

## IMPLEMENTASI *EQUIVALENCE PARTITIONING TESTING* PADA FITUR *BOOKING* DAN *JADWAL WEBSITE* PRAKTIK GIGI MANDIRI drg. Susilawati (<https://frontend-webklinik.vercel.app/>)

Muhammad Nasir<sup>1</sup>, Keysha Maulina Halimi<sup>2</sup>, Tiara Ariyanto Putri<sup>3</sup>, Muhammad Rahmat Maryadi<sup>4</sup>, Rayhan Ananda Hafiz Pradipta<sup>5</sup>, Hassan Nasrallah Matouq<sup>6</sup>, Aditya Wicaksono<sup>7\*</sup>

<sup>1-7</sup>Teknologi Rekayasa Perangkat Lunak, IPB University

m\_nasir@apps.ipb.ac.id<sup>1</sup>, maulinakeysha@apps.ipb.ac.id<sup>2</sup>, 3007tiara@apps.ipb.ac.id<sup>3</sup>, muhammaryadi@apps.ipb.ac.id<sup>4</sup>, rayhan\_ananda@apps.ipb.ac.id<sup>5</sup>, sanhassan@apps.ipb.ac.id<sup>6</sup>, adityawicaksono@apps.ipb.ac.id<sup>7\*</sup>,

*Submitted November 26, 2024; Revised March 4, 2025; Accepted April 4, 2025*

### Abstrak

Perkembangan teknologi informasi dalam bidang kesehatan telah mendorong digitalisasi layanan, termasuk dalam pengelolaan reservasi dan jadwal pada praktik gigi. *Website* Praktik Gigi Mandiri dikembangkan untuk membantu Klinik Gigi drg. Susilawati dalam mengelola pasien secara efisien melalui fitur *Booking* dan *Jadwal*. Penelitian ini bertujuan untuk menguji fungsionalitas kedua fitur tersebut menggunakan metode *Black Box Testing* dengan teknik *Equivalence Partitioning*, yang berfokus pada pengujian fungsi tanpa mempertimbangkan struktur internal sistem. Pengujian dilakukan pada beberapa skenario utama, termasuk melihat jadwal, menambah jadwal, mengedit jadwal, menghapus jadwal, dan melakukan reservasi. Hasil pengujian menunjukkan bahwa fitur *Booking* memiliki tingkat keberhasilan sebesar 50%, sedangkan fitur *Jadwal* mencapai 57,14%. Secara keseluruhan, tingkat keberhasilan gabungan dari kedua fitur tersebut adalah 53,85%, yang menunjukkan bahwa sebagian besar fungsi berjalan sesuai dengan yang diharapkan. Namun, terdapat beberapa skenario yang gagal akibat kurangnya validasi input dan penanganan error yang belum optimal. Keterbatasan pengujian ini terletak pada fokusnya pada fungsionalitas tanpa melibatkan pengujian kinerja dan keamanan. Oleh karena itu, diperlukan pengujian lanjutan untuk memastikan stabilitas dan keamanan sistem sebelum diimplementasikan secara penuh. Secara keseluruhan, *Website* Praktik Gigi Mandiri sudah mampu menjalankan fungsi utamanya dengan baik, namun diperlukan perbaikan lebih lanjut untuk meningkatkan keandalan dan pengalaman pengguna.

**Kata Kunci :** *Black Box Testing, Equivalence Partitioning, Website* Praktik Gigi, Pengujian Fungsional

### Abstract

*The advancement of information technology in the healthcare sector has driven the digitalization of services, including the management of reservations and schedules in dental practices. The Praktik Gigi Mandiri Website was developed to assist the dental clinic of drg. Susilawati in efficiently managing patient appointments through the Booking and Scheduling features. This study aims to test the functionality of these features using the Black Box Testing method with the Equivalence Partitioning technique, focusing on functional Testing without considering the internal structure of the system. Testing was conducted on several main scenarios, including viewing schedules, adding schedules, editing schedules, deleting schedules, and making reservations. The test results indicate that the Booking feature achieved a success rate of 50%, while the Scheduling feature reached 57.14%. Overall, the combined success rate of both features is 53.85%, showing that most functions operated as expected. However, some scenarios failed due to insufficient input validation and inadequate error handling. The limitation of this Testing lies in its focus on functionality without involving performance and security Testing. Therefore, further Testing is needed to ensure system stability and security before full implementation. Overall, the Praktik Gigi Mandiri Website has successfully performed its main functions, but further improvements are necessary to enhance system reliability and user experience.*

**Keywords :** *Black Box Testing, Equivalence Partitioning, Dental Practice Website, Functional Testing*

## 1. PENDAHULUAN

Dalam pembuatan suatu *website*, tahap akhir yang sangat penting untuk dilakukan sebelum akhirnya *website* dipublikasikan adalah tahap pengujian[1]. *Software Testing* atau pengujian perangkat lunak adalah sebuah proses untuk menemukan *bug* atau kendala yang terdapat pada aplikasi/*website*/sistem yang telah dikembangkan oleh pengembang dan memastikan keseluruhan fungsi sudah berfungsi dengan baik[2]. Melalui pengujian, penguji dapat menemukan masalah yang dimiliki oleh *website* dan dapat memberikan saran perbaikan serta meningkatkan keamanan pada sebuah *website*[3]. Dalam pengujian perangkat lunak, terdapat beberapa metode yang dapat digunakan, yaitu metode *White Box* dan *Black Box*[4]. Pengujian *Black Box*, atau dikenal sebagai pengujian kotak hitam, adalah metode pengujian yang menitikberatkan pada evaluasi fungsi perangkat lunak berdasarkan persyaratan yang telah ditetapkan, tanpa mempertimbangkan struktur internal atau cara kerja internal aplikasi tersebut[5]. Fokus pada pengujian dengan teknik metode ini adalah dari segi tampilan, respon, fungsionalitas, dan alur dari sistem/aplikasi/*website* yang disesuaikan dengan kebutuhan pengguna [6].

