



Implementasi Aplikasi Berbasis Web pada Apotek Rostie

Novinaldi¹, Erien Nada Azandra², Yanni Suherman³, Yolanda Ssa'adah⁴

^{1,2,4}Sistem Informasi, STMIK Jaya Nusa

³Manajemen Informatika, AMIK Jaya Nusa

¹novinaldi1309@gmail.com, ²eriennadaazandra@gmail.com, ³yanishu68@gmail.com, ⁴yolandassaadah03@gmail.com

Abstract

Rostie Pharmacy is a healthcare business unit that still uses a manual system to manage its operational data, such as recording incoming medicines (stock) and outgoing medicines (sales). This manual recording system has several weaknesses, including vulnerability to recording errors, time-consuming processes, and difficulties in searching and generating reports. These issues affect the speed of stock monitoring and the accuracy of available data. The data collection methods used include field research, library research, and laboratory research. The system is designed using Unified Modeling Language (UML), developed with the Waterfall method, and implemented using CodeIgniter 4 (CI4), PHP, and MySQL as the database. The result of this research is an integrated information system that improves operational efficiency, minimizes recording errors, and speeds up access to information at Rostie Pharmacy. It is expected that this system will support the work processes of pharmacy staff, ease decision-making, and enhance the overall quality of medicine management.

Keywords: Information System, Pharmacy, Incoming Medicines, Outgoing Medicines, CI4, PHP, MySQL.

Abstrak

Apotek Rostie merupakan salah satu unit usaha dibidang kesehatan yang masih menggunakan sistem manual dalam pengelolaan data operasionalnya, seperti pencatatan obat masuk (stok) dan obat keluar (penjualan). Sistem pencatatan manual tersebut memiliki berbagai kelemahan, diantaranya rentan terhadap kesalahan pencatatan, membutuhkan waktu lama, serta menyulitkan proses pencarian dan pembuatan laporan. Hal ini berdampak pada lambatnya proses pemantauan stok obat serta kurang akuratnya data yang tersedia. Metode pengumpulan data yang digunakan meliputi penelitian lapangan (*field research*), penelitian perpustakaan (*library research*), dan penelitian laboratorium (*laboratory research*). Sistem dirancang menggunakan *Unified Modeling Language* (UML), dikembangkan dengan metode *Waterfall*, serta diimplementasikan menggunakan *CodeIgniter 4* (CI4), PHP, dan MySQL sebagai basis data. Hasil dari penelitian ini adalah sistem informasi terintegrasi yang mampu meningkatkan efisiensi operasional, meminimalkan kesalahan pencatatan, dan mempercepat akses informasi di Apotek Rostie. Diharapkan sistem ini dapat menunjang proses kerja pengelola apotek, memudahkan dalam pengambilan keputusan, serta meningkatkan kualitas pengelolaan obat secara keseluruhan.

Kata Kunci: Sistem Informasi, Apotek, Obat Masuk, Obat Keluar, CI4, PHP, MySQL.

© 2026 Author
Creative Commons Attribution 4.0 International License



1. Pendahuluan

Perkembangan teknologi informasi (TI) yang pesat telah membawa perubahan signifikan diberbagai sektor kehidupan, termasuk dibidang kesehatan. Penerapan sistem informasi kini menjadi salah satu elemen penting dalam meningkatkan efisiensi operasional, mendukung transparansi, serta meminimalisir kesalahan dalam pengelolaan data.

Sistem informasi merupakan sebuah sistem di dalam suatu organisasi mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, sifatnya manajerial, serta aktivitas strategis dari suatu organisasi serta menyediakan pihak luar tertentu dengan laporan yang dibutuhkan. [1]

Dengan adanya sistem informasi, maka *user* dapat mengetahui informasi-informasi apa saja yang dimiliki oleh *server host* ataupun *database*, dan dapat menggunakan informasi tersebut sesuai dengan kebutuhannya. [2]

Website dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang disediakan melalui jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang di seluruh dunia. [3]

Pada era digital saat ini, sistem informasi berbasis web menjadi salah satu komponen utama dalam pengelolaan berbagai kegiatan di perusahaan atau organisasi [4]. Sistem informasi berbasis web, khususnya, telah terbukti memberikan banyak manfaat, seperti kemudahan akses data secara real-time, pengelolaan data yang terintegrasi, dan kemampuan menghasilkan laporan secara otomatis. Dengan sistem ini, organisasi atau perusahaan dapat menjalankan operasionalnya dengan lebih cepat, tepat, dan efisien.

