

Perancangan Sistem Informasi Laporan Keuangan dan Masa Aktif Hosting Berbasis Web Menggunakan Codeigniter 4

Akbar Rizky Dio Saputra^{*1}, Muhammad Galih Arifin², Khairul Ikhsan³, Riyandul Aziz⁴

¹²³⁴Prodi S1 Informatika, ¹²³⁴STMIK Amikom Surakarta
Sukoharjo - Indonesia

Email: ¹akbar.10355@mhs.amikomsolo.ac.id,

²muhhammad.10368@mhs.amikomsolo.ac.id, ³khairul.10356@mhs.amikomsolo.ac.id,

⁴riyan@dosen.amikomsolo.ac.id

Abstract

The rapid development of information technology has encouraged digital-based solutions in various business processes, including financial recording and hosting service management. Sampryzen Hosting, as a provider of game hosting services, requires an integrated system to simplify financial data management and monitoring of hosting active periods, which previously were done manually and prone to errors. This study aims to design and develop a web-based financial reporting and hosting active period information system using the CodeIgniter 4 framework. The prototyping method was applied to enable iterative interaction between developers and users during system development. The system features include financial transaction recording, monthly report generation, server list management, and role-based access control for admin and user. To evaluate the system, Blackbox Testing was conducted to ensure functional suitability, and System Usability Scale (SUS) testing was performed to measure system usability. The results of the trial show that all functional components operate correctly based on Blackbox Testing, while the average SUS score of 44 indicates that the system's usability still needs improvement in terms of interface simplicity and user guidance. These findings demonstrate that the system is functionally ready but requires further refinement before full implementation.

Keywords: Admin, CodeIgniter 4, Financial report, Hosting, Prototyping, User.

Abstraksi

Perkembangan teknologi informasi mendorong hadirnya sistem digital yang dapat membantu proses bisnis, termasuk dalam pencatatan keuangan dan pengelolaan masa aktif hosting. Sampryzen Hosting sebagai penyedia layanan hosting game membutuhkan sistem yang mampu mengelola laporan keuangan dan informasi server secara efisien karena pencatatan manual sering menimbulkan kesalahan dan keterlambatan. Penelitian ini bertujuan merancang dan membangun sistem informasi laporan keuangan dan masa aktif hosting berbasis web menggunakan framework CodeIgniter 4. Metode prototyping digunakan untuk memfasilitasi proses pengembangan yang bersifat interaktif antara pengguna dan pengembang. Fitur utama sistem meliputi pencatatan transaksi, pembuatan laporan bulanan, pengelolaan server list, serta akses berbasis peran untuk admin dan user. Pengujian dilakukan menggunakan Blackbox Testing dan System Usability Scale (SUS). Hasil uji coba menunjukkan bahwa seluruh fungsi sistem berjalan

dengan baik berdasarkan Blackbox Testing, sedangkan hasil SUS rata-rata sebesar 44 menunjukkan bahwa sistem masih perlu penyempurnaan terutama pada aspek tampilan dan kemudahan penggunaan. Dengan demikian, sistem dinyatakan layak secara fungsional, namun memerlukan pengembangan lebih lanjut sebelum diterapkan secara penuh.

Kata Kunci: Admin, CodeIgniter 4, Hosting, Laporan Keuangan, Prototyping, User

1. PENDAHULUAN

Teknologi informasi yang semakin maju membawa dampak besar dalam semua bidang kehidupan termasuk dalam pencatatan keuangan dan layanan digital seperti domain dan *hosting*. Digitalisasi kini menjadi kebutuhan utama dalam menjalankan bisnis agar dapat bersaing dan bertahan di era industri 4.0. Salah satu solusi yang banyak diterapkan adalah pengembangan sistem informasi berbasis *web* yang mampu mengelola data secara *real time*, akurat, dan efisien [1]. Pencatatan laporan keuangan dan pemantauan masa aktif *hosting* merupakan komponen penting dalam manajemen layanan digital. Sistem manual yang masih sering digunakan, seperti pencatatan pada buku atau *spreadsheet* rawan kesalahan dan tidak efisien dalam menampilkan informasi yang diperlukan secara cepat[2].

Aplikasi berbasis *web* tidak hanya meningkatkan efisiensi, tetapi juga membantu meminimalisir kesalahan manusia [3]. *Framework CodeIgniter 4* menjadi pilihan banyak pengembang aplikasi karena keunggulannya dalam pengembangan sistem berbasis *web*, *CodeIgniter* memiliki struktur file sederhana, dan mendukung model MVC (*Model-View-Controller*) yang memungkinkan pengembangan aplikasi lebih cepat dan terstruktur[4]. Hal ini diperkuat oleh penelitian Anggraini yang menyatakan bahwa penerapan *CodeIgniter* dalam sistem penjualan mampu meningkatkan keakuratan dan efisiensi pengolahan data [5].