Salah satu teknik dari metode *Black Box Testing* adalah *Equivalence Partitioning*[7]. *Equivalence Partitioning* merupakan teknik pengujian dengan kondisi penguji untuk meningkatkan efisiensi dan efektivitas pengujian perangkat lunak dengan membagi input menjadi kelas-kelas data guna mengurangi jumlah kasus uji tanpa mengorbankan kualitas sistem. Penerapannya membantu mengidentifikasi dan memperbaiki fungsionalitas yang gagal, meningkatkan keandalan sistem, serta memastikan kualitas[8], [9], [10]. Teknik ini membantu mengidentifikasi *bug* secara lebih efektif dan efisien[11].

Perkembangan teknologi informasi di bidang kesehatan meningkatkan layanan pasien. Klinik Gigi drg. Susilawati memanfaatkan sistem berbasis web untuk mengelola jadwal dan praktik secara lebih efisien, terutama dalam administrasi seperti pengelolaan data pasien dan otomatisasi reservasi[12]. *Website* ini menyediakan fitur melihat, menambah, mengedit, dan menghapus jadwal, layanan, serta harga guna mempercepat proses administrasi bagi pasien dan asisten dokter[13]. Pengujian diperlukan untuk memastikan fitur berjalan sesuai kebutuhan pengguna[14]. Teknologi ini juga mempermudah akses catatan medis dan mendukung pengambilan keputusan di titik perawatan[15]. serta mempercepat pendaftaran pasien, pengelolaan jadwal, dan pencatatan rekam medis[16]. Integrasi teknologi informasi dalam layanan kesehatan berperan penting dalam meningkatkan kualitas dan efisiensi operasional.

*Website* Klinik Gigi drg. Susilawati yang diberi nama *Website* Praktik Gigi Mandiri menyediakan fitur *Booking* dan Jadwal, yang dapat membantu pasien dalam mengatur janji temu. Fitur-fitur ini sangat penting untuk memberikan pengalaman pengguna yang baik, meningkatkan efisiensi layanan, dan mendukung operasional klinik secara keseluruhan[17]. *Website* Praktik Gigi Mandiri drg. Susilawati memiliki beberapa fitur, namun fitur *Booking* dan Jadwal dipilih sebagai fokus pengujian karena kedua fitur ini memiliki peran yang sangat krusial dalam operasional klinik[18]. Namun fitur *Booking* dan Jadwal dipilih sebagai fokus pengujian dimana kedua fitur ini mempunyai peran yang sangat krusial di dalam operasional klinik. Fitur *Booking* memungkinkan pasien melakukan reservasi janji temu, sedangkan fitur Jadwal digunakan oleh staf klinik untuk mengatur ketersediaan dokter dan layanan. Kendala pada kedua fitur ini dapat memengaruhi

kualitas layanan kesehatan, seperti kesalahan dalam penjadwalan, kelebihan pemesanan (*overbooking*), atau hilangnya data reservasi. Oleh karena itu, memastikan kedua fitur ini beroperasi dengan lancar dan tanpa kesalahan menjadi aspek utama dalam pengujian guna menjamin pengalaman pengguna yang optimal serta mendukung efisiensi operasional klinik[19], [20].

Beberapa penelitian telah menerapkan teknik *Equivalence Partitioning* dalam berbagai konteks. Iskhak dan Rizkika (2021) menguji kualitas RESTful API pada Sistem Informasi Klinik Pratama UPN "Veteran" Yogyakarta[21][22]. Suyudi et al. (2020) menerapkan teknik ini untuk menguji sistem pendaftaran PAUD berbasis web[12]. Rosmiati (2021) menggunakannya dalam pengujian Sistem Informasi Inventarisasi Barang di Madrasah Tsanawiyah Negeri 2 Palangka Raya[13]. Pramudita (2020) menerapkannya pada aplikasi Ecampus[23]. Azizah et al. (2024) menguji sistem untuk pemilik usaha F&B[4]. sementara Hardika et al. (2024) menggunakannya pada website Garuda Farm[5]. Maulidya et al. (2024) memastikan kualitas Sistem Pencatatan Pesanan Berbasis Website [24]. dan Nadhifah et al. (2024) menilai fungsionalitas situs web Wingpos dengan metode Black Box Testing yang membandingkan output aktual dengan hasil yang diharapkan berdasarkan input yang diberikan[25].

Meskipun penelitian-penelitian di atas telah menerapkan teknik *Equivalence Partitioning* dalam berbagai sistem, penelitian ini memiliki fokus yang spesifik pada pengujian fitur *Booking* dan *Jadwal* pada *Website* Praktik Gigi Mandiri drg. Susilawati. Selain itu, penelitian ini tidak hanya mengevaluasi RESTful API atau sistem backend, tetapi juga menguji antarmuka pengguna serta interaksi

langsung yang dilakukan oleh pasien dan staf klinik.