Namun, di tengah arus digitalisasi yang semakin maju, masih banyak institusi yang bergantung pada sistem manual dalam pengelolaan data mereka. Salah satu contohnya adalah Apotek Rostie di Padang. Hingga saat ini, pengelolaan operasional di Apotek Rostie ini masih dilakukan secara manual, yaitu dengan mencatat data obat masuk (stok obat), obat keluar (penjualan) dalam buku atau arsip fisik. Sistem manual ini memiliki banyak keterbatasan, seperti memakan waktu, rentan terhadap kesalahan manusia (human error), dan sulit diakses jika diperlukan secara cepat.

Masalah lain yang dihadapi adalah lamanya waktu yang dibutuhkan pengelola apotek dalam memantau ketersediaan obat, serta pengambilan data untuk pembuatan laporan penggunaan atau distribusi obat. Proses manual yang dilakukan selama ini menyebabkan kurang akuratnya data sehingga menjadi suatu tantangan tersendiri.

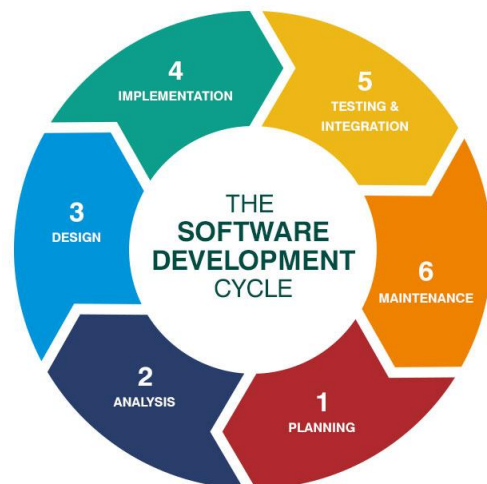
Untuk mengatasi permasalahan tersebut, diperlukan sebuah sistem informasi yang dapat menggantikan metode manual dengan proses yang lebih otomatis dan terstruktur. Sistem informasi berbasis web menawarkan solusi yang relevan untuk kebutuhan ini, dengan kemampuan mencatat, menyimpan, dan mengelola data secara digital. Sistem ini memungkinkan akses yang lebih mudah, cepat, serta penyajian data yang akurat dan dapat langsung diakses oleh pengelola apotek.

Penerapan sistem informasi berbasis web di Apotek Rostie tidak hanya dapat meningkatkan efisiensi operasional, tetapi juga mendukung pengambilan keputusan yang lebih baik dengan menyediakan informasi yang real-time dan terpercaya. Dengan sistem ini, apotek dapat meminimalisir risiko kesalahan, meningkatkan kualitas pelayanan kepada pelanggan, dan memberikan data obat yang lebih akurat dalam pengelolaannya.

Penulis merancang dan mengimplementasikan sistem informasi apotek berbasis web pada Apotek Rostie yang mempermudah pengelola apotek dalam mendata obat masuk dan obat keluar (penjualan) serta pengambilan data obat untuk pembuatan laporan penggunaan atau distribusi obat.

2. Metode Penelitian

Software Development Life Cycle (SDLC) merupakan proses pengembangan atau mengubah suatu sistem perangkat lunak dengan menggunakan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem perangkat lunak. SDLC juga merupakan pola yang diambil untuk mengembangkan sistem perangkat lunak, yang terdiri dari tahap-tahap: rencana (*planning*), analisis (*analysis*), desain (*design*), implementasi (*implementation*), uji coba (*testing*) dan pengelolaan (*maintenance*). Berikut gambaran SDLC :



Gambar 1. Tahapan *Software Development Life Cycle*

Dalam rekayasa perangkat lunak, konsep SDLC mendasari berbagai jenis metodologi pengembangan perangkat lunak. Metodologi-metodologi ini membentuk suatu kerangka kerja untuk perencanaan dan pengendalian pembuatan sistem informasi, yaitu proses pengembangan perangkat lunak.[5]

