

PERANCANGAN SISTEM INFORMASI GYM BERBASIS DIGITAL DENGAN METODE WATERFALL

Bagus Satya Ramadhan^{1*}, Apriade Voutama²,

^{1,2}Universitas Singaperbangsa Karawang; Jl. HS. Ronggo Waluyo, Puseurjaya, Telukjambe Timur, Karawang, Jawa Barat 41361; 0267641177

Keywords:

Sistem Informasi;
Manajemen Gym; PHP;
MySQL; *Waterfall*; *Black Box Testing*.

Correspondent Email:

satyaramadhanbagus@gmail.com



Copyright © 2026 The Author(s), Published by Jurnal Informatika dan Teknik Elektro Terapan. This is an open-access article distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Abstrak. Pengelolaan administrasi pusat kebugaran secara manual menimbulkan berbagai permasalahan seperti ketidakakuratan data anggota, lambatnya proses pembayaran, dan tidak tersedianya laporan keuangan secara langsung. Penelitian ini bertujuan merancang dan membangun Sistem Informasi Manajemen Gym XYZ berbasis web menggunakan PHP dan MySQL dengan metode pengembangan *Waterfall*. Sistem yang dibangun mencakup lima modul utama, yaitu *Login & Autentikasi*, *Pendaftaran Member*, *Pembayaran Iuran*, *Pengelolaan Jadwal Kelas*, serta *Laporan & Rekap*. Sistem ini menerapkan *Role-Based Access Control* (RBAC) yang membagi hak akses ke dalam dua peran, yaitu Admin dan Karyawan. Pengujian dilakukan menggunakan metode *Black Box Testing* terhadap 22 skenario uji pada 5 modul utama dan seluruh fungsionalitas berjalan sesuai yang diharapkan dengan tingkat keberhasilan 100%.

Abstract. Manual administration management in fitness centers causes various problems such as inaccurate member data, slow payment processes, and unavailability of real-time financial reports. This study aims to design and build a web-based Gym XYZ Management Information System using PHP and MySQL with the *Waterfall* development method. The system includes five main modules: *Login & Authentication*, *Member Registration*, *Fee Payment*, *Class Schedule Management*, and *Reports & Recap*. Testing was conducted using the *Black Box Testing* method on 22 test scenarios across 5 main modules, with a 100% success rate.

1. PENDAHULUAN

Perkembangan teknologi informasi telah membawa perubahan mendasar di berbagai sektor industri, termasuk dalam pengelolaan pusat kebugaran. Kemajuan ini membuka peluang bagi pengelola gym untuk beralih dari sistem administrasi konvensional menuju pengelolaan berbasis digital yang lebih efisien dan terstruktur. Sistem informasi berbasis komputer memegang peranan penting dalam mendukung operasional bisnis, terutama dalam hal pengelolaan data yang akurat dan terintegrasi [1].

Di era transformasi digital, efektivitas pengolahan data menjadi kebutuhan yang tidak

dapat diabaikan dalam industri kebugaran. Sistem informasi merupakan sekumpulan komponen yang saling terhubung untuk mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi guna mendukung pengambilan keputusan dalam suatu organisasi [2],[3]. Sistem informasi berbasis *web* memiliki keunggulan dalam hal aksesibilitas karena pengguna dapat mengakses sistem kapan saja dan dari mana saja tanpa instalasi perangkat lunak tambahan, menjadikannya sangat relevan untuk lingkungan bisnis yang dinamis seperti pusat kebugaran [4].

Salah satu tantangan utama yang masih dihadapi banyak pusat kebugaran adalah

pengelolaan administrasi yang belum terkomputerisasi. Sistem manual menciptakan berbagai permasalahan seperti proses yang memakan waktu lama, kerentanan terhadap kesalahan pencatatan, dan sulitnya menghasilkan laporan yang akurat serta tepat waktu [5]. Penelitian terdahulu membuktikan bahwa sistem informasi keanggotaan *fitness* yang dirancang dengan baik mampu menyederhanakan proses dari pendaftaran hingga pembuatan laporan secara signifikan [6], [7].

Gap yang ditemukan pada penelitian-penelitian sebelumnya adalah belum adanya sistem manajemen gym yang mengintegrasikan fitur verifikasi kehadiran berbasis QR code sekaligus menerapkan *Role-Based Access Control* secara menyeluruh dalam satu platform. Penelitian ini hadir untuk mengisi kesenjangan tersebut dengan membangun sistem informasi manajemen gym yang lengkap, terintegrasi, dan aman. Tujuan penelitian ini adalah merancang dan membangun sistem informasi gym berbasis *web* untuk meningkatkan efisiensi administrasi XYZ Fitness Center menggunakan PHP, MySQL, dan metode *Waterfall*.