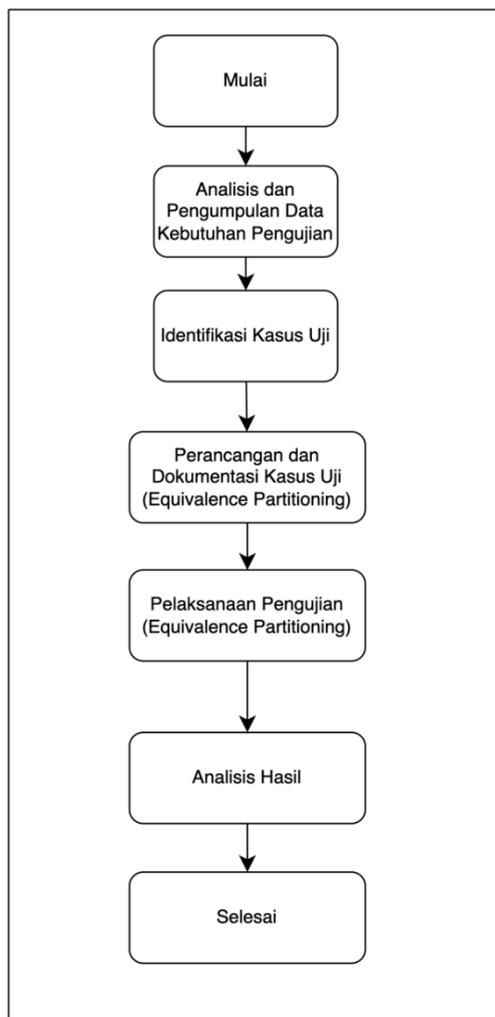
Penelitian ini bertujuan untuk melakukan pengujian dengan metode *Black Box Testing* khususnya dengan teknik *Equivalence Partitioning* dan menemukan kesalahan yang terdapat pada *Website* Praktik Gigi Mandiri, terutama pada fitur *Booking* dan *Jadwal*. Pengujian ini diharapkan dapat mendukung tim pengembang dalam mengidentifikasi dan memperbaiki kesalahan yang ditemukan, sehingga sistem dapat disempurnakan sebelum digunakan oleh para pengguna[26].

## 2. METODE PENELITIAN

### Tahapan Penelitian

Penelitian ini diawali dengan menganalisis kebutuhan pengujian dan pengumpulan data terkait fitur utama *Website* Praktik Gigi Mandiri, terutama fitur *booking* dan *jadwal*. Setelah data dikumpulkan, kasus uji diidentifikasi untuk memenuhi kebutuhan fungsional *website*. Setelah itu, perancangan dan dokumentasi kasus uji dilakukan dengan teknik *Equivalence Partitioning*. Akhir dari penelitian ini adalah pelaksanaan pengujian dan analisis hasil untuk menemukan dan mencatat kesalahan serta memberikan saran untuk perbaikan *website* di masa mendatang.

Untuk memberikan gambaran yang lebih jelas, alur penelitian ini dapat dilihat pada diagram alir (*flowchart*) berikut:



Gambar 1. Tahapan Penelitian

### Black Box Testing

Penelitian ini menggunakan metode *Black Box Testing* untuk mengevaluasi fungsionalitas *website* tanpa mempertimbangkan struktur internal program[5]. Fokus utama pada pengujian ini adalah memastikan bahwa fitur *website* memenuhi kebutuhan pengguna. Untuk memastikan kualitas dan kemudahan penggunaan, pendekatan ini dilakukan dari sudut pandang pengguna dengan memperhatikan elemen alur kerja, antarmuka, dan respons *website*.

Dalam *Black Box Testing*, terdapat beberapa teknik yang dapat digunakan, seperti *Boundary Value Analysis*, *Decision Table Testing*, dan *Equivalence Partitioning*. Dibandingkan dengan teknik

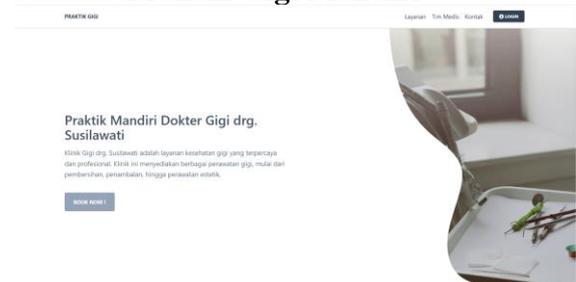
lain, *Equivalence Partitioning* dipilih karena lebih efektif dalam mengelompokkan input dan mengurangi jumlah kasus uji tanpa mengurangi kualitas pengujian[21].

### Equivalence Partitioning

Teknik *Equivalence Partitioning* yang dilakukan pada pengujian ini dilakukan dengan cara membagi dua inputan menjadi input valid dan input invalid, sehingga mengurangi jumlah kasus uji yang diperlukan, tetapi tetap menghasilkan hasil yang lebih baik[6], [25]. Penggunaan teknik ini dipilih karena lebih berfokus pada kebutuhan fungsional *website*, sehingga penguji dapat mendefinisikan *Test Case* dan mengevaluasi fungsionalitas *website* [9].