3. Hasil dan Pembahasan

3.1 Hasil Pengembangan Sistem

Secara teknis, sistem yang dikembangkan memiliki beberapa fitur yang dapat diakses oleh pengguna (user). Dengan adanya fitur tersebut, pengguna dapat:

1. Sistem informasi yang dibangun berupa aplikasi berbasis website yang dapat membantu apoteker dalam melakukan pencatatan data obat secara elektronik dan terstruktur.
3. Sistem yang dikembangkan dapat membantu apoteker memantau manajemen persediaan obat dalam mencatat data obat masuk dari PBF serta mencatat obat keluar untuk penjualan.
4. Aplikasi ini mendukung proses transaksi penjualan yang dilakukan oleh kasir dan secara otomatis mencetak nota serta memperbarui data stok obat.
5. Sistem ini mampu menampilkan informasi stok obat secara real-time dan cepat, termasuk peringatan jika jumlah stok berada di bawah batas minimum.
6. Sistem dapat menghasilkan laporan-laporan penting, seperti laporan penjualan harian, mingguan, bulanan, laporan obat masuk dan keluar, serta jumlah stok yang akurat.
7. Sistem menyediakan fitur login multiuser dengan hak akses yang berbeda sesuai peran pengguna, seperti Apoteker, Kasir, dan Pemilik Sarana Apotek (PSA), untuk menjaga keamanan dan integritas data.

3.2. Pembahasan

Pembahasan ini menguraikan hasil analisis, perancangan, implementasi, serta pengujian sistem informasi apotek berbasis web yang telah dikembangkan untuk Apotek Rostie Padang. Fokus utama pembahasan adalah bagaimana sistem mampu menjawab permasalahan operasional yang sebelumnya berjalan secara manual.

3.2.1 Permasalahan System Manual dan Dampaknya Berdasarkan hasil observasi dan wawancara dengan Apoteker Pengelola Apotek (APA), ditemukan beberapa permasalahan utama dalam operasional Apotek Rostie yang masih menggunakan sistem manual, yaitu:

- a) Pencatatan stok dan penjualan masih menggunakan buku fisik, sehingga rawan

kesalahan pencatatan (human error) dan sering terjadi ketidaksesuaian data stok.

- b) Proses pemantauan ketersediaan obat lambat, karena harus mengecek stok satu per satu secara manual.
- c) Pembuatan laporan penjualan dan laporan obat masuk memakan waktu, karena seluruh data dihitung ulang secara manual sebelum dilaporkan.

Kondisi ini menyebabkan ketidakefisienan, rendahnya akurasi data, serta keterlambatan dalam pengambilan keputusan seperti menentukan kebutuhan pengadaan obat.

3.2.2 Penerapan Sistem Informasi Berbasis Web sebagai Solusi

Hasil analisis sistem menunjukkan bahwa sistem berbasis web mampu memberikan solusi lebih efektif karena mendukung:

- a) Pengelolaan data obat secara real-time dan terintegrasi
- b) Akses data yang lebih cepat dan mudah
- c) Rekapitulasi laporan otomatis sehingga lebih akurat.

Sistem dirancang menggunakan UML dan dikembangkan menggunakan framework CodeIgniter, bahasa pemrograman PHP, dan database MySQL, dengan pendekatan Waterfall.

Unified Modeling Language(UML) adalah suatu teknik untuk memodelkan sistem. UML ditemukan oleh Grady Booch, Ivar Jacobson, dan James Rumbaugh [6].

MySQL adalah sebuah program database server yang mampu menerima dan mengirimkan datanya sangat cepat, multi user serta menggunakan perintah dasar SQL. MySQL juga merupakan sebuah database server yang free, artinya bebas menggunakan database untuk keperluan pribadi atau usaha tanpa harus membeli atau membayar lisensinya.[7]

Fitur sistem yang dibangun ini meliputi :

- a) Pengelolaan data obat (CRUD)
- b) Pencatatan obat masuk
- c) Pencatatan penjualan
- d) Pembuatan purchase order (PO)
- e) Monitoring stok menipis
- f) Laporan penjualan harian/bulanan/tahunan
- g) Cetak laporan dan invoice
- h) Notifikasi PO pending

Fitur ini dirancang langsung untuk menjawab kekurangan pada sistem lama.