Sistem penjualan berbasis *web* dengan *framework CodeIgniter 4* dapat memberikan pengalaman pengguna yang lebih baik dan meningkatkan produktivitas operasional perusahaan [6]. Sistem informasi penjualan sangat membantu dalam pembuatan laporan dan pengelolaan transaksi bisnis [7]. Dalam bidang pengelolaan arsip dan data transaksi, integrasi sistem informasi juga terbukti meningkatkan efisiensi pencairan data dan keamanan arsip [8].

Dalam studi kasus pengembangan sistem penjualan hosting, pemanfaatan *platform WHMCS* serta integrasi sistem informasi dinilai penting untuk menyederhanakan pengelolaan *domain*, *hosting*, serta laporan keuangannya [9]. Sementara itu, Kusnadi dan Putra menekankan pentingnya transformasi digital melalui sistem *e-commerce* berbasis *web* untuk meningkatkan daya saing UMKM dan menjangkau pasar yang lebih luas [10].

Terakhir, pengembangan sistem penjualan dengan pendekatan *framework* seperti *CodeIgniter* juga terbukti mampu memfasilitasi bisnis lokal dalam membangun sistem *e-commerce* seperti yang dilakukan oleh Ferdinandus [11] menggunakan *CMS Opencart* untuk meningkatkan penjualan *DnD Collection*.

Sampryzen Hosting adalah sebuah usaha yang bergerak di bidang pelayanan dan penyewaan *hosting game*. Jumlah pelanggan yang sudah mencapai 100 orang perbulan membuat pemilik mengalami kesulitan jika melakukan pencatatan keuangan dan masa aktif *hosting* secara manual. Karena itu, penelitian ini bertujuan untuk membuat sebuah sistem informasi laporan keuangan dan masa aktif hosting berbasis *web* dengan menggunakan *framework CodeIgniter 4*. Dengan adanya sistem ini diharapkan dapat menjadi solusi yang efisien, akurat, dan *user-friendly* dalam mendukung pengelolaan layanan *hosting* secara menyeluruh.

2. TINJAUAN PUSTAKA

Dalam perkembangan teknologi saat ini, banyak kegiatan administrasi yang dulunya dilakukan secara manual dan mulai diganti dengan sistem informasi berbasis web. Salah satu alat bantu dalam membuat sistem tersebut adalah *framework CodeIgniter*. *Framework* ini banyak digunakan karena mempermudah programmer dalam menyusun dan mengatur alur kerja sistem.

Penelitian dari Reza Fahlevi dkk. [12] menunjukkan bahwa sistem penggajian manual di sebuah perusahaan jasa seperti PO Arista Tehnik Jakarta sering mengalami kesalahan dalam penghitungan gaji. Oleh karena itu, mereka merancang sebuah sistem penggajian berbasis *web* dengan *CodeIgniter* dan *MySQL*. Hasilnya, sistem tersebut bisa menampilkan data karyawan, jabatan, serta laporan gaji secara otomatis dan efisien dibanding cara manual.

Penelitian lainnya dilakukan oleh Rahman Sastrawan dkk. [13] di Masjid An-Nur. Mereka membuat sistem akuntansi berbasis *web* menggunakan *CodeIgniter* untuk mempermudah bendahara masjid dalam mencatat pemasukan dan pengeluaran uang. Sebelumnya, semua pencatatan dilakukan secara manual dan sulit untuk dilaporkan kepada masyarakat. Sistem yang dibuat mampu menyajikan laporan keuangan dengan lebih rapi dan mudah diakses oleh pengurus masjid.

Dari contoh kedua penelitian di atas, dapat disimpulkan bahwa penggunaan *framework CodeIgniter* dalam pembuatan sistem informasi sangat membantu dalam mempermudah pekerjaan, menghindari kesalahan, dan mempercepat proses pengolahan data. Hal ini mendukung tujuan dari penelitian ini untuk membuat sistem laporan keuangan dan masa aktif *hosting* yang lebih efisien, mudah digunakan, dan dapat menampilkan data secara *real time*.

3. METODE PENELITIAN

Peneliti menggunakan metode *Prototyping* dalam penelitian ini dikarenakan metode ini memiliki keunggulan ketika digunakan dalam perancangan dan pembuatan sebuah sistem dalam waktu yang singkat. Dengan menghasilkan *prototype* sistem sebagai perantara pengembang dan pengguna untuk berinteraksi ketika proses pengembangan sistem informasi ini[14]. Beberapa tahapan dalam metode *prototyping* adalah pengumpulan data, rancangan awal, pengembangan *prototype*, dan analisis performa.