2. TINJAUAN PUSTAKA

Sistem Informasi. Sistem informasi merupakan sekumpulan komponen yang saling terhubung untuk mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi guna mendukung pengambilan keputusan dan kendali dalam suatu organisasi [2]. Penerapan sistem informasi berbasis *web* pada usaha kebugaran terbukti mampu meningkatkan efisiensi operasional dan akurasi data secara signifikan dibandingkan dengan sistem pencatatan manual [5].

Metode *Waterfall*. Metode *Waterfall* adalah pendekatan pengembangan perangkat lunak yang bersifat linier dan sekuensial, di mana setiap fase harus diselesaikan terlebih dahulu sebelum fase berikutnya dimulai. Fase-fase dalam metode ini meliputi *Requirements Analysis*, *System Design*, *Implementation*, *Testing*, *Deployment*, dan *Maintenance* [8]. Metode ini terbukti efektif diterapkan pada proyek sistem informasi dengan kebutuhan yang sudah terdefinisi sejak awal [9].

PHP dan MySQL. PHP merupakan bahasa pemrograman skrip sisi *server* yang bersifat

open-source dan dirancang khusus untuk pengembangan aplikasi *web* dinamis, yang memungkinkan pengembang membangun halaman *web* yang berinteraksi langsung dengan basis data secara efisien [10]. Kombinasi PHP dan MySQL menjadi pasangan teknologi yang sangat populer karena kompatibilitasnya yang baik serta ketersediaannya dalam paket pengembangan seperti XAMPP [11].

Unified Modeling Language (UML). UML merupakan bahasa standar yang digunakan untuk visualisasi, spesifikasi, dan dokumentasi sistem perangkat lunak, yang mencakup berbagai jenis diagram seperti *Use Case Diagram*, *Activity Diagram*, dan *Sequence Diagram* [12]. Penerapan diagram UML membantu pengembang mengidentifikasi hak akses dan alur kerja sistem sebelum implementasi dilakukan [13].

Entity Relationship Diagram (ERD). ERD digunakan sebagai representasi grafis yang menggambarkan entitas-entitas yang terlibat beserta atribut dan relasi antar entitas dalam sebuah basis data. ERD menjadi tahap awal yang krusial dalam perancangan *database* agar terhindar dari anomali penyimpanan dan pembaruan data [14].

Black Box Testing. *Black Box Testing* adalah metode pengujian perangkat lunak yang berfokus pada evaluasi fungsionalitas sistem dari perspektif pengguna akhir, tanpa memperhatikan struktur kode internal. Metode ini memverifikasi apakah setiap fungsi sistem menghasilkan *output* yang sesuai dengan yang diharapkan berdasarkan masukan yang diberikan [15].

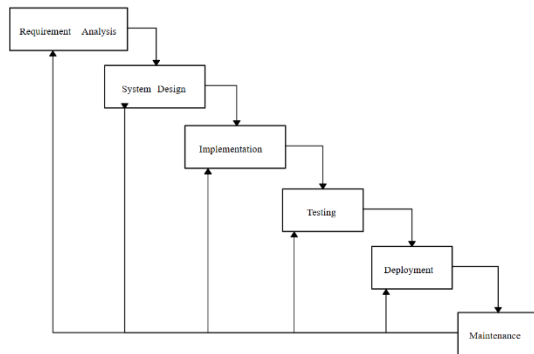
Role-Based Access Control (RBAC). RBAC merupakan mekanisme keamanan sistem yang membatasi akses pengguna berdasarkan peran yang ditetapkan. Penerapan RBAC dalam sistem informasi terbukti efektif dalam menjaga keamanan data dan membatasi akses pengguna sesuai dengan tingkatan otorisasi yang telah ditetapkan [6].

3. METODE PENELITIAN

3.1 Tahapan Penelitian

Penelitian ini menggunakan metode *Waterfall* yang terdiri dari enam fase utama yang dilaksanakan selama 16 minggu secara berurutan. Metode *Waterfall* dipilih karena

kebutuhan sistem XYZ Fitness Center telah terdefinisi dengan jelas sejak awal melalui observasi dan wawancara langsung, sehingga alur pengembangan yang linier dan terstruktur menjadi pilihan yang paling tepat.



Gambar 1. Tahapan Penelitian Metode Waterfall

Rincian kegiatan dan *output* dari setiap fase disajikan pada Tabel 1 berikut.