Berdasarkan studi literatur, *Equivalence Partitioning* memiliki kelebihan dalam hal efektivitas dan efisiensi pengujian dibandingkan dengan *Boundary Value Analysis* dan *Decision Table Testing*. Teknik ini mampu mengurangi duplikasi pengujian dengan mengelompokkan input yang memiliki perilaku serupa[12], [13].

### Website Praktik Gigi Mandiri

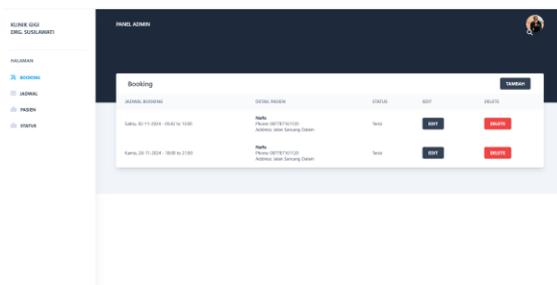


Sumber: Hasil Penelitian

### Gambar 2. Beranda Website Praktik Gigi Mandiri

*Website* Praktik Gigi Mandiri adalah sebuah *website* yang bertujuan untuk membantu praktik gigi drg. Susilawati dalam mengelola reservasi dan jadwal pasien, layanan, serta harga secara otomatis. Sistem ini memungkinkan pasien untuk melakukan reservasi langsung melalui antarmuka web, serta menyederhanakan manajemen data

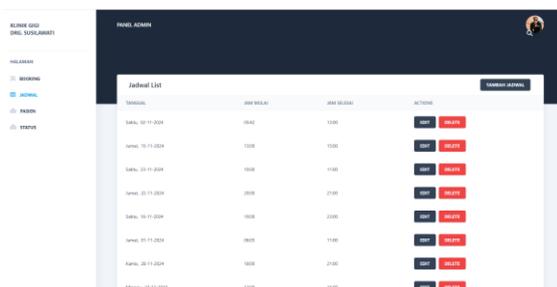
melalui input otomatis ke dalam spreadsheet. Dengan berbagai fitur seperti melihat, menambah, mengedit, dan menghapus jadwal, layanan, serta harga, *website* ini diharapkan mampu mempercepat dan mempermudah proses administrasi klinik, baik bagi pasien maupun asisten dokter. *Website* ini terbagi menjadi dua, yaitu bagi pasien dan bagi admin. Pada *website* yang dipergunakan untuk admin, terdapat empat fitur utama seperti *Booking*, *Jadwal*, *Pasien*, dan *Status*.



Sumber: Hasil Penelitian

**Gambar 3. Tampilan fitur *Booking***

Gambar di atas merupakan tampilan saat admin pergi ke fitur *Booking*. Dalam fitur ini, admin dapat melakukan lihat, tambah, edit, dan hapus data pasien yang telah melakukan reservasi. Pada tampilan fitur ini juga ditampilkan data pasien berupa jadwal kunjungan, detail pasien yang meliputi nama, nomor telepon, dan alamat, serta status reservasi pasien.



Sumber: Hasil Penelitian

**Gambar 4. Tampilan fitur *Jadwal***

Gambar di atas merupakan tampilan dari fitur *Jadwal* yang ada pada *Website* Praktik Gigi Mandiri. Pada fitur ini, admin dapat melakukan lihat, tambah, edit, dan hapus jadwal pasien. Dalam fitur ini ditampilkan

jadwal list seperti tanggal, jam mulai, dan jam selesai.

### Rencana Pengujian

Pengujian *Website* Praktik Gigi Mandiri dilaksanakan pada 22 November 2024, dengan tim penguji beranggotakan lima orang. Fitur yang akan diuji pada *website* ini adalah fitur *Booking* dan *Jadwal*. Pengujian dilakukan menggunakan metode Black Box, yang menitikberatkan pada evaluasi fungsi website tanpa mempertimbangkan struktur internal dari aplikasi yang dikembangkan[9]. Teknik *Equivalence Partitioning* dipilih dengan cara membagi hasil menjadi valid dan invalid untuk memastikan bahwa setiap fungsi dapat menangani berbagai kondisi dengan benar [6]. Pada fitur *Jadwal*, akan diuji dengan empat aktivitas yang berbeda, yaitu lihat, tambah, edit, dan hapus, sedangkan pada fitur *Booking*, hanya akan diuji dengan tiga aktivitas, yaitu tambah, edit, dan hapus.

Data uji dipilih berdasarkan skenario nyata yang diambil dari data aktual pada *website*, serta skenario hipotetis yang dirancang untuk mengevaluasi batasan dan kemungkinan kesalahan pada *website*. Lingkungan pengujian menggunakan perangkat keras berupa komputer dengan spesifikasi minimal prosesor Intel Core i5, RAM 8GB, dan koneksi internet stabil. Pengujian dilakukan pada beberapa browser, termasuk Google Chrome dan Mozilla Firefox, untuk memastikan kompatibilitas dan konsistensi tampilan.

Hasil pengujian diharapkan dapat fitur menunjukkan bahwa fungsionalitas *website* sudah berjalan dengan baik dan sesuai harapan. Segala kesalahan dan temuan lainnya yang ada selama pengujian akan didokumentasikan dan disusun secara terperinci untuk memudahkan perbaikan pada *website*.