3.2.3 Hasil Implementasi Sistem

Setelah sistem diimplementasikan, terdapat beberapa peningkatan signifikan dalam operasional apotek:

- a) Peningkatan Efisiensi Pengelolaan Obat Sistem otomatis melakukan:

- 1) Update stok ketika obat masuk atau terjual.
- 2) Menampilkan stok secara akurat pada dashboard.

Efisiensi meningkat karena apoteker tidak lagi mencatat manual dan menghitung stok secara manual.

- b) Percepatan Pemantauan Ketersediaan Obat
Fitur “stok menipis” dan ringkasan dashboard memudahkan apoteker mengetahui kondisi stok terkini hanya dalam satu tampilan. Hal ini membantu pengambilan keputusan terkait pengadaan lebih cepat dan akurat.
- c) Akses Data yang Lebih Akurat dan Cepat
Laporan dapat dihasilkan secara otomatis dalam bentuk harian, bulanan, hingga tahunan.
Sistem juga menyediakan fitur cetak laporan dan stok dalam format yang siap dipresentasikan atau diserahkan kepada pihak terkait.
- d) Minimnya Kesalahan Pencatatan
Seluruh proses pencatatan terkomputerisasi sehingga human error dapat diminimalisir, sesuai dengan tujuan awal penelitian.

3.2.3 Hasil Pengujian Sistem

Pengujian menggunakan Black Box Testing, dan seluruh skenario pengujian menunjukkan hasil *valid* dan sesuai harapan.

Contoh hasil pengujian yang valid:

- a) Login dan dashboard tampil dengan benar.
- b) Stok otomatis menyesuaikan ketika data obat ditambah atau dijual.
- c) Menu cetak laporan bekerja sesuai harapan.
- d) PO pending tampil dengan akurat.

Hal ini menunjukkan bahwa fitur-fitur berjalan stabil dan sesuai kebutuhan apotek.

3.2.4 Kesesuaian Sistem dengan Tujuan Penelitian

Berdasarkan hasil analisis dan implementasi, sistem ini terbukti mampu memenuhi ketiga tujuan utama penelitian:

- a) Mempermudah pengelolaan operasional apotek
Sistem membantu pengelola dalam mencatat obat masuk, keluar, dan membuat laporan secara otomatis.
- b) Mempercepat pemantauan ketersediaan obat
Dashboard dan notifikasi stok menipis membuat pemantauan lebih cepat.
- c) Menyediakan akses data yang cepat dan akurat
Laporan digital memastikan data lebih tepat dan mudah diakses kapan saja.

3.2.5 Keterbatasan Sistem

Beberapa keterbatasan sistem diantaranya :

- a) Sistem hanya digunakan untuk operasional Apotek Rostie, tidak terintegrasi dengan apotek lain.
- b) Sistem hanya mengelola data administratif, bukan data medis pasien.
- c) Belum mendukung pembayaran elektronik atau integrasi bank/payment gateway

3.2.6 Implikasi dan Rekomendasi Pengembangan

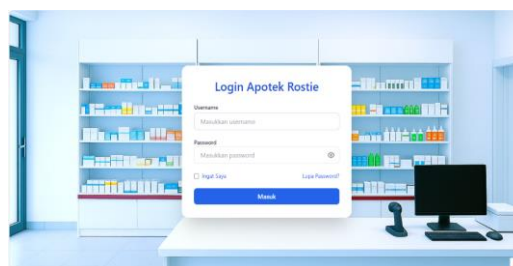
Peneliti merekomendasikan:

- a) Pengembangan sistem ke platform mobile untuk akses yang lebih fleksibel.
- b) Pembaruan fitur secara berkala agar tetap relevan dengan kebutuhan apotek modern.
- c) Pemeliharaan rutin untuk memastikan stabilitas dan keamanan sistem.

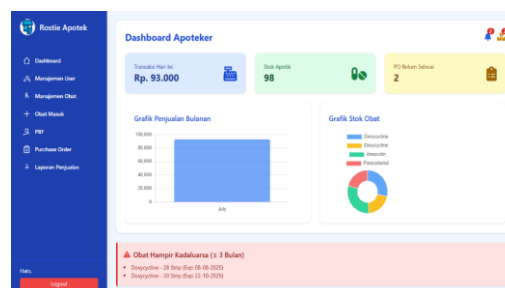
Rekomendasi ini memungkinkan sistem berkembang menjadi lebih komprehensif dimasa depan.

3.3. Disain Input dan Output

Berikut ini akan ditampilkan beberapa contoh tampilan input dan output dari implementasi aplikasi web pada apotik Rostie, tampilan input seperti Dashboard Login, Dashboard Apoteker, Dashboard Kasir, Dashboard PSA dapat dilihat di gambar 2 sampai gambar 5. Sedangkan untuk tampilan output dapat dilihat pada gambar 6 sampai gambar 8.



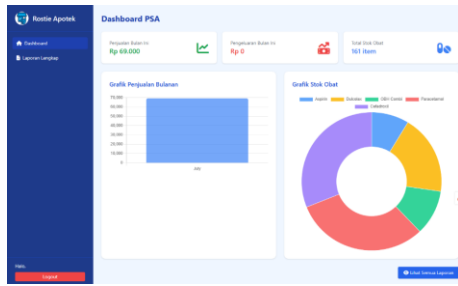
Gambar 2. Dashboard Login



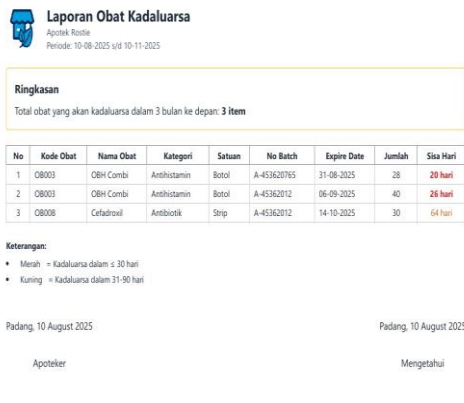
Gambar 3. Dashboard Apoteker

Tanggal	Kode	Total	Kasir
2025-07-28	780-20250728-0002	Rp14.000	Hana
2025-07-28	780-20250728-0001	Rp15.000	Hana

Gambar 4. Dashboard Kasir



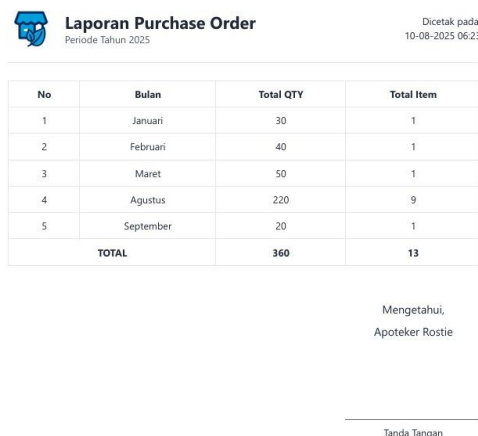
Gambar 5. Dashboard PSA



Gambar 6. Laporan Obat Kadaluausa



Gambar 7. Output Nota Penjualan



Gambar 8. Laporan Purchase Order

3.4. Pengujian

Pengujian sistem adalah pengujian program perangkat lunak yang lengkap dan terintegrasi. Perangkat lunak atau yang sering dikenal dengan sebutan software hanyalah satuan elemen dari sistem berbasis komputer yang lebih besar. Biasanya, perangkat lunak dihubungkan dengan perangkat lunak dan perangkat keras lainnya.

Pengujian perangkat lunak adalah proses atau rangkaian proses yang dirancang untuk memastikan bahwa program computer menjalankan apa yang seharusnya dilakukan dan sebaliknya, memastikan program agar tidak melakukan hal yang tidak diharapkan. Sebuah perangkat lunak seharusnya dapat diprediksi dan konsisten. Tugas utama dari seorang penguji adalah untuk menemukan bug atau error sebanyak mungkin serta mengetahui bagaimana error atau bug itu dihasilkan. Sehingga dapat dikatakan bahwa pengujian adalah proses mengeksekusi program dengan tujuan untuk menemukan error. Pengujian merupakan proses penting dalam siklus pengembangan software untuk memastikan kualitas dari sebuah software. [5]

Proses pengujian perangkat lunak adalah serangkaian aktivitas yang sistematis dan terstruktur, yang dirancang untuk memvalidasi bahwa sebuah sistem perangkat lunak memenuhi persyaratan yang telah ditentukan dan untuk mendeteksi masalah atau bug yang mungkin ada. Proses ini melibatkan serangkaian langkah-langkah yang saling terkait, yaitu [8] :

- a) **Perencanaan Pengujian (Test Planning)**
Ini adalah tahap awal dalam proses pengujian perangkat lunak. Disini, tim pengujian menentukan strategi dan pendekatan pengujian, menentukan sumber daya yang dibutuhkan, dan menentukan jadwal dan estimasi waktu pengujian. Selain itu, tim pengujian juga mengidentifikasi risiko yang mungkin ada dan merencanakan mitigasinya.
- b) **Analisis dan Desain Pengujian (Test Analysis and Design)**
Ditahap ini, tim pengujian menganalisis persyaratan dan desain perangkat lunak untuk mengidentifikasi apa saja yang harus diuji. Tim pengujian juga merancang kasus-kasus pengujian dan skenario pengujian yang akan digunakan selama proses pengujian.
- c) **Implementasi dan Eksekusi Pengujian (Test Implementation and Execution)**
Setelah kasus-kasus pengujian dan scenario pengujian dirancang, tahap selanjutnya adalah implementasi dan eksekusi pengujian. Tim pengujian menjalankan kasus-kasus pengujian pada perangkat lunak dan mencatat hasilnya.

- d) **Evaluasi Siklus Pengujian (Test Cycle Closure)**
Setelah semua kasus pengujian dijalankan, tim pengujian mengevaluasi hasil pengujian dan menentukan apakah perangkat lunak telah memenuhi persyaratan yang ditentukan. Jika terdapat kasus pengujian yang gagal, tim pengujian menganalisis penyebabnya dan membuat rekomendasi untuk perbaikan.
- e) **Pemeliharaan (Maintenance)**
Setelah perangkat lunak dirilis dan digunakan, tim pengujian terus memantau perangkat lunak dan melakukan pengujian regresi untuk memastikan bahwa perubahan atau pembaruan pada perangkat lunak tidak menimbulkan masalah baru.

Proses pengujian perangkat lunak ini membantu memastikan bahwa perangkat lunak yang dikembangkan bebas dari bug atau kesalahan, berfungsi sebagaimana mestinya, dan memenuhi semua persyaratan yang telah ditentukan. Proses ini juga membantu dalam memahami kualitas perangkat lunak dan memberikan informasi yang dibutuhkan untuk membuat keputusan tentang apakah perangkat lunak siap untuk dirilis atau tidak.

Pengujian perangkat lunak dapat dibedakan menjadi dua yaitu Black Box Testing dan White Box Testing. Berikut merupakan metode pengujian system [5] :

3.4.1. Black Box Testing

Black Box Testing atau yang sering dikenal dengan sebutan pengujian fungsional merupakan metode pengujian Perangkat Lunak yang digunakan untuk menguji perangkat lunak tanpa mengetahui struktur internal kode atau program.

3.4.2. White Box Testing

White Box Testing merupakan metode pengujian perangkat lunak dimana struktur internal diketahui untuk menguji siapa yang akan menguji perangkat lunak. Pengujian ini membutuhkan pengetahuan internal tentang kemampuan sistem dan pemrograman. Contoh dari pengujian white box adalah ketika menguji alur dengan menelusuri pengulangan pada logika pemrograman.

Black Box testing merupakan metode uji fungsionalitas sistem aplikasi. Dalam melakukan pengujian menggunakan masukan data acak dengan tujuan untuk mendapatkan hasil yang pasti. Dikatakan pasti artinya bila salah ,maka ditolak oleh sistem informasi atau data input tersebut tidak dapat disimpan dalam data base, sedangkan bila data input benar maka dapat diterima / masuk di database sistem informasi. [9]

Proses Black Box Testing dengan cara mencoba program yang telah dibuat dengan mencoba memasukkan data pada setiap formnya. Pengujian ini diperlukan untuk mengetahui program tersebut berjalan sesuai dengan yang dibutuhkan oleh perusahaan. [10]

Dalam pengujian ini, tester menyadari apa yang harus dilakukan oleh program tetapi tidak memiliki pengetahuan tentang bagaimana melakukannya.

Misalnya menguji proses login maka kasus uji yang dibuat adalah :

- Jika user memasukkan nama pemakai (username) dan kata sandi (password) yang benar.
- Jika user memasukkan nama pemakai dan kata sandi yang salah, misalnya nama pemakai benar tetapi kata sandi salah atau sebaliknya, atau keduanya salah.

Teknik pengujian pada Black Box Testing ada banyak macamnya yaitu [9]:

- Teknik Equivalence Partitioning : yaitu dengan cara melakukan partition atau pembagian menjadi beberapa partisi dari input data.
- Teknik Boundary Value Analysis : yaitu dengan cara mencari adakah error dari luar atau sisi dalam software, minimum maupun maximum nilai dari error yang ditemukan.
- Teknik Fuzzing : yaitu merupakan teknik untuk mencari Bug/gangguan dari software dengan menggunakan injeksi data yang terbilang cacat .
- Teknik Cause-Effect Graph : ialah suatu Teknik testing dimana menggunakan graphic sebagai acuannya. Dimana dalam grafik ini menggambarkan relasi diantara efek dan penyebabnya.
- Teknik Orthogonal Array Testing : adalah jenis Teknik yang digunakan jika input domain yang relative terbilang kecil ukurannya, tetapi cukup berat untuk digunakan dalam skala besar.
- Teknik All Pair Testing : yaitu semua pasangan dari test case didesain sedemikian rupa agar dapat dieksekusi semua kemungkinan kombinasi diskrit dari seluruh pasangan berdasar input parameternya, Tujuan testing ini adalah memiliki pasangan test case yang mencakup semua pasangan tersebut.
- Teknik State Transition : Teknik ini berguna untuk melakukan pengetesan terhadap kondisi dari mesin dan navigasi dalam bentuk grafik.

Penyimpulan keberhasilan sistem berhasil atau tidak dapat dilakukan pembuatan tabel pengujian yang akan dilakukan dengan metode Black Box berbasis Equivalence Partitions. Berikut tabel pengujiannya:

Tabel 1. Hasil pengujian dengan metode blackbox testing

No	Skenario Test	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Data obat tidak diisi lengkap, kemudian klik tombol Simpan	Sistem menolak dan menampilkan pesan "Please fill out this field"	Sesuai harapan	Valid

2	Data obat diisi lengkap dan valid, kemudian klik tombol Simpan	Sistem menyimpan data dan menampilkan pesan "Berhasil menyimpan data"	Sesuai harapan	Valid
3	Klik tombol Edit dan ubah data obat dengan data yang valid	Sistem menyimpan perubahan dan menampilkan pesan "Berhasil mengedit data"	Sesuai harapan	Valid
4	Gunakan fitur pencarian untuk mencari nama obat tertentu	Sistem menampilkan data obat yang sesuai dengan kata kunci	Sesuai harapan	Valid

Selanjutnya pengujian dilakukan untuk melihat data obat yang ada pada Apotik Rostie. Berikut tampilan table dari hasil pengujian untuk melihat data Obat :

Tabel 2. Pengujian lihat stok Apotik

No	Skenario Test	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Login sebagai apoteker dan buka dashboard	Sistem menampilkan summary box seperti: Transaksi Hari Ini, Stok Apotik, dan PO Belum Selesai	Sesuai harapan	Valid
2	Tidak ada data stok apotik dalam database	Sistem menampilkan nilai 0 pada summary box stok apotik	Sesuai harapan	Valid
3	Tambahkan data obat dan cek apakah jumlah stok apotik bertambah	Summary box "Stok Apotik" langsung memperbarui jumlah sesuai data yang dimasukkan	Sesuai harapan	Valid

Pemeliharaan sistem merupakan tahap penting setelah sistem informasi apotek selesai dibangun dan diimplementasikan. Tujuan dari tahap ini adalah untuk memastikan sistem tetap berjalan dengan baik, mengatasi berbagai permasalahan yang muncul,

serta menyesuaikan sistem dengan kebutuhan baru di masa mendatang. Pemeliharaan sistem dalam proyek ini dilakukan dalam beberapa bentuk, yaitu:

1. Pemeliharaan Korektif (Corrective Maintenance)
Pemeliharaan ini dilakukan untuk memperbaiki kesalahan atau bug yang ditemukan dalam sistem, baik dari sisi tampilan antarmuka, logika pemrograman, maupun data yang tidak sesuai.

2. Pemeliharaan Adaptif (Adaptive Maintenance)
Pemeliharaan adaptif dilakukan ketika sistem perlu disesuaikan dengan perubahan lingkungan, seperti pembaruan browser, framework, atau kebutuhan pengguna.

3. Pemeliharaan Perfektif (Perfective Maintenance)
Dilakukan untuk meningkatkan kinerja sistem atau menambah fungsionalitas agar sistem semakin efisien dan mudah digunakan.

4. Pemeliharaan Preventif (Preventive Maintenance)
Merupakan pemeliharaan yang dilakukan untuk mencegah terjadinya kerusakan di masa depan. Hal ini mencakup backup data rutin, pengecekan performa server, dan pemantauan log sistem untuk mencegah gangguan besar.

4. Kesimpulan

Sistem Informasi Apotek Berbasis Web pada Apotek Rostie Padang, dimana sistem tersebut mampu mempermudah proses pengelolaan operasional obat, mulai dari pencatatan obat masuk, dan penjualan secara lebih terstruktur dan efisien.

Sistem informasi yang dikembangkan mampu mempercepat proses pemantauan ketersediaan obat, sehingga pihak apotek dapat mengambil tindakan lebih cepat apabila terjadi kekosongan stok atau kebutuhan pengadaan ulang.

Sistem ini juga mampu menyediakan akses data yang cepat dan akurat, sehingga proses pencarian data obat, transaksi, dan laporan dapat dilakukan dengan mudah melalui fitur pencarian dan rekap otomatis yang tersedia di dalam sistem.

References

- [1] L. Nurlaela, A. Dharmalau, and N. T. Parida, "Rancangan Sistem Informasi Inventory Barang Berbasis Web Studi Kasus Pada CV. Limoplast," *Syntax Idea*, vol. 2, no. 5, 2020.
- [2] E. Effendi, R. S. A. Sagalai, and S. Rezeki, "Jenis-Jenis Sistem Informasi dan Model Sistem Informasi," *J. Pendidik. dan Konseling*, vol. 5, no. 2, 2023.
- [3] W. Waliyah, F. Aprilyani, M. Muchlis, and V. Febriyanti, "Pengembangan Sistem Informasi Penjualan Berbasis Web," *J. Sitem Inf. STMIK Antar Bangsa*, vol. 14, no. 2, 2025, doi: 10.47709/digitech.v4i1.3955.
- [4] K. Samosir and Y. G. Nengsih, "Pengembangan Sistem Informasi Berbasis Web dengan Menggunakan Metode Agile: Studi Kasus Pengelolaan Proyek TI," *J. Inform. Komput. dan Bisnis*, vol. 4, no. 2, 2024, [Online]. Available: <https://jurnal.itbaas.ac.id/index.php/jikobis>

- [5] F. N. Hasanah and R. S. Untari, *Rekayasa Perangkat Lunak*. Sidoarjo: UMSIDA Press, 2020.
- [6] M. Pato, H. A. Gani, and M. Idhkan, "Perancangan Sistem Informasi Manajemen Pembelajaran Mata Kuliah Kewirausahaan dalam Mendukung Program Merdeka Belajar Kampus Merdeka (MBKM) di Perguruan Tinggi," *Cived*, vol. 10, no. 1, 2023, doi: 10.24036/cived.v10i1.373112.
- [7] G. Indrawan and I. N. Y. Setyawan, *Database MySQL Dengan Pemrograman PHP*. Depok: PT RajaGrafindo Persada, 2018.
- [8] S. R. Wicaksono, *Pengujian Perangkat Lunak : Strategi, Metode dan Implementasi*. Malang: CV. Seribu Bintang, 2023.
- [9] U. Uminingsih, M. N. Ichsanudin, M. Yusuf, and S. Suraya, "Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode Black Box Testing Bagi Pemula," *STORAGE – J. Ilm. Tek. dan Ilmu Komput.*, vol. 1, no. 2, 2022.
- [10] J. Shadiq, A. Safei, and R. W. R. Loly, "Pengujian Aplikasi Peminjaman Kendaraan Operasional Kantor Menggunakan BlackBox Testing," *Inf. Manag. Educ. Prof.*, vol. 5, no. 2, 2021, doi: 10.51211/imbi.v5i2.1561.