Gambar 1. Metode Penelitian

Digunakannya metode ini diharapkan dapat memberikan gambaran kepada pengguna terkait desain dan fungsi sistem yang dirancang. Tahapan yang dilakukan dalam penelitian ini dengan detail sebagai berikut:

3.1. Pengumpulan Data

Pada tahap ini data diperoleh melalui wawancara kepada pemilik dan admin sampyren *hosting*. Data yang diperoleh meliputi fitur apa saja yang dibutuhkan untuk merancang *website* pengelolaan keuangan seperti: *server list*, *user list* dan laporan keuangan.

3.2. Rancangan Awal

Pada tahap ini dilakukan analisis berdasarkan kebutuhan pengguna yang diperoleh dari tahapan sebelumnya. Analisis dilakukan untuk memahami alur sistem secara keseluruhan dan merumuskannya menjadi sebuah *flowchart* yang menggambarkan bagaimana sistem berjalan.

3.3. Pengembangan Prototype

Setelah dilakukan analisis kebutuhan dan merumuskan *flowchart*, tahap selanjutnya adalah mengembangkan sebuah rancangan dari sistem. *Prototype* yang dikembangkan meliputi halaman *login*, *dashboard*, *administrator*, *server list*, dan laporan bulanan.

3.4. Analisis Performa

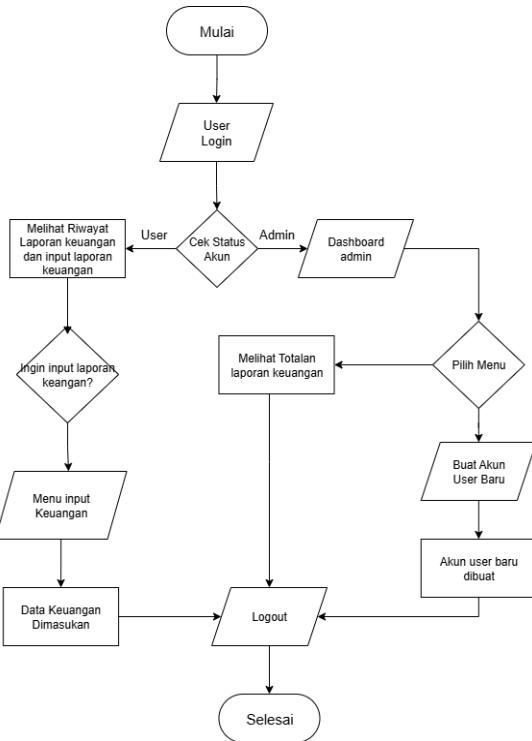
Tahap terakhir yang dilakukan adalah analisis performa atau pengetesan dengan menggunakan *blackbox testing*. Digunakannya metode ini diharapkan dapat mengetahui setiap tombol dan fitur yang dibuat dapat berfungsi dengan baik.

4. HASIL DAN PEMBAHASAN

Dalam penelitian ini, peneliti membuat beberapa hal untuk mendukung proses perancangan sistem informasi laporan keuangan dan masa aktif *hosting*. Beberapa hal yang dibuat adalah *flowchart*, *use case diagram*, dan *class diagram* untuk membantu perancangan *prototype*. *Prototype* dibuat untuk membuat visual dari sistem yang akan dirancang. Dan pada tahap analisis menggunakan metode *blackbox testing* dan *System Usability Scale* untuk menentukan apakah sistem dapat digunakan dengan mudah.

4.1. Flowchart

Flowchart digunakan untuk menentukan alur sistem informasi laporan keuangan dan masa aktif *hosting*. Alur dimulai dari awal *user login* hingga mengakses seluruh fitur dan kemudian *logout*. Detailnya bisa dilihat pada Gambar 2.

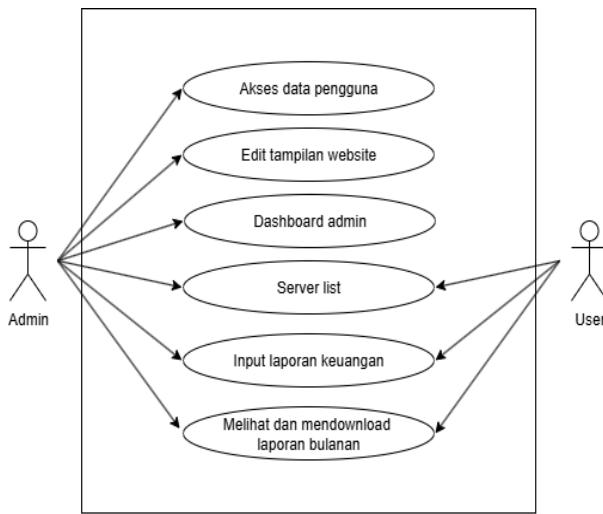


Gambar 2. Flowchart

Pada Gambar 2. menjelaskan tentang alur Sistem Informasi Laporan Keuangan dan Masa Aktif *Hosting*. Proses dimulai dengan *login*, kemudian akun akan diverifikasi dan melakukan pengecekan status akun. Jika terverifikasi *admin*, maka akan diarahkan ke *dashboard admin*. Admin dapat mengakses semua menu yang ada, selain itu *admin* juga dapat mengakses menu yang hanya bisa diakses oleh admin yaitu membuat akun baru dan menghapusnya. Jika terverifikasi sebagai user hanya bisa mengakses menu yang dapat digunakan untuk melihat laporan keuangan, input laporan keuangan dan total laporan keuangan.

4.2. Use Case Diagram

Use case diagram untuk menentukan hak akses dari masing masing *role* akun. Pada sistem informasi keuangan dan masa aktif *hosting* mempunyai 2 jenis *role* yaitu *admin* dan *user*. Perbedaan hak akses dapat menjadi keamanan tambahan untuk sistem dikarenakan akun *user* dibatasi aksesnya hanya untuk melihat *server list*, input laporan keuangan, dan mengolah laporan bulanan. Detailnya bisa dilihat pada Gambar 3.

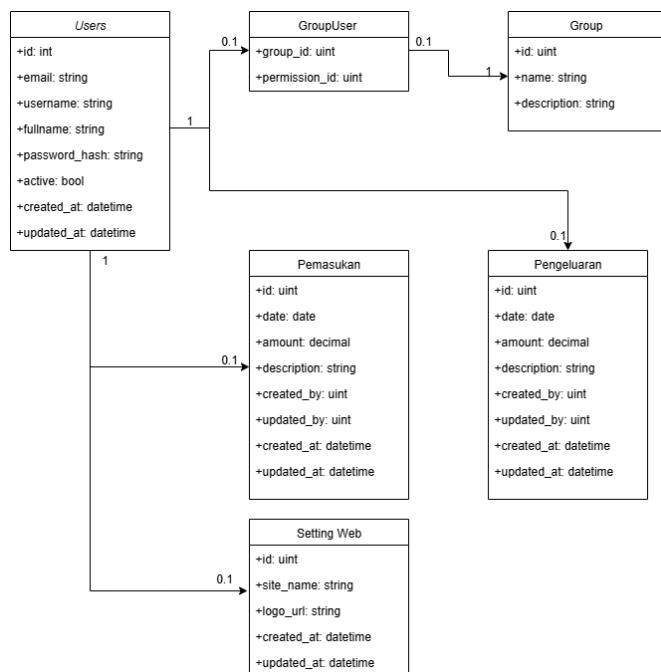


Gambar 3. *Use case Diagram*

Pada Gambar 3. menjelaskan perbedaan akses antara *Admin* dan *User* biasa. *Admin* memiliki hak akses penuh seperti mengelola data pengguna, edit *website*, input laporan, hingga melihat laporan bulanan. Sedangkan *User* hanya dapat input laporan keuangan dan melihat serta mendownload laporan keuangan bulanan.

4.3. Class Diagram

Class diagram untuk menjelaskan alur dari pemrosesan database yang dilakukan dalam sistem. Setiap entitas mempunyai hubungan antar entitas agar sistem dapat berjalan dengan baik. Detailnya bisa dilihat pada Gambar 4.

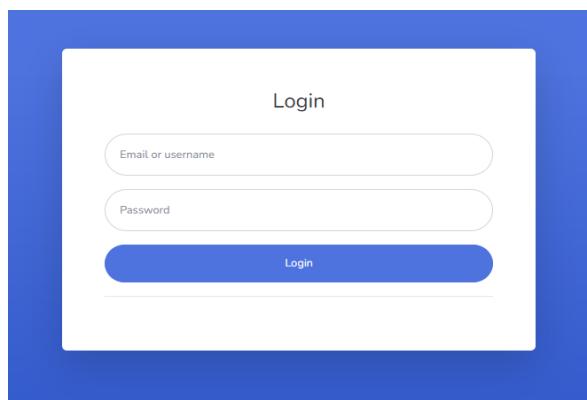


Gambar 4. *Class Diagram*

Pada Gambar 4. menjelaskan struktur basis data *Sistem Informasi Laporan Keuangan*. Entitas utama terdiri dari *Users*, *GroupUser*, *Group*, Pemasukan, Pengeluaran, dan *Setting Web*. *Users* terhubung dengan *GroupUser* dan *Group* untuk menentukan hak akses (*admin* atau *user*). Entitas Pemasukan dan Pengeluaran menyimpan data transaksi keuangan, sedangkan *Setting Web* digunakan untuk konfigurasi dasar aplikasi. Struktur ini memungkinkan sistem mengatur perbedaan peran pengguna sekaligus mengelola data keuangan dan pengaturan website secara terintegrasi