**Tabel 1. Rencana Pengujian**

ID	Fungsi	Deskripsi Pengujian	Hasil yang diharapkan
PJ01	Lihat Jadwal	Pengujian mengecek apakah pengguna dapat melihat jadwal yang sudah ditetapkan.	Semua jadwal yang tersedia ditampilkan lengkap dengan kolom Tanggal, Jam Mulai, Jam Selesai, dan Actions.
PJ02	Tambah Jadwal	Pengujian menambahkan jadwal baru ke dalam sistem.	Sistem dapat menerima input dan data berhasil disimpan. Serta menampilkan pesan sukses: "Jadwal added successfully!".
PJ03	Tambah Jadwal	Pengujian menambahkan jadwal baru ke dalam sistem dengan skenario gagal.	Sistem gagal menambahkan data jadwal terbaru.
PJ04	Edit Jadwal	Pengujian mengubah jadwal yang sudah ada di sistem.	Jadwal berhasil diubah dengan isi yang sesuai dengan apa yang diinginkan dan muncul pesan atau popup bahwa jadwal sudah berhasil disimpan.
PJ05	Edit Jadwal	Pengujian mengubah jadwal yang sudah ada di sistem dengan skenario gagal.	Jadwal gagal diubah dan masih menampilkan rincian jadwal sebelum dilakukan perubahan.
PJ06	Hapus Jadwal	Pengujian menghapus jadwal tertentu pada sistem.	Muncul pesan konfirmasi dan user memilih "Ok", sehingga jadwal berhasil terhapus dan data berhasil di-update.
PJ07	Hapus Jadwal	Pengujian menghapus jadwal tertentu pada sistem dengan skenario gagal	Jadwal gagal terhapus dari sistem dan masih ada dalam daftar jadwal.

PB01	Tambah <i>Booking</i>	Pengujian tambah <i>booking</i> berjalan dengan lancar atau tidak.	Menambah <i>booking</i> dengan benar dan muncul alert bahwa berhasil
PB02	Tambah <i>Booking</i>	Pengujian tambah <i>booking</i> berjalan dengan lancar atau tidak dengan skenario gagal	Data <i>booking</i> terbaru gagal ditambahkan.
PB03	Edit <i>Booking</i>	Pengujian melakukan edit <i>booking</i> .	Saat tombol "Simpan" ditekan akan muncul pesan "Booking updated successfully!" dan semua data berhasil diubah dengan benar.
PB04	Edit <i>Booking</i>	Pengujian melakukan edit <i>booking</i> dengan skenario gagal.	Data <i>booking</i> gagal diubah dan masih menampilkan data sebelumnya.
PB05	Hapus <i>Booking</i>	Pengujian menghapus jadwal <i>booking</i> yang sudah ada di sistem	Jadwal <i>booking</i> berhasil dihapus dan akan me- <i>refresh</i> ulang <i>booking</i> . Muncul pesan atau popup "Booking has been deleted".
PB06	Hapus <i>Booking</i>	Pengujian menghapus jadwal <i>booking</i> yang sudah ada di sistem dengan skenario gagal.	Sistem gagal menghapus <i>booking</i> dan masih ditampilkan pada halaman <i>Booking</i> .

Sumber: Hasil Penelitian

### 3. HASIL DAN PEMBAHASAN

#### Tahap Pengujian Sistem

Pengujian yang dilakukan pada *Website* Praktik Gigi Mandiri, khususnya pada fitur *Booking* dan Jadwal bertujuan untuk mengidentifikasi kesalahan sebelum digunakan oleh pengguna. Setelah menyusun rancangan pengujian dan melaksanakan proses pengujian, hasil yang diperoleh dicatat dalam formulir pengujian.

Selanjutnya, hasil tersebut dianalisis dan disimpulkan berdasarkan evaluasi terhadap kesesuaian antara hasil yang diperoleh dengan hasil yang diharapkan. Jika hasilnya sesuai, maka disimpulkan sebagai valid, sedangkan jika tidak sesuai, akan dinyatakan tidak valid. Jika ditemukan kesalahan saat pengujian maka akan dilakukan perbaikan pada bagian yang terdeteksi memiliki masalah. Berikut ini adalah tabel yang memuat kesimpulan dari hasil pengujian yang telah dilakukan berdasarkan rancangan rencana pengujian:

**Tabel 2. Hasil Pengujian Fitur Jadwal**

ID	Fungsi	Test Case	Hasil yang diharapkan	Hasil
PJ01	Lihat Jadwal	Masuk ke halaman "Jadwal".	Website menampilkan halaman Jadwal dan konten pada halaman jadwal yang tersedia ditampilkan lengkap dengan kolom Tanggal, Jam Mulai, Jam Selesai, dan Actions.	Valid
PJ02	Tambah Jadwal	<b>Tanggal:</b> 11/23/2024 <b>Jam mulai:</b> 07:00 AM. <b>Jam selesai:</b> 08:00 AM	Website dapat menerima input dan data berhasil disimpan. Serta menampilkan pesan sukses: "Jadwal added successfully!".	Valid
PJ03	Tambah Jadwal	<b>Tanggal:</b> 11/23/2024 <b>Jam mulai:</b> 07:00 AM. <b>Jam selesai:</b>	Website tidak dapat menambahkan data terbaru dan menampilkan pesan error "Data Jadwal Gagal ditambahkan"	Invalid