Fase	Periode	Kegiatan	Output
<i>Requirements Analysis</i>	Minggu 1-2	Wawancara stakeholder, observasi, dokumentasi kebutuhan	Dokumen kebutuhan sistem
<i>System Design</i>	Minggu 3-5	Perancangan basis data, arsitektur, desain UI/UX	<i>Blueprint</i> sistem
<i>Implementation</i>	Minggu 6-11	Pengembangan <i>backend</i> , <i>frontend</i> , integrasi basis data	<i>Source code</i> dan basis data
<i>Testing</i>	Minggu 12-14	<i>Black Box Testing</i> , perbaikan <i>bug</i> , <i>regression testing</i>	Laporan hasil pengujian
<i>Deployment</i>	Minggu 15	Konfigurasi server, migrasi data, pelatihan pengguna	Sistem berjalan
<i>Maintenance</i>	Minggu 16+	Monitoring, perbaikan <i>bug</i> , pengembangan lanjutan	Sistem terpelihara

Tabel 1. Rencana tahapan pengembangan sistem metode waterfall

3.2 Penerapan Setiap Fase

Fase 1 *Requirements Analysis* (Minggu 1–2). Dilakukan wawancara langsung dengan pemilik dan staf administrasi XYZ Fitness Center serta observasi terhadap proses pencatatan manual yang sedang berjalan. Hasil observasi mengidentifikasi tiga permasalahan utama, yaitu ketidakakuratan data anggota akibat pencatatan manual, lambatnya proses pembayaran iuran, dan tidak tersedianya laporan keuangan secara langsung bagi pemilik. Seluruh kebutuhan yang diidentifikasi kemudian didokumentasikan dalam dokumen spesifikasi kebutuhan sistem.

Fase 2 *System Design* (Minggu 3–5). Kebutuhan yang telah disetujui ditransformasikan menjadi rancangan teknis yang komprehensif, mencakup basis data MySQL beserta skema relasi antar entitas, arsitektur sistem berbasis PHP, desain antarmuka dalam bentuk *wireframe* dan prototipe, serta pembuatan diagram UML yang meliputi *Use Case Diagram*, *Activity Diagram*, dan *Sequence Diagram*.

Fase 3 *Implementation* (Minggu 6–11). Seluruh rancangan diwujudkan menjadi kode program menggunakan PHP dan MySQL, mencakup pengembangan logika bisnis sisi *server*, pembangunan antarmuka berbasis HTML dan CSS, konfigurasi basis data, serta integrasi seluruh komponen. Pengembangan dilakukan secara modular mengikuti lima modul utama yang telah didefinisikan.

Fase 4 *Testing* (Minggu 12–14). Pengujian dilakukan menggunakan metode *Black Box Testing* untuk memverifikasi fungsionalitas seluruh modul dari sudut pandang pengguna akhir. Setiap *bug* yang ditemukan segera diperbaiki dan diuji ulang melalui *regression testing* sebelum sistem dinyatakan layak diluncurkan.

Fase 5 *Deployment* (Minggu 15). Sistem diluncurkan ke lingkungan operasional nyata mencakup konfigurasi *server*, migrasi data dari sistem pencatatan lama, serta pelatihan intensif kepada seluruh pengguna akhir agar sistem dapat dimanfaatkan secara optimal.

Fase 6 *Maintenance* (Minggu 16+). Fase ini bersifat berkelanjutan mencakup *monitoring* rutin, penanganan gangguan dari pengguna, perbaikan *bug* pasca-rilis, serta pengembangan fitur tambahan sesuai kebutuhan.

3.3 Kebutuhan Fungsional Sistem

Kebutuhan fungsional sistem mencakup lima modul utama yang disajikan pada Tabel 2 berikut.

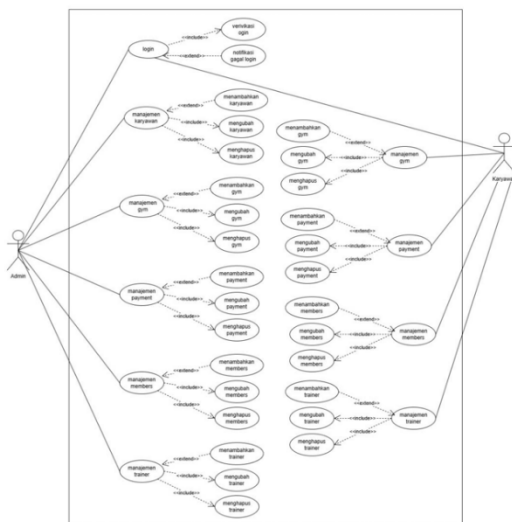
No	Nama Modul	Fungsi	Pengguna
1	Login & Autentikasi	Mengelola keamanan akses sistem menggunakan RBAC	Admin, Member
2	Pendaftaran Member	Mengelola data anggota secara digital	Admin
3	Pembayaran Iuran	Mencatat transaksi dan histori pembayaran anggota	Admin, Kasir
4	Pengelolaan Jadwal	Mengatur jadwal kelas, instruktur, dan kapasitas peserta	Admin
5	Laporan & Rekap	Menyediakan laporan data anggota dan keuangan <i>real-time</i>	Admin

Tabel 2. Modul Sistem Informasi XYZ Fitness Center

4. HASIL DAN PEMBAHASAN

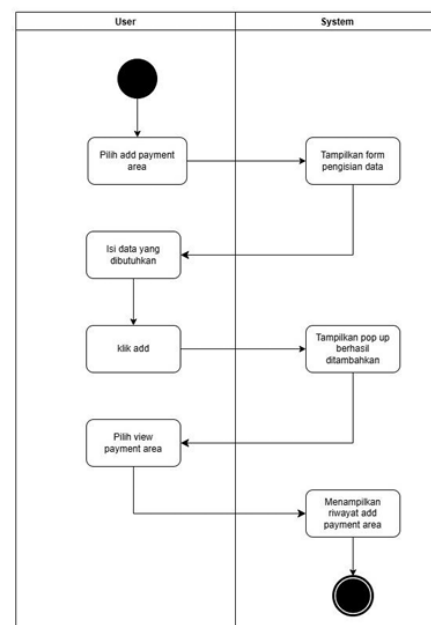
4.1 Perancangan Diagram UML

Perancangan sistem dimulai dengan pembuatan diagram UML yang mencakup tiga jenis diagram utama sebagai cetak biru sistem sebelum tahap implementasi dilakukan.



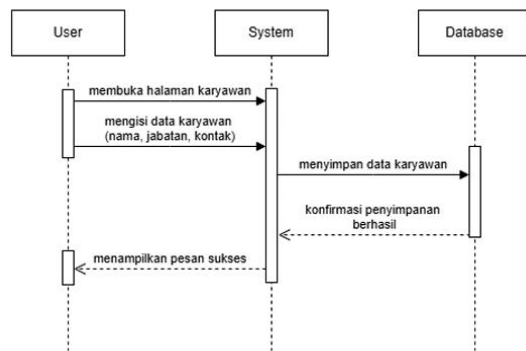
Gambar 2. Use Case Diagram

Use Case Diagram. Diagram ini menggambarkan interaksi antara dua aktor utama, yaitu Admin dan Karyawan, dengan berbagai fitur yang tersedia dalam sistem. Admin memiliki akses penuh terhadap seluruh fitur sistem, mulai dari proses *login* hingga pengelolaan data karyawan, *gym*, *payment area*, *members*, dan *trainer*. Setiap modul manajemen memiliki tiga *use case* inti yakni menambahkan, mengubah, dan menghapus data. Terdapat relasi *include* yang menghubungkan *use case* utama dengan fungsi pendukungnya, serta relasi *extend* untuk alur alternatif seperti notifikasi gagal *login* dan konfirmasi penghapusan data. Karyawan memiliki akses yang lebih terbatas sesuai hak akses yang diberikan oleh sistem.



Gambar 3. Activity Diagram Payment Area

Activity Diagram. Diagram ini dirancang untuk menggambarkan alur kerja setiap proses dalam sistem. Terdapat 16 *activity diagram* yang mencakup proses *login*, tambah, edit, dan hapus data untuk seluruh modul. Setiap proses penambahan data mengikuti pola yang konsisten, yaitu pengguna mengisi formulir, mengklik tombol, lalu sistem menampilkan konfirmasi keberhasilan. Untuk proses penghapusan, sistem menambahkan langkah konfirmasi terlebih dahulu guna mencegah penghapusan yang tidak disengaja.



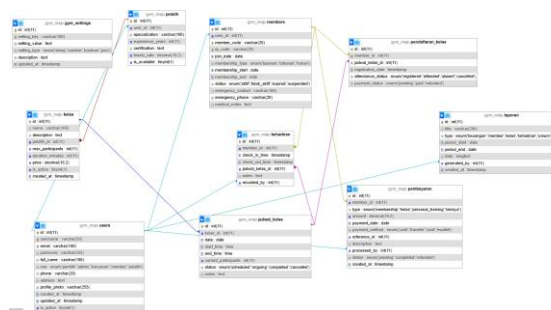
Gambar 4. Sequence Diagram Karyawan

Sequence Diagram. Diagram ini menggambarkan interaksi antara tiga komponen utama, yaitu *User*, *System*, dan *Database*, secara berurutan berdasarkan waktu. Terdapat 16 *sequence diagram* yang mencerminkan setiap proses bisnis dalam sistem. Pada proses *login*, pengguna memasukkan kredensial, sistem mengirimkan permintaan validasi ke *database*, lalu menampilkan *dashboard* jika berhasil atau pesan *error* jika gagal.

4.2 Perancangan Struktur Database dan ERD

Table	Action	Rows	Type	Collation	Size	Overhead
gym_settings	✱ Browse	1	InnoDB	utf8mb4_general_ci	32.0 K B	-
jadwal_kelas	✱ Browse	1	InnoDB	utf8mb4_general_ci	32.0 K B	-
kehadiran	✱ Browse	1	InnoDB	utf8mb4_general_ci	64.0 K B	-
kelas	✱ Browse	1	InnoDB	utf8mb4_general_ci	32.0 K B	-
laporan	✱ Browse	1	InnoDB	utf8mb4_general_ci	32.0 K B	-
members	✱ Browse	1	InnoDB	utf8mb4_general_ci	64.0 K B	-
pelatih	✱ Browse	1	InnoDB	utf8mb4_general_ci	32.0 K B	-
pembayaran	✱ Browse	1	InnoDB	utf8mb4_general_ci	48.0 K B	-
pendaftaran_kelas	✱ Browse	1	InnoDB	utf8mb4_general_ci	48.0 K B	-
users	✱ Browse	1	InnoDB	utf8mb4_general_ci	48.0 K B	-
10 tables	Sum				82 InnoDB utf8mb4_general_ci 432.0 K B	0 B

Gambar 5. Perancangan Struktur Database



Gambar 6. Entity Relationship Diagram (ERD)

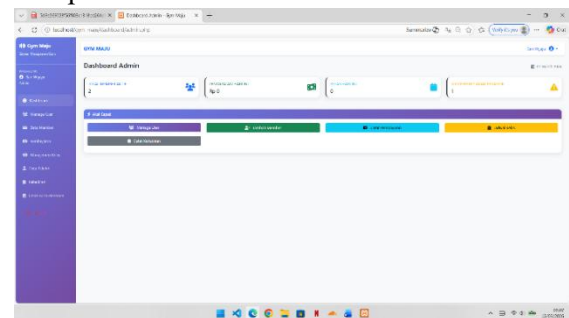
Sistem menggunakan sepuluh tabel *database* yang saling berelasi dengan total 82 baris data

dan ukuran 432 KiB menggunakan *engine* InnoDB. Tabel-tabel tersebut meliputi *gym_settings*, *jadwal_kelas*, *kehadiran*, *kelas*, *laporan*, *members*, *pelatih*, *pembayaran*, *pendaftaran_kelas*, dan *users*. Tabel *users* menjadi tabel utama yang menyimpan data seluruh pengguna sistem dan berelasi dengan tabel *pelatih* melalui *user_id* serta tabel *laporan* melalui *created_by*. Entitas *pendaftaran_kelas* menghubungkan *members* dan *jadwal_kelas* sebagai relasi *many-to-many*, sementara *kehadiran* mencatat partisipasi *member* pada setiap sesi.

4.3 Implementasi Sistem

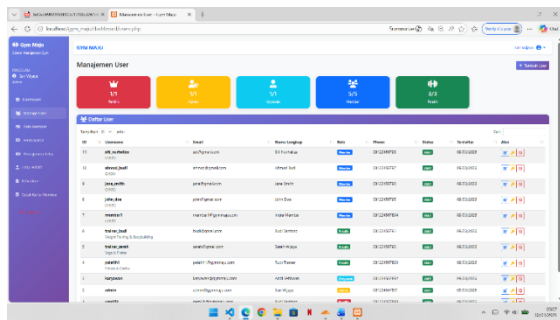
4.3.1 admin

Tahap implementasi merupakan perwujudan nyata dari seluruh rancangan yang telah dibuat. Sistem dibangun menggunakan PHP dengan basis data MySQL dan diakses melalui antarmuka berbasis *web*. Berikut adalah penjelasan dari setiap halaman yang telah diimplementasikan.



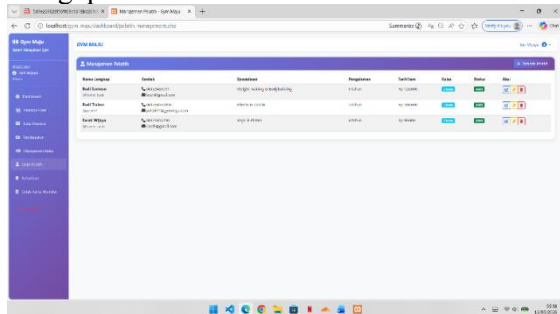
Gambar 7. Halaman Dashboard

Halaman Dashboard. Halaman utama yang pertama kali tampil setelah pengguna berhasil *login*. *Dashboard* menampilkan ringkasan informasi operasional secara *real-time*, meliputi jumlah total *member* aktif, total pendapatan, jumlah kelas yang berjalan, serta statistik kehadiran harian. Tampilan ini dirancang agar Admin maupun Karyawan dapat langsung memantau kondisi operasional gym tanpa harus membuka menu satu per satu.



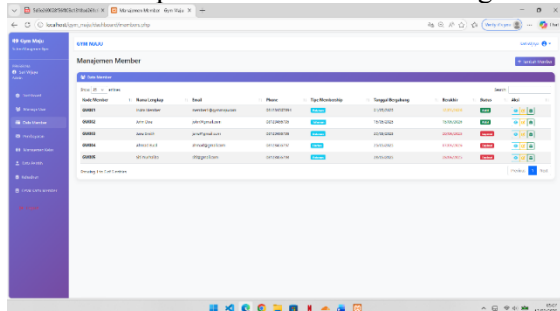
Gambar 8. Halaman Manajemen User

Halaman Manajemen User. Digunakan oleh Admin untuk mengelola akun pengguna yang dapat mengakses sistem. Admin dapat menambahkan akun baru, mengatur peran (*role*) pengguna sebagai Admin atau Karyawan, mengubah data pengguna yang sudah ada, serta menonaktifkan akun yang tidak lagi diperlukan. Fitur ini memastikan bahwa hanya pihak yang berwenang yang dapat masuk dan mengoperasikan sistem.



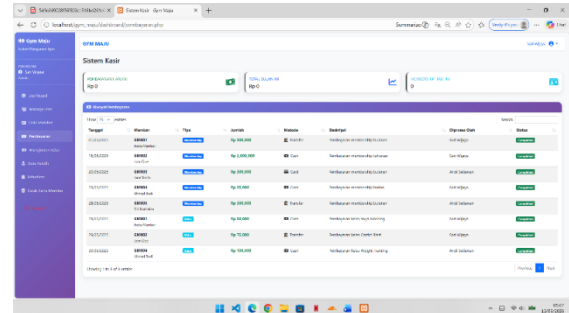
Gambar 9. Halaman Manajemen Pelatih

Halaman Manajemen Pelatih. Menampilkan daftar seluruh pelatih yang terdaftar beserta informasi lengkapnya seperti nama, spesialisasi, jadwal mengajar, dan status aktif. Admin dapat menambahkan pelatih baru, memperbarui data yang sudah ada, serta menghapus data pelatih yang sudah tidak aktif. Data pelatih terhubung langsung dengan modul pengelolaan jadwal kelas sehingga penugasan instruktur dapat dilakukan secara terintegrasi.



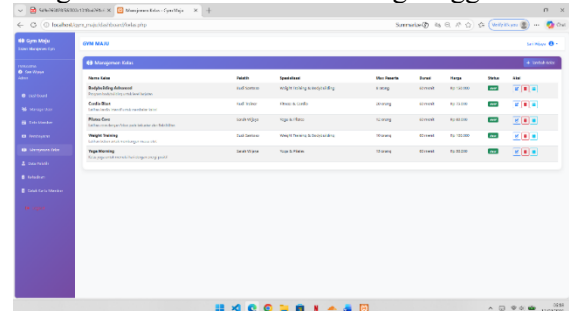
Gambar 10. Halaman Manajemen Member

Halaman Manajemen Member. Salah satu halaman inti dari sistem yang mengelola seluruh data keanggotaan. Admin dapat mendaftarkan *member* baru secara digital dengan mengisi formulir yang mencakup data pribadi, jenis keanggotaan, tanggal bergabung, dan tanggal berakhir masa aktif. Sistem secara otomatis membuat kode unik untuk setiap *member* baru dan memperbarui status keanggotaan menjadi *Expired* apabila masa berlaku sudah habis.



Gambar 11. Halaman Sistem Kasir

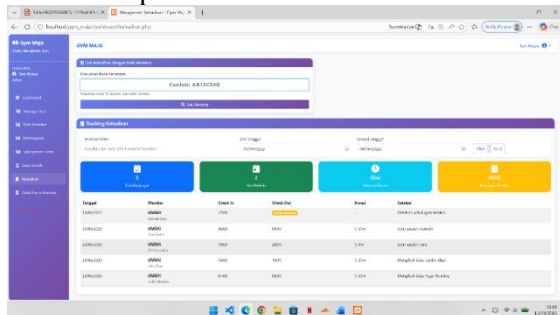
Halaman Sistem Kasir. Dirancang untuk mempermudah proses pencatatan pembayaran iuran *member*. Karyawan dapat memilih *member* yang akan melakukan pembayaran, memilih jenis layanan yang dibayarkan, serta menentukan metode pembayaran antara lain tunai (*cash*), transfer bank, maupun kartu (*card*). Setiap transaksi yang berhasil dicatat secara otomatis ke dalam *database* dan dapat dilihat kembali melalui riwayat pembayaran dengan filter berdasarkan rentang tanggal.



Gambar 12. Halaman Manajemen Kelas

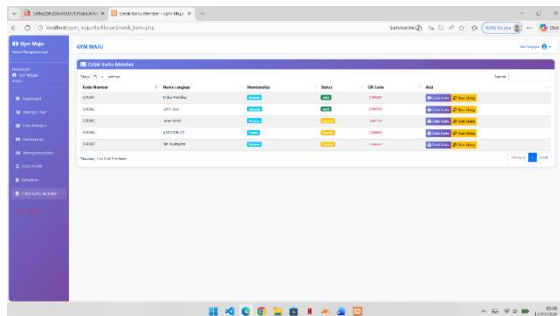
Halaman Manajemen Kelas. Memungkinkan Admin untuk mengatur jadwal kelas kebugaran yang tersedia, meliputi nama kelas, instruktur yang bertugas, kapasitas maksimal peserta, serta jadwal waktu pelaksanaan. Apabila jumlah pendaftar sudah mencapai batas kapasitas, sistem secara otomatis menutup

pendaftaran dan menampilkan pesan bahwa kelas telah penuh.



Gambar 13. Halaman Manajemen Kehadiran

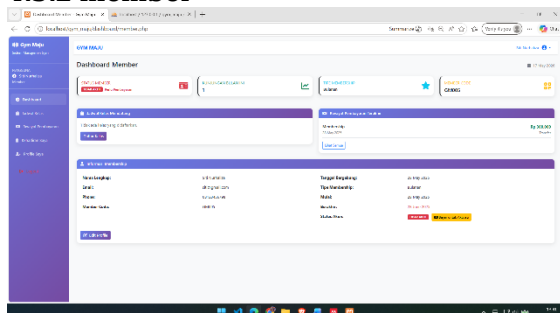
Halaman Manajemen Kehadiran. Berfungsi untuk mencatat dan memantau kehadiran *member* pada setiap sesi kelas. Data kehadiran terekam secara digital dan dapat difilter berdasarkan tanggal, nama kelas, maupun nama *member*, sehingga memudahkan Admin dalam memantau tingkat partisipasi anggota secara akurat.



Gambar 14. Halaman Cetak Kartu Member

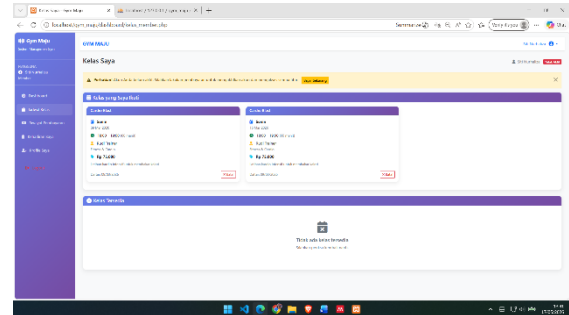
Halaman Cetak Kartu Member. Menyediakan fitur pembuatan kartu identitas *member* secara digital yang dilengkapi dengan kode QR (*QR code*) unik. Kartu *member* dapat dicetak langsung dari sistem dan digunakan sebagai alat verifikasi kehadiran pada setiap kunjungan ke gym, menggantikan proses pencatatan manual yang sebelumnya rentan terhadap kesalahan.

4.3.2 member



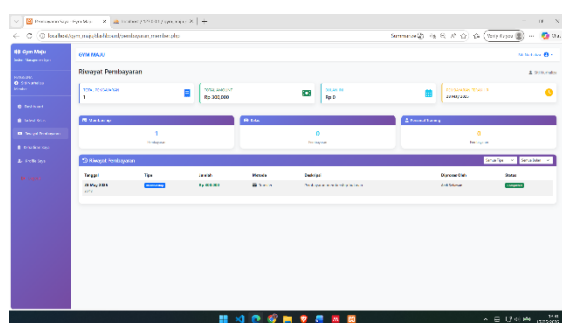
Gambar 15. Halaman Dashboard Member

Halaman Dashboard Member. Halaman ini merupakan tampilan utama yang diakses oleh *member* setelah berhasil login. *Dashboard* menampilkan empat informasi utama secara ringkas, yaitu status keanggotaan, jumlah kunjungan bulan ini, tipe *membership*, dan kode *member* unik. Di bawahnya terdapat informasi jadwal kelas mendatang yang telah didaftarkan serta riwayat pembayaran terakhir. Apabila masa keanggotaan *member* telah habis, sistem secara otomatis menampilkan status *Tidak Aktif* beserta tombol *Bayar untuk Aktivasi* sehingga *member* dapat langsung melakukan perpanjangan tanpa harus menghubungi Admin terlebih dahulu.



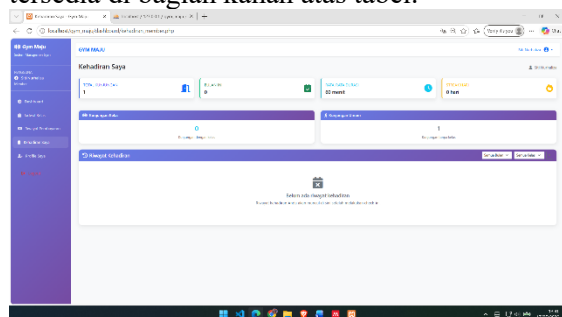
Gambar 16. Halaman jadwal kelas

Halaman Jadwal Kelas. Halaman ini menampilkan dua bagian utama, yaitu kelas yang sedang diikuti oleh *member* dan kelas yang tersedia untuk didaftarkan. Pada bagian kelas yang diikuti, *member* dapat melihat detail lengkap setiap kelas seperti nama kelas, hari pelaksanaan, jam, durasi, nama pelatih, spesialisasi, dan biaya kelas. *Member* juga dapat membatalkan pendaftaran kelas langsung dari halaman ini menggunakan tombol *Batal*. Apabila akun *member* dalam status tidak aktif, sistem menampilkan notifikasi peringatan di bagian atas halaman yang mengarahkan *member* untuk melakukan pembayaran terlebih dahulu sebelum dapat mengakses seluruh fitur kelas.



Gambar 17. Halaman riwayat pembayaran

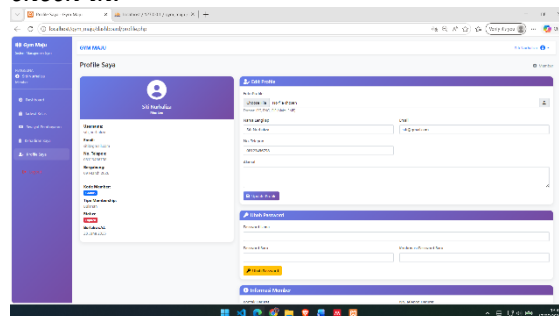
Halaman Riwayat Pembayaran. Halaman ini menyediakan rekap seluruh riwayat transaksi keuangan *member* secara personal. Terdapat empat kartu ringkasan di bagian atas yang menampilkan total pembayaran, total nominal, pembayaran bulan ini, dan tanggal pembayaran terakhir. Di bawahnya tersedia tabel riwayat pembayaran yang memuat informasi lengkap setiap transaksi meliputi tanggal, tipe pembayaran (*Membership*, *Kelas*, atau *Personal Training*), jumlah, metode pembayaran, deskripsi, nama staf yang memproses, serta status transaksi. *Member* dapat menyaring data transaksi berdasarkan tipe maupun bulan menggunakan filter yang tersedia di bagian kanan atas tabel.



Gambar 18. Halaman kehadiran

Halaman Kehadiran. Halaman ini menampilkan rekap data kehadiran *member* secara personal. Terdapat empat statistik utama di bagian atas, yaitu total kunjungan keseluruhan, jumlah kunjungan bulan ini, rata-rata durasi kunjungan, dan *streak* hari berturut-turut. Di bawah statistik, halaman dibagi menjadi dua bagian yaitu kunjungan dengan kelas dan kunjungan umum tanpa kelas. Riwayat kehadiran lengkap ditampilkan pada tabel di bawahnya dan dapat disaring berdasarkan bulan maupun nama kelas. Apabila *member* belum memiliki riwayat kehadiran, sistem menampilkan pesan

informatif yang memberitahu bahwa data akan muncul setelah *member* melakukan *check-in*.



Gambar 19. Halaman profil

Halaman Profil. Halaman ini memungkinkan *member* untuk mengelola data dirinya secara mandiri. Pada sisi kiri halaman ditampilkan informasi lengkap akun *member* meliputi *username*, email, nomor telepon, tanggal bergabung, kode *member*, tipe *membership*, status keanggotaan, dan tanggal berlaku *membership*. Apabila masa berlaku sudah habis, status ditampilkan dengan label *Expired* berwarna merah sebagai penanda visual yang jelas. Pada sisi kanan terdapat tiga formulir yang dapat diisi oleh *member*, yaitu formulir pembaruan foto profil dan data pribadi, formulir ubah *password*, serta formulir informasi kontak darurat. Seluruh perubahan data dapat disimpan langsung oleh *member* tanpa perlu bantuan Admin, sehingga meningkatkan kemandirian dan kenyamanan pengguna dalam mengelola akunnya.

4.4 Pengujian Sistem

Pengujian sistem dilakukan menggunakan metode *Black Box Testing* terhadap 22 skenario uji yang mencakup 5 modul utama. Ringkasan hasil pengujian disajikan pada Tabel 3 berikut.

No	Modul	Jumlah Skenario	Skenario Uji	Hasil
1	Login & Autentikasi	5	Login valid, password salah, akun tidak aktif, akses tanpa login, logout	Semua Pass
2	Pendaftaran Member	5	Tambah member, data duplikat, edit data, hapus data, status expired otomatis	Semua Pass

3	Pembayaran Iuran	4	Catat pembayaran <i>membership</i> , kelas tambahan, metode bayar, filter riwayat	Semua Pass
4	Pengelolaan Jadwal	4	Tambah kelas, edit & hapus kelas, kapasitas penuh, nonaktifkan kelas	Semua Pass
5	Laporan & Rekap	4	Statistik <i>dashboard</i> , filter kehadiran, cetak kartu member, buat ulang QR code	Semua Pass
Total	5 Modul	22 Skenario	-	100% Pass

Tabel 3. Ringkