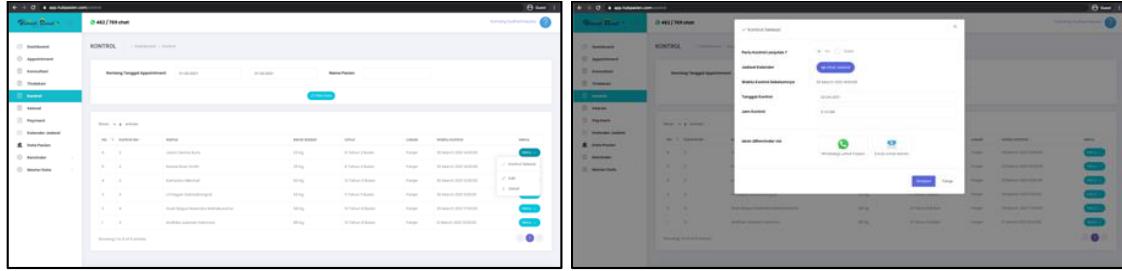
Gambar 5. Implementasi Halaman Manajemen Tindakan/Operasi dan Halaman Input Data Tindakan



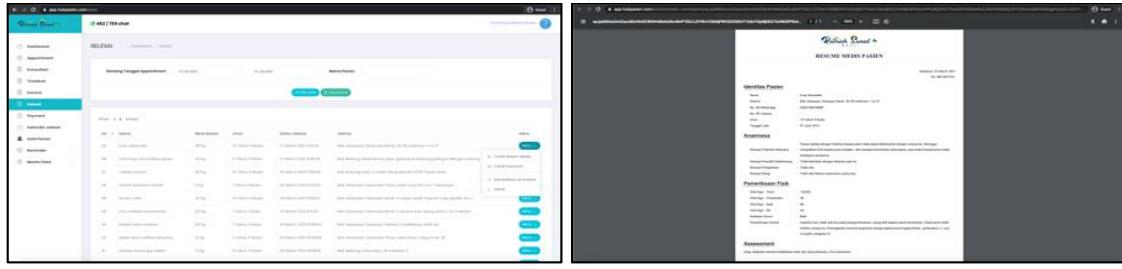
Gambar 6. Implementasi Halaman Input Jadwal Kontrol dan Halaman Input Data Rekam Medis Pasien



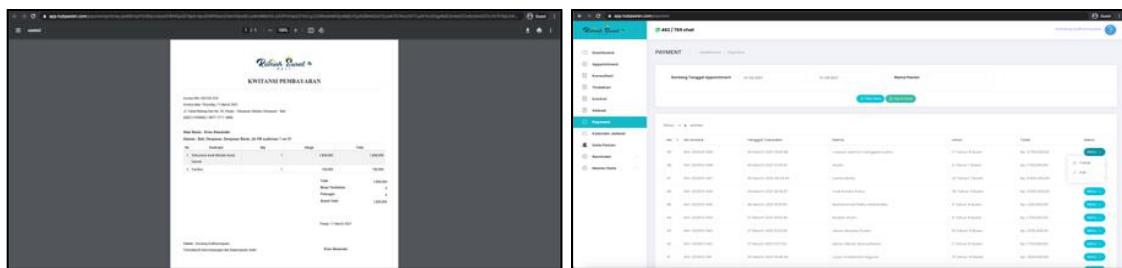
Gambar 7. Implementasi Halaman Input Transaksi Pembayaran dan Halaman Daftar Tindakan yang sudah Selesai



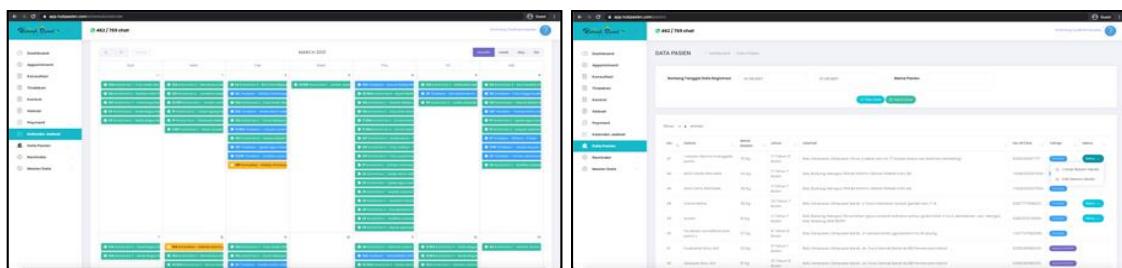
Gambar 8. Implementasi Halaman Manajemen Kontrol dan Kontrol Selesai



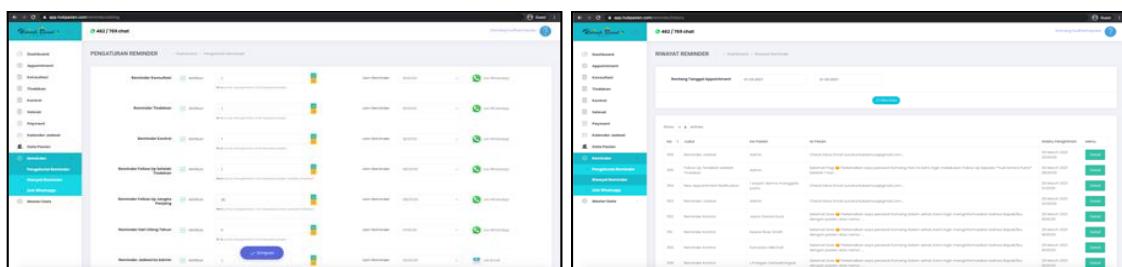
Gambar 9. Implementasi Halaman Manajemen Data Selesai Kontrol dan Halaman Cetak Rekam Medis Pasien



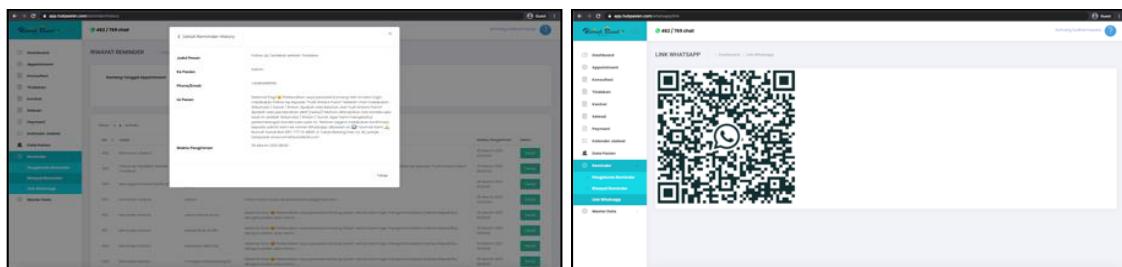
Gambar 10. Implementasi Halaman Cetak Kwitansi Pembayaran dan Halaman Manajemen Data Transaksi Pembayaran



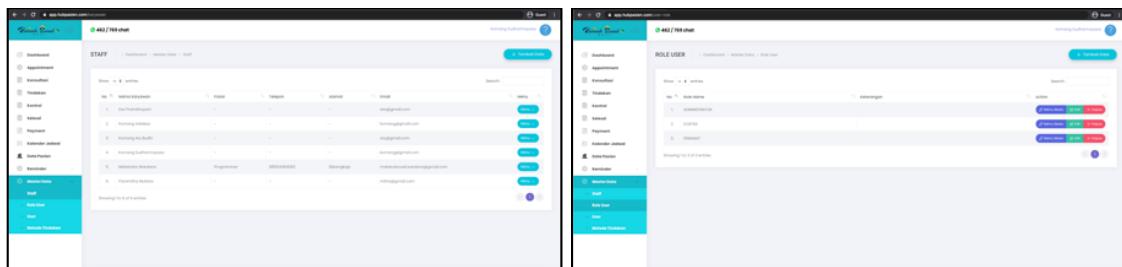
Gambar 11. Implementasi Halaman View Jadwal Dokter dan Perawat dan Halaman Manajemen Data Pasien



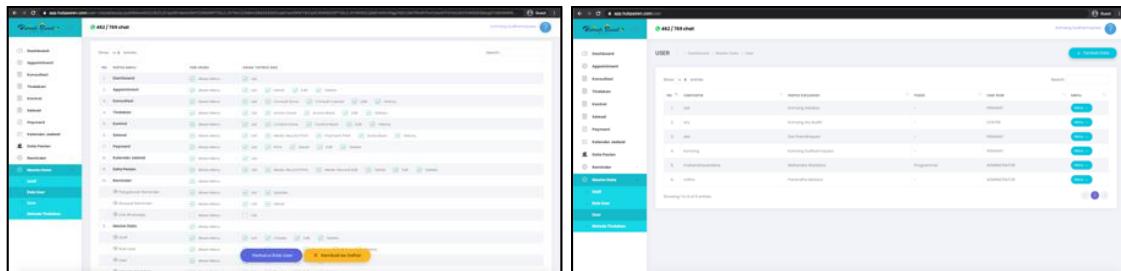
Gambar 12. Implementasi Halaman Pengaturan Aplikasi dan Halaman Riwayat *Reminder*



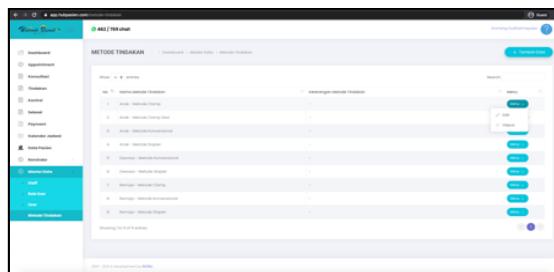
Gambar 13. Implementasi Halaman Detail Riwayat *Reminder* dan Halaman Integrasi WhatsApp Klinik dengan Aplikasi



Gambar 14. Implementasi Halaman Manajemen Data *Staff* dan Halaman Manajemen *User Role*



Gambar 15. Implementasi Halaman Edit *User Role* dan Halaman Manajemen *Staff*



Gambar 16. Implementasi Halaman Manajemen Data Tindakan

Black box testing adalah metode pengujian aplikasi tanpa mengetahui bagaimana sistem intern dalam aplikasi tersebut bekerja [15]. Dalam implementasi Aplikasi *Reminder* Pasien Berbasis Web sebagai *Software as a service* yang menggunakan *Waterfall Model*, *tester* akan melakukan pengujian dengan menekan tombol, memasukkan data, dan melihat apakah hasil yang diterima sesuai dengan yang diharapkan [16]. Tujuan dari pengujian ini adalah untuk memastikan bahwa aplikasi dapat berfungsi dengan baik dan memberikan output yang sesuai untuk pengguna akhir, tanpa memperhatikan bagaimana aplikasi tersebut mencapai hasil tersebut [17]. Berikut adalah contoh hasil pengujian menggunakan *black box testing*.

Tabel 1. Pengujian Halaman *Login* Menggunakan Black Box Testing

Tabel 4. Pengujian Alur dan Menggunakan Black Box Testing			
Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
<i>User</i> dapat melakukan <i>login</i> menggunakan <i>username</i> dan <i>password</i> yang benar	Sistem diharapkan dapat melakukan <i>login</i> menggunakan <i>username</i> atau <i>password</i> yang benar	<i>User</i> berhasil masuk ke halaman beranda	Valid
<i>User</i> memasukan <i>username</i> dan/atau <i>password</i> salah	Sistem diharapkan dapat menolak data karena tidak ada atau salah	Sistem berhasil menolak akses <i>login</i> dan menampilkan pesan <i>username</i> atau <i>password</i> salah	Valid

Tabel 2. Pengujian Halaman Beranda Menggunakan Black Box Testing

Tabel 2. Pengujian Harian Harian Beranda Menggunakan Black Box Testing			
Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
<i>User</i> dapat menuju halaman <i>appointment</i> ketika mengklik menu <i>appointment</i>	Sistem diharapkan dapat menampilkan halaman <i>appointment</i>	<i>User</i> berhasil masuk ke halaman <i>appointment</i>	Valid
<i>User</i> dapat menuju halaman konsultasi ketika mengklik menu konsultasi	Sistem diharapkan dapat menampilkan halaman konsultasi	<i>User</i> berhasil masuk ke halaman konsultasi	Valid

Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
<i>User</i> dapat menuju halaman tindakan ketika mengklik menu tindakan	Sistem diharapkan dapat menampilkan halaman tindakan	<i>User</i> berhasil masuk ke halaman tindakan	Valid
<i>User</i> dapat menuju halaman kontrol ketika mengklik menu kontrol	Sistem diharapkan dapat menampilkan halaman kontrol	<i>User</i> berhasil masuk ke halaman kontrol	Valid
<i>User</i> dapat menuju halaman selesai ketika mengklik menu selesai	Sistem diharapkan dapat menampilkan halaman selesai	<i>User</i> berhasil masuk ke halaman selesai	Valid
<i>User</i> dapat menuju halaman <i>payment</i> ketika mengklik menu <i>payment</i>	Sistem diharapkan dapat menampilkan halaman <i>payment</i>	<i>User</i> berhasil masuk ke halaman <i>payment</i>	Valid
<i>User</i> dapat menuju halaman kalender jadwal ketika mengklik menu kalender jadwal	Sistem diharapkan dapat menampilkan halaman kalender jadwal	<i>User</i> berhasil masuk ke halaman kalender jadwal	Valid
<i>User</i> dapat menuju halaman data pasien ketika mengklik menu data pasien	Sistem diharapkan dapat menampilkan halaman data pasien	<i>User</i> berhasil masuk ke halaman data pasien	Valid
<i>User</i> dapat menuju halaman <i>reminder</i> ketika mengklik menu <i>reminder</i>	Sistem diharapkan dapat menampilkan halaman <i>reminder</i>	<i>User</i> berhasil masuk ke halaman <i>reminder</i>	Valid
<i>User</i> dapat menuju halaman <i>master data</i> ketika mengklik menu <i>master data</i>	Sistem diharapkan dapat menampilkan halaman <i>master data</i>	<i>User</i> berhasil masuk ke halaman <i>master data</i>	Valid

Langkah terakhir dalam implementasi aplikasi *reminder* pasien berbasis web sebagai *software as a service* menggunakan *Waterfall Model* adalah fase peluncuran dan pemeliharaan [18]. Dalam fase ini, aplikasi tersebut akan diluncurkan dan dapat digunakan oleh operator klinik untuk mengingatkan pasien mengenai jadwal kontrol dan tindakan medis yang harus dilakukan. Operator klinik juga bertanggung jawab untuk melakukan pemeliharaan dan perbaikan aplikasi jika terjadi masalah atau *error*. Pemeliharaan dan perbaikan ini dilakukan agar aplikasi tetap dapat berfungsi dengan baik dan memenuhi kebutuhan pasien dan operator klinik. Berikut adalah dokumentasi operator klinik dalam menggunakan aplikasi.



Gambar 16. Penggunaan Aplikasi oleh Operator Klinik

4. KESIMPULAN

Implementasi aplikasi *reminder* pasien berbasis web sebagai *software as a service* menggunakan *Waterfall Model* telah dilakukan dengan baik. Tahapan pengembangan yang menggunakan *Waterfall Model* telah mendukung terwujudnya suatu aplikasi yang dapat digunakan klinik dalam kontrol operasional pada sebuah klinik. Fitur aplikasi berhasil diimplementasikan adalah *create an appointment by patient* dan admin klinik, manajemen

janji/pertemuan pasien dengan klinik, manajemen konsultasi, manajemen tindakan, manajemen kontrol, *follow up reminder*, integrasi whatsapp dan email, manajemen *report* transaksi keuangan, manajemen kalender perjanjian antara pasien dengan klinik hingga *application setting*. Aplikasi juga telah melewati tahapan pengujian menggunakan *black box testing* dan telah digunakan oleh operator klinik.

DAFTAR PUSTAKA

- [1] H. Amalia, T. Retnasari, and S. Rachmawati, “Pemanfaatan Teknologi Informasi Untuk Meningkatkan Pelayanan Akademik Rumah Tahfidz dan TPQ Sakinah Cipayung Jakarta Timur,” *Jurnal Abdimas BSI*, vol. 3, no. 2, pp. 228–235, Jul. 2020, doi: 10.31294/jabdmas.v3i2.8550.
- [2] E. Mulyadi, A. Trihariprasetya, and I. G. Wiryawan, “Penerapan Sistem Presensi Mobile dengan Menggunakan Sensor GPS (Klinik Pratama X Di Jember),” *j. nas. pendidik. teknik. inform.*, vol. 9, no. 1, p. 11, Apr. 2020, doi: 10.23887/janapati.v9i1.23174.
- [3] D. P. Putra *et al.*, “Rancang Bangun Sistem Informasi Booking Antrian pada Klinik Berbasis Website,” *Jurnal Riset Sistem Informasi*, vol. 1, no. 2, pp. 76–87, 2023, doi: 10.32795/resi.v1i2.3638.
- [4] P. G. S. C. Nugraha, N. W. Wardani, and I. W. Sukarmayasa, “Rancang Bangun Sistem Informasi *Software Point of Sale (POS)* dengan Metode Waterfall Berbasis Web,” *JST*, vol. 10, no. 1, pp. 91–103, 2021, doi: 10.23887/jstundiksha.v10i1.29748.
- [5] N. M. M. R. Desmayani, N. W. Wardani, P. G. S. C. Nugraha, and G. S. Mahendra, “Sistem Informasi Laporan Keuangan pada Salon Berbasis Website Dengan Metode SDLC,” *Sistem Inf. Komput. Terap. Indones.*, vol. 4, no. 2, pp. 68–77, Dec. 2021, doi: 10.33173/jsikti.118.
- [6] I. K. A. Asmarajaya, K. O. Sanjaya, D. M. D. U. Putra, G. S. Mahendra, and F. N. U. Hasanah, “Sistem Informasi Keuangan pada Perusahaan Kost Elit dengan Metode Waterfall,” *SWABUMI*, vol. 9, no. 2, pp. 100–108, Sep. 2021, doi: 10.31294/swabumi.v9i2.10970.
- [7] A. A. Loujien, Z. A. Anwari, P. D. Maharanny, K. D. A. Kusuma, and G. S. Mahendra, “Penginputan Data Donor Darah PMI Menggunakan Aplikasi Berbasis Website,” *Jurnal Teknologi Ilmu Komputer*, vol. 1, no. 1, pp. 1–6, 2022.
- [8] D. M. J. Putra, I. N. Y. Anantra, P. D. J. Pratama, I. K. S. M. Wikrama, I. P. A. Kusuma, and G. S. Mahendra, “Pengembangan Layanan Sistem Informasi Ikan Segar pada Kabupaten Buleleng BULLfresh Berbasis Web,” *Jurnal Teknologi Ilmu Komputer*, vol. 1, no. 1, pp. 23–29, 2022.
- [9] N. P. A. Rainita, A. A. I. C. Athalia, M. D. P. Ananta, I. K. P. T. Pramana, M. P. Prayoga, and G. S. Mahendra, “Sistem Informasi Penerimaan Siswa Baru Pada Agensi Modisa Modelling Berbasis Web,” *Jurnal Teknologi Ilmu Komputer*, vol. 1, no. 1, pp. 16–22, 2022.
- [10] T. Santhi *et al.*, “Pengembangan Sistem Informasi To Do List Berbasis Website dalam Meninjau Kegiatan Mahasiswa Undiksha,” *Jurnal Teknologi Ilmu Komputer*, vol. 1, no. 1, pp. 30–36, 2022.
- [11] G. A. Supriatmaja, I. P. M. Y. Pratama, K. Mahendra, K. D. D. Widayaputra, J. Deva, and G. S. Mahendra, “Sistem Informasi Perpustakaan Menggunakan Framework Bootstrap Dengan PHP Native dan Database MySQL Berbasis Web Pada SMP Negeri 2 Dawan,” *Jurnal Teknologi Ilmu Komputer*, vol. 1, no. 1, pp. 7–15, 2022.
- [12] D. A. Putra *et al.*, “Sistem Informasi Pembayaran SPP Berbasis Website Pada SMK Muhammadiyah 3 Tegaldlimo,” *RESI*, vol. 1, no. 2, pp. 88–95, 2023, doi: 10.32795/resi.v1i2.3637.
- [13] G. S. Mahendra and I. K. A. Asmarajaya, “Evaluation Using Black Box Testing and System Usability Scale in the Kidung Sekar Madya Application,” *SinkrOn*, vol. 7, no. 4, pp. 2292–2302, Oct. 2022, doi: 10.33395/sinkron.v7i4.11755.

- [14] I. G. B. W. Atmaja, K. N. A. Kusuma, A. A. E. Wirayuda, I. K. Widiantara, N. Premadhipa, and G. S. Mahendra, “Penerapan Metode Prototype pada Perancangan Sistem Informasi Pengaduan Masyarakat Buleleng Berbasis Website,” *RESI*, vol. 1, no. 2, pp. 55–65, 2023, doi: 10.32795/resi.v1i2.3553.
- [15] M. Lailiya, N. L. W. S. R. Ginantra, and G. S. Mahendra, “Website-Based Budget Adjustment Information System at PT. Taspen (Persero) Denpasar Branch Office,” *JOMLAI*, vol. 1, no. 1, pp. 31–42, Mar. 2022, doi: 10.55123/jomlai.v1i1.162.
- [16] D. M. D. U. Putra, G. S. Mahendra, and E. Mulyadi, “Sistem Informasi Penerimaan Siswa Baru pada SMP Negeri 3 Cibal Berbasis Web,” *INSERT*, vol. 3, no. 1, pp. 42–52, 2022, doi: 10.23887/insert.v3i1.50513.
- [17] I. W. W. Karsana and G. S. Mahendra, “Sistem Informasi Geografis Pemetaan Lokasi Puskesmas Menggunakan Google Maps API di Kabupaten Badung,” *Jurnal Komputer dan Informatika*, vol. 9, no. 2, pp. 160–167, Oct. 2021, doi: 10.35508/jicon.v9i2.5214.
- [18] G. S. Mahendra and I. K. A. Asmarajaya, “Konservasi Kidung Sekar Madya dalam Aplikasi Berbasis Android Menggunakan Successive Approximation Model,” *JustIn : Jurnal Sistem dan Teknologi Informasi*, vol. 10, no. 4, pp. 542–549, Oct. 2022, doi: 10.26418/justin.v10i4.56806.