PJ04	Edit Jadwal	08:00 AM <b>Tanggal:</b> 11/24/2024	Website berhasil melakukan perubahan jadwal disertai dengan tanggal yang berhasil diubah sesuai dengan yang disunting.	Valid
PJ05	Edit Jadwal	<b>Tanggal:</b> 11/24/2024	Website gagal melakukan perubahan pada jadwal yang disertai tanggal gagal diubah, menampilkan pesan "Gagal mengedit Jadwal" serta jadwal masih menampilkan tanggal yang sama	Invalid
PJ06	Hapus Jadwal	Hapus jadwal teratas	Muncul pesan konfirmasi dan user memilih "Ok", sehingga jadwal berhasil terhapus dan data berhasil di-update.	Valid
PJ07	Hapus Jadwal	Hapus jadwal teratas	Muncul pesan konfirmasi dan user memilih "Ok", tetapi jadwal gagal terhapus dan masih ada dalam daftar jadwal.	Invalid

Sumber: Hasil Penelitian

Berdasarkan hasil pengujian fitur Jadwal pada tabel di atas, beberapa fungsi seperti Lihat Jadwal dan beberapa skenario pada Tambah Jadwal, Edit Jadwal, serta Hapus Jadwal berjalan sesuai harapan. Baik dari skenario berhasil ataupun skenario gagal, hasil pengujian pada fitur ini menunjukkan bahwa sistem telah dirancang dengan cara yang tepat, sehingga dapat menjaga data tetap konsisten dan mencegah kesalahan dalam operasional.

**Tabel 3. Hasil Pengujian Booking**

ID	Fungsi	Test Case	Hasil yang diharapkan	Hasil
PB01	Tambah Booking	<b>Nama pasien:</b> Rayhan <b>Jadwal pasien:</b> 13.00-15.00 <b>Status pasien:</b> Terisi	<i>Website</i> berhasil melakukan penambahan data <i>Booking</i> dengan benar dan muncul pesan bahwa penambahan data <i>booking</i> berhasil	Valid
PB02	Tambah Booking	<b>Nama pasien:</b> Rayhan <b>Jadwal pasien:</b> 13.00-15.00 <b>Status pasien:</b> Terisi	<i>Website</i> gagal melakukan penambahan data <i>booking</i> dan menampilkan pesan error berupa "Gagal Menambahkan data <i>booking</i> "	Invalid
PB03	Edit Booking	<b>Nama pasien:</b> Rayhan <b>Jadwal pasien:</b> 14.00-16.00 <b>Status pasien:</b> Terisi	Saat tombol "Simpan" ditekan akan muncul pesan "Booking updated successfully!" dan semua data berhasil diubah	Valid

		<b>pasien:</b> Terisi	dengan benar.	
PB 04	Edit Booking	<b>Nama pasien:</b> Rayhan <b>Jadwal pasien:</b> 13.00-15.00 <b>Status pasien:</b> Terisi	Saat tombol "Simpan" ditekan akan muncul pesan "Booking updated successfully!" dan data yang lain tetap sama seperti sebelumnya.	Invalid
PB05	Hapus Booking	Hapus <i>Booking</i> teratas	<i>Website</i> berhasil melakukan penghapusan <i>booking</i> disertai pesan atau popup "Booking has been deleted", <i>website</i> akan Kembali ke halaman <i>booking</i> .	Valid
PB06	Hapus Booking	Hapus <i>Booking</i> teratas	Jadwal <i>booking</i> gagal dihapus dan masih ada dalam daftar.	Invalid

Sumber: Hasil Penelitian

Dari hasil pengujian yang telah dilakukan dapat disimpulkan bahwa fitur *Booking* dan Jadwal telah berjalan sesuai dengan hasil yang diharapkan, tetapi diperlukan tindakan tambahan untuk memastikan bahwa setiap kesalahan yang mungkin akan terjadi telah diantisipasi. Langkah-langkah ini diharapkan dapat meningkatkan kualitas sistem secara keseluruhan, sehingga dapat digunakan sepenuhnya untuk operasional klinik.

### Perhitungan Persentase Keberhasilan

Setelah melakukan pengujian terhadap dua fitur utama, perhitungan persentase keberhasilan akan dihitung menggunakan rumus berikut:

$$\text{Persentase Keberhasilan} = (\text{Jumlah Test Case Valid} / \text{Jumlah Test Case Total}) \times 100\%$$

Dari rumus yang ada, selanjutnya rumus diterapkan pada perhitungan persentase keberhasilan dari masing-masing fitur pengujian, yaitu *Booking* dan *Jadwal*. Berikut merupakan hasil dari perhitungan persentase keberhasilan pada fitur *Jadwal*:

$$\text{Persentase Keberhasilan Fitur Jadwal} = (4/7) \times 100\% = 57.14\%$$

Hasil fitur menunjukkan bahwa persentase keberhasilan dari pengujian pada fitur *Jadwal* adalah mencapai **57.14%**. Hal ini menandakan bahwa sebagian besar fungsi pada fitur *Jadwal* sudah berjalan sesuai harapan. Dengan demikian, diperlukan langkah evaluasi dan revisi pada sistem untuk meningkatkan tingkat keberhasilan hingga mendekati 100%.

Berikutnya adalah perhitungan pada fitur *Booking*. Berikut adalah hasil perhitungan persentase keberhasilan pada fitur *Booking*:

$$\text{Persentase Keberhasilan Fitur Booking} = (3/6) \times 100\% = 50\%$$

Hasil fitur menunjukkan bahwa persentase keberhasilan dari pengujian pada fitur *Booking* adalah mencapai **50%**. Hal ini menandakan bahwa hanya setengah dari fungsi yang diuji pada fitur *Booking* berjalan sesuai dengan yang diharapkan. Persentase ini fitur menunjukkan adanya potensi peningkatan pada fitur *Booking*, baik dari segi penanganan skenario *valid* maupun *invalid*, untuk memastikan sistem dapat memenuhi kebutuhan pengguna secara menyeluruh. Evaluasi lebih

mendalam diperlukan agar fitur ini dapat berfungsi secara optimal.

Setelah mendapatkan hasil perhitungan persentase keberhasilan dari masing-masing fitur, berikutnya adalah total persentase keseluruhan dari kedua fitur. Hasil persentase keseluruhan dari pengujian pada kedua fitur ini didapatkan sebagai berikut:

$$\text{Persentase Keberhasilan Keseluruhan} = (13/7) \times 100\% = 53.85\%$$

Persentase keberhasilan keseluruhan dari pengujian untuk fitur *Booking* dan *Jadwal* adalah **53.85%**.

Analisis error dilakukan untuk memahami penyebab kegagalan fitur *Booking* dan *Jadwal*. Masalah utama meliputi validasi input yang lemah, ketidakcocokan data dengan *database*, serta kurangnya notifikasi informatif bagi pengguna. Validasi input yang tidak memadai menyebabkan data tidak sesuai format atau kosong tetap diproses, sementara ketidaksesuaian data mengakibatkan duplikasi atau konflik dalam *database*. Kurangnya notifikasi error membuat pengguna kesulitan memahami penyebab kegagalan transaksi.

Solusi teknis yang dapat diterapkan mencakup penguatan validasi input dengan regex, pengecekan duplikasi sebelum penyimpanan, serta validasi hubungan antar data. Selain itu, peningkatan *user experience* dilakukan dengan menambahkan notifikasi yang lebih jelas dan ramah pengguna agar mereka dapat memahami serta mengatasi error dengan mudah.

Jika tidak segera diperbaiki, error ini dapat menyebabkan kebingungan, frustrasi, kehilangan data, dan menurunnya kepercayaan pengguna terhadap sistem. Akibatnya, kredibilitas *website* Praktik Gigi Mandiri dapat terpengaruh. Oleh karena itu,

perbaikan diperlukan agar sistem lebih andal dan memberikan pengalaman pengguna yang lebih baik.

#### 4. SIMPULAN

Pengujian Website Praktik Gigi Mandiri berfokus pada fitur Booking dan Jadwal dengan tingkat keberhasilan masing-masing 50% dan 57,14%. Meskipun sebagian besar fungsi berjalan baik, perbaikan validasi input dan penanganan error diperlukan untuk meningkatkan keandalan sistem. Pengujian ini terbatas pada aspek fungsional tanpa mencakup kinerja dan keamanan, sehingga diperlukan pengujian lanjutan sebelum implementasi penuh. Secara keseluruhan, sistem sudah cukup baik, tetapi perbaikan lebih lanjut akan meningkatkan kualitas dan stabilitasnya.

#### DAFTAR PUSTAKA

- [1] R. Pressman and B. Maxim, *Software Engineering: A Practitioner's Approach*, 9th ed. McGraw-Hill Education, 2020.
- [2] Ian. Sommerville, *Software engineering*. Pearson, 2016.
- [3] A. Zirwan, "Pengujian dan Analisis Keamanan Website Menggunakan Acunetix Vulnerability Scanner," *Jurnal Informasi dan Teknologi*, pp. 70–75, Mar. 2022, doi: 10.37034/jidt.v4i1.190.
- [4] D. N. Azizah *et al.*, "Analysis and Testing of the Combox Web Application System Using Black Box Testing with the Equivalence Partitioning Method," *International Journal of Electrical Engineering, Mathematics and Computer Science*, vol. 1, no. 4, pp. 37–43, Nov. 2024, doi: 10.62951/ijeemcs.v1i4.118.
- [5] B. Hardika *et al.*, "Pengujian Blackbox Testing Website Garuda Farm Menggunakan Teknik Equivalence Partitioning," *JURNAL KRIDATAMA SAINS DAN TEKNOLOGI*, vol. 6, no. 02, pp. 740–753, Dec. 2024, doi: 10.53863/kst.v6i02.1420.
- [6] P. Ammann and J. Offutt, *Introduction to Software Testing*. Cambridge University Press, 2016. doi: 10.1017/9781316771273.
- [7] P. C. Jorgensen, "Software Testing Fourth Edition A Craftsman's Approach".
- [8] B. B. Sasongko, F. Malik, F. Ardiansyah, A. F. Rahmawati, F. D. Adhinata, and D. P. Rakhmadani, "Pengujian Blackbox Menggunakan Teknik Equivalence Partitions pada Aplikasi Petgram Mobile," *Journal ICTEE*, vol. 2, no. 1, p. 10, Mar. 2021, doi: 10.33365/jictee.v2i1.1012.
- [9] A. Agustian, I. Andryani, S. Khoerunisa, A. Pangestu, and A. Saifudin, "Implementasi Teknik Equivalence Partitioning pada Pengujian Aplikasi E-learning Berbasis Web," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 3, no. 3, p. 178, Aug. 2020, doi: 10.32493/jtsi.v3i3.5371.
- [10] A. Krismadi *et al.*, "Jurnal Teknologi Sistem Informasi dan Aplikasi Pengujian Black Box berbasis Equivalence Partitions pada Aplikasi Seleksi Promosi Kenaikan Jabatan," vol. 2, no. 4, pp. 2654–4229, 2019, [Online]. Available: <http://openjournal.unpam.ac.id/index.php/JTSSI/index>
- [11] E. S. J. Atmadji, I. R. Sanjaya, and H. A. Putranto, "Pemanfaatan boundary value analysis dan equivalence partitioning pada automated testing aplikasi berbasis website," *Angkasa: Jurnal Ilmiah Bidang Teknologi*, vol. 15, no. 1, p. 97, May 2023, doi: 10.28989/angkasa.v15i1.1645.
- [12] M. Y. Suyudi, A. P. Pratiwi, R. F. Mawahdah, Y. A. Purwara, and I. Kusyadi, "Teknik Pengujian Equivalent Partitions pada Aplikasi Sistem Pendaftaran PAUD berbasis

- WEB dengan Menggunakan Black Box,” *Jurnal Informatika Universitas Pamulang*, vol. 5, no. 2, p. 198, Jun. 2020, doi: 10.32493/informatika.v5i2.5351.
- [13] Rosmiati, “Analisis Dan Pengujian Sistem Menggunakan Black Box Testing Equivalence Partitioning,” *Jurnal Sains Komputer dan Teknologi Informasi*, vol. 3, no. 2, pp. 56–63, May 2021, doi: 10.33084/jsakti.v3i2.1932.
- [14] R. Pramudita, “Pengujian Black Box pada Aplikasi Ecampus Menggunakan Metode Equivalence Partitioning,” *INFORMATICS FOR EDUCATORS AND PROFESSIONAL: Journal of Informatics*, vol. 4, no. 2, p. 193, Jun. 2020, doi: 10.51211/itbi.v4i2.1347.
- [15] D. D. Andreanto and A. N. Handayani, “Pelayanan Kesehatan Masyarakat Melalui Pemanfaatan Teknologi Digital Society 5.0,” *Jurnal Inovasi Teknologi dan Edukasi Teknik*, vol. 2, no. 5, pp. 220–223, May 2022, doi: 10.17977/um068v2i52022p220-223.
- [16] Y. Budiono and H. B. Santoso, “Sistem Informasi Klinik Gigi Berbasis Web Studi Kasus: Lotus Dental Care,” *Jurnal EKSIS*, vol. 09, no. 1, May 2016.
- [17] V. Garousi, M. Felderer, and M. V. Mäntylä, “The need for multivocal literature reviews in software engineering,” in *Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering*, New York, NY, USA: ACM, Jun. 2016, pp. 1–6. doi: 10.1145/2915970.2916008.
- [18] Y. Singh and S. Saha, “Improving Effectiveness of Equivalence Partitioning Using Combinatorial Testing,” *Journal of Software Engineering and Applications*, 2010.
- [19] D. Mardiyati and Y. Saputra, “Implementasi Sistem Informasi Manajemen Klinik Menggunakan Metode Black Box Testing,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 1, Jan. 2025, doi: 10.23960/jitet.v13i1.6015.
- [20] H. Susilo, N. Abdillah, M. Ihksan, H. D. Morika, and I. Y. Darma, “Analisa Dan Perancangan Sistem Informasi Booking Antrian Pelayan Pada Klinik Medika Saintika Berbasis Website,” *Jurnal Kesehatan Medika Saintika*, vol. 14, no. 1, Jun. 2023.
- [21] M. Ishkak and S. Rizkika, “Implementasi Metode Pengujian Equivalence Partitioning pada Pengembangan RESTful API Sistem Informasi Klinik Pratama UPN,” *Telematika: Jurnal Informatika dan Teknologi Informasi*, 2021.
- [22] H. Zhu, P. A. V. Hall, and J. H. R. May, “Software Unit Test Coverage and Adequacy,” *ACM Comput Surv*, vol. 29, no. 4, pp. 366–427, Dec. 1997, doi: 10.1145/267580.267590.
- [23] L. Copeland, *A Practitioner’s Guide to Software Test Design*. Boston: Artech House, 2004.
- [24] I. Maulidya *et al.*, “154-166 Equivalence Partitioning,” *Bina Insani ICT Journal*, vol. 11, no. 2, pp. 154–166, 2024.
- [25] J. Nadhifah *et al.*, “Black Box Testing on the Wingpos Website Using the Equivalence Partitioning Technique,” *International Journal of Information Engineering and Science*, vol. 1, no. 4, pp. 81–88, Nov. 2024, doi: 10.62951/ijies.v1i4.128.
- [26] B. M. Haqqoni, I. Winarno, M. N. Musthofa, M. Sakdi, and A. Saifudin, “Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode Blackbox Testing Bagi Pemula,” *LOGIC: Jurnal Ilmu Komputer dan Pendidikan*, vol. 2, no. 4, 2024, [Online]. Available: <https://journal.mediapublikasi.id/index.php/logic>