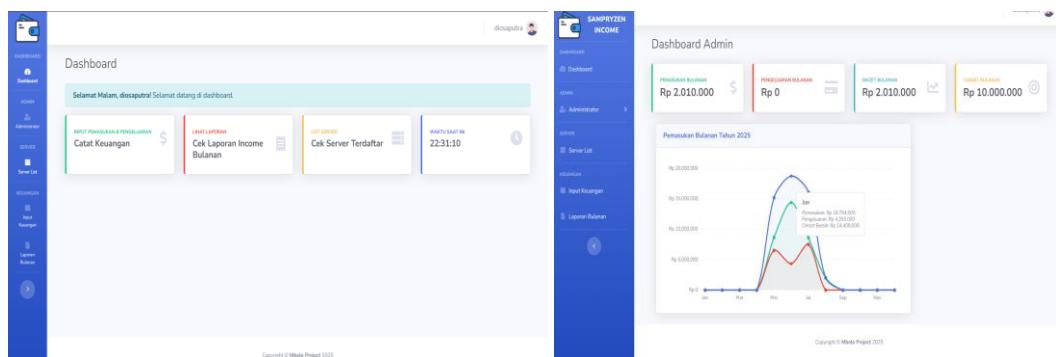
4.4. Desain Prototype

Prototype didesain berdasarkan kebutuhan sistem yang telah didapatkan dari proses wawancara. Beberapa halaman utama yang akan dirancang adalah tampilan *login*, *dashboard admin*, *server list* dan laporan keuangan.



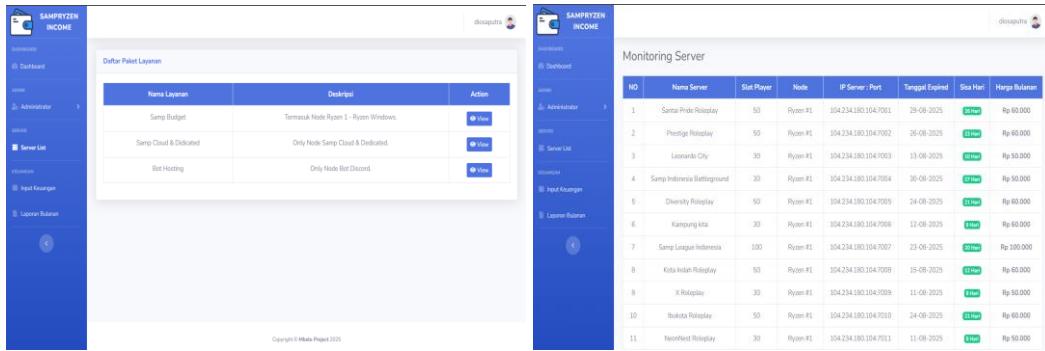
Gambar 5. *Login*

Gambar 5. adalah halaman yang digunakan untuk *login*, sistem akan melakukan verifikasi sesuai dengan kategori akun yang digunakan. Jika akun yang digunakan akun admin maka akan diarahkan ke *dashboard admin* seperti pada Gambar 6.



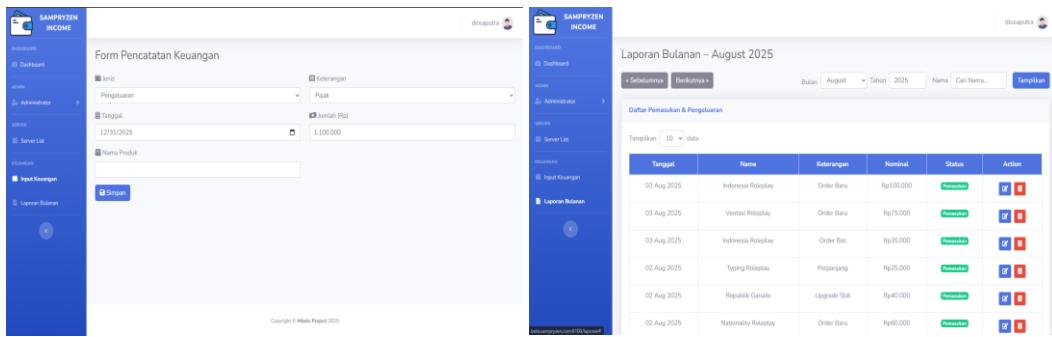
Gambar 6. *Dashboard Admin*

Gambar 6. adalah tampilan *dashboard admin*, terdapat beberapa menu yang dapat dipilih seperti: *dashboard admin*, *server list*, dan laporan keuangan bulanan.



Gambar 7. Server List

Gambar 7. adalah tampilan menu *server list*, pada halaman ini dapat digunakan untuk menampilkan server yang telah diinput lengkap dengan nama *server*, *slot player*, *node*, *ip server*, tanggal *expired*, sisa durasi dan harga bulannya.



Gambar 8. Laporan Keuangan

Gambar 8. adalah halaman yang digunakan untuk melakukan input pemasukan, pengeluaran dan laporan dari total pemasukan dan pengeluaran setiap bulannya.

4.5. Blackbox Testing

Tabel 1. Blackbox Testing

Input	Proses	Output	Kesimpulan
Masukkan username dan password > klik login	Get data dari database berdasarkan email	Masuk ke halaman dashboard	Berhasil
Klik Profil	Beralih ke halaman profil	Tampilan profile	Berhasil
Ubah Password	Update database password	Muncul modals input password lama dan input password baru	Berhasil

Klik Catatan Keuangan	Beralih ke halaman keuangan	Menampilkan halaman input keuangan	Berhasil
Insert data keuangan	Add data ke database bagian table Pemasukan	Muncul notifikasi data berhasil dimasukan.	Berhasil
Klik Laporan Bulanan	Berpindah ke halaman laporan	Menampilkan riwayat input	Berhasil
Klik logo edit	Mengambil data dari table Pemasukan	Menampilkan modals dan detail pemasukan	Berhasil
Klik Simpan	Update Data terbaru untuk detail	Menampilkan notifikasi berhasil di edit	Berhasil
Klik Server List	Beralih ke halaman server	Menampilkan pilihan menu setiap paket	Berhasil
Klik Logo view	Beralih Ke halaman detail server list	Menampilkan data list server yang terdaftar	Berhasil
Klik Administrator > dashboard admin	Beralih ke halaman dashboard admin	Menampilkan chart dan total pemasukan bulanan	Berhasil
Klik Administrator > Settings	Beralih ke halaman settings	Menampilkan halaman setting untuk website	Berhasil
Klik Simpan perubahan	Update Database di table Settings	Muncul notifikasi berhasil diupdate	Berhasil
Klik Reset Ke Default	Update Database di table Settings	Muncul notifikasi tampilan web disetting ke default	Berhasil
Klik Administrator > Users	Beralih ke halaman users list	Menampilkan halaman list user yang terdaftar	Berhasil
Klik Detail	Mengambil Data setiap user dan beralih ke halaman detail	Menampilkan halaman detail users	Berhasil

Setelah dilakukan perancangan sistem, dilakukan tahap analisis performa dari sistem dengan menggunakan metode *blackbox testing* untuk menguji fungsional sistem dan *System Usability Scale* (SUS) untuk mengetahui apakah sistem dapat digunakan

dengan mudah oleh pengguna. Dari hasil pengujian pada Tabel 1. tersebut, semua fitur sistem berjalan sesuai dengan fungsinya. Mulai dari login, setting profil, catatan keuangan, laporan bulanan, hingga *server list* dapat digunakan tanpa kendala. Fitur admin seperti *dashboard*, pengaturan *website*, dan manajemen user juga berfungsi dengan baik. Dengan demikian, sistem sudah memenuhi kebutuhan dan dapat digunakan sesuai tujuan.

4.6. System Usability Scale (SUS)

Tabel 2. *System Usability Scale*

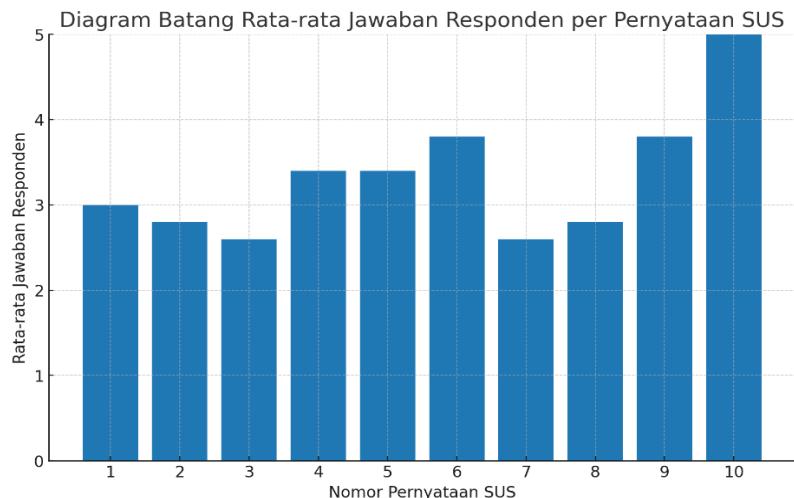
No	Pernyataan	Nilai				
		1	2	3	4	5
1	Saya akan sering menggunakan sistem laporan keuangan dan hosting ini.				✓	
2	Menurut saya sistem ini terlalu rumit untuk digunakan.		✓			
3	Menurut saya sistem ini mudah dioperasikan tanpa harus banyak belajar.				✓	
4	Menurut saya harus meminta bantuan kepada orang lain agar bisa menggunakan sistem ini.		✓			
5	Saya merasa menu dan fitur dalam sistem ini saling mendukung dan terhubung dengan baik.				✓	
6	Menurut saya ada beberapa hal yang tidak stabil dalam sistem ini (misalnya tampilan atau cara kerja menu).		✓			
7	Menurut saya orang lain (misalnya teman mahasiswa) akan cepat mengerti cara memakai sistem ini.					✓
8	Menurut saya sistem ini membingungkan ketika digunakan.		✓			
9	Saya merasa percaya diri saat menggunakan sistem ini untuk mencatat laporan atau cek masa aktif hosting.					✓
10	Menurut saya harus mempelajari beberapa hal dulu sebelum bisa mengoperasikan sistem ini.		✓			

Pengujian *System Usability Scale* (SUS) dilakukan terhadap 5 responden calon user dengan menggunakan pertanyaan pada Tabel 2. untuk mengetahui sejauh mana sistem informasi laporan keuangan dan masa aktif *hosting* ini mudah digunakan oleh pengguna. Dari hasil pengisian kuisioner, skor SUS tiap responden adalah: 45,0; 40,0; 47,5; 40,0; dan 47,5. Jika dirata-rata, skor SUS yang diperoleh adalah 44,0.

Berdasarkan standar interpretasi SUS, skor tersebut termasuk dalam kategori kurang baik (*poor*). Hal ini menunjukkan bahwa meskipun sistem sudah bisa digunakan dan semua fitur berjalan, dari sisi kemudahan penggunaan (*usability*) sistem ini masih belum sepenuhnya memuaskan pengguna.

Beberapa hal yang kemungkinan membuat nilai SUS rendah antara lain: tampilan antarmuka yang masih terasa rumit, adanya bagian menu yang dianggap kurang konsisten, serta kebutuhan bantuan dari orang lain untuk memahami cara penggunaan sistem.

Dengan hasil ini, bisa disimpulkan bahwa sistem masih perlu diperbaiki terutama pada bagian tampilan (UI) agar lebih sederhana, memperbaiki konsistensi antar menu, dan menyediakan panduan penggunaan yang jelas. Dengan perbaikan tersebut diharapkan sistem akan lebih *user friendly* dan lebih mudah digunakan baik oleh *admin* maupun *user* biasa.



Gambar 9. Hasil Pengujian *System Usability Scale* (SUS)

5. KESIMPULAN

Berdasarkan proses perancangan, pembuatan, dan pengujian sistem informasi laporan keuangan dan masa aktif *hosting* yang telah dilakukan, dapat disimpulkan bahwa sistem ini dibuat untuk memberikan solusi terhadap permasalahan pencatatan manual yang selama ini digunakan oleh *Sampryzen Hosting*. Sistem dirancang agar mampu membantu admin maupun user dalam mengelola transaksi keuangan, melihat laporan bulanan, serta memantau masa aktif *server* secara lebih terstruktur, akurat, dan efisien. Hasil uji coba menggunakan *Blackbox Testing* menunjukkan bahwa seluruh fitur dalam

sistem berjalan sesuai dengan fungsinya, mulai dari proses login, pengelolaan data pengguna, input keuangan, pembuatan laporan bulanan, hingga pengelolaan *server list*. Hal ini mengindikasikan bahwa dari sisi fungsionalitas, sistem telah memenuhi kebutuhan utama dan siap untuk digunakan dalam lingkungan operasional.

Namun, berdasarkan hasil pengujian *System Usability Scale* (SUS) dengan skor rata-rata 44, dapat disimpulkan bahwa aspek kemudahan penggunaan masih perlu ditingkatkan. Tampilan antarmuka, konsistensi menu, serta penyediaan panduan penggunaan menjadi beberapa aspek yang harus diperbaiki agar sistem lebih mudah dioperasikan oleh pengguna, baik admin maupun user biasa. Secara keseluruhan, sistem ini telah berhasil dirancang dan diuji secara fungsional, namun sebelum diterapkan secara penuh masih diperlukan penyempurnaan pada bagian antarmuka pengguna dan pengalaman penggunaan (*user experience*).

DAFTAR PUSTAKA

- [1] G. A. Wulandari, A. A. Riadi, dan A. Susanto, "Sistem Informasi Pengelolaan Stok Bahan Baku Roti secara Real-Time berbasis Web," *Edumatic: Jurnal Pendidikan Informatika*, vol. 9, no. 1, hlm. 159–168, Apr 2025, doi: 10.29408/edumatic.v9i1.29605.
- [2] K. S. Kartini, N. Tri, A. Putra, K. J. Atmaja, N. Putu, dan S. Widiani, "SISTEM INFORMASI PENJUALAN PADA SALAD YOO," 2022. [Daring]. Tersedia pada: <https://ejournal.catuspata.com/index.php/jkdn/index>
- [3] M. Ridwan, T. H. Sinaga, dan M. Elsera, "PENERAPAN FRAMEWORK CODEIGNITER DALAM PERANCANGAN APLIKASI MANAJEMEN IURAN PERUMAHAN GRIYA MANDIRI," 2022.
- [4] Y. Anggraini, D. Pasha, dan A. Setiawan, "SISTEM INFORMASI PENJUALAN SEPEDA BERBASIS WEB MENGGUNAKAN FRAMEWORK CODEIGNITER (STUDI KASUS : ORBIT STATION)," *Jurnal Teknologi dan Sistem Informasi (JTSI)*, vol. 1, no. 2, hlm. 64–70, 2020, [Daring]. Tersedia pada: <http://jim.teknokrat.ac.id/index.php/JTSI>
- [5] R. Fahmi, A. Salam, P. Studi Manajemen Informatika, S. Indonesia Banda Aceh, dan K. Banda Aceh, "Jurnal Sistem Komputer (SISKOM) Rancang Bangun Platform Penjualan Domain Dan Hosting Berbantuan Whmcs Berbasis Web," *Jurnal Sistem Komputer (SISKOM)*, vol. 3, no. 1, 2023, doi: 10.35870/siskom.v3i1.793.
- [6] Y. Kusnadi dan D. W. Putra, "E-Commerce Berbasis Website pada UMKM Menggunakan Framework Codeigniter 4 (Studi Kasus: Toko Wakuteka)," *Jurnal Teknologi Informatika dan Komputer*, vol. 10, no. 1, hlm. 257–271, Mar 2024, doi: 10.37012/jtik.v10i1.2101.
- [7] B. P. Rahadi, I. Fauzi, N. Prawitasari, dan W. Haryono, "Perancangan Sistem Kasir Berbasis Web pada Toko Komputer Intechcom untuk Meningkatkan Efisiensi Manajemen," 2024.

- [8] S. Auliana dkk., "PENERAPAN FRAMEWORK CODEIGNITER 4 DALAM APLIKASI PENJUALAN PAKAIAN BERBASIS WEB PADA DISTRO DISTROBUTOR," 2023, doi: 10.46306/ncabet.v3i1.
- [9] A. Selay dkk., "SISTEM INFORMASI PENJUALAN," 2023.
- [10] N. Firdaus dan D. Irfan, "Rancang Bangun Sistem Informasi Arsip Berbasis Web Menggunakan Framework Codeigniter," *Voteteknika (Vocational Teknik Elektronika dan Informatika)*, vol. 8, no. 1, hlm. 44, Mar 2020, doi: 10.24036/voteteknika.v8i1.107759.
- [11] F. Lidang Witi, Y. D. Da, Y. Khwuta, J. Sam, R. Ende, dan F. Ntt, "Rancang Bangun Sistem E-Commerce Menggunakan Aplikasi Content Management System OpenCart Untuk Meningkatkan Penjualan (Studi kasus pada DnD Collection)," 2022.
- [12] R. Fahlevi, Z. Zulhalim, dan A. S. Rini, "PERANCANGAN APLIKASI PENGAJIAN KARYAWAN BERBASIS WEB MENGGUNAKAN FRAMEWORK CODEIGNITER PADA PO ARISTA TEHNIK JAKARTA," *Jurnal Manajemen Informatika Jayakarta*, vol. 1, no. 2, hlm. 95, Apr 2021, doi: 10.52362/jmijayakarta.v1i2.446.
- [13] Rahman Sastrawan, Chornolius Hendreo, dan Arief Rio Maulana, "PERANCANGAN SISTEM INFORMASI AKUNTANSI BERBASIS FRAMEWORK CODEIGNITER PADA MASJID AN NUR KABUPATEN SANGGAU," *JURNAL LENTERA AKUNTANSI*, vol. 9, no. 2, hlm. 144–156, Nov 2024, doi: 10.34127/jrakt.v9i2.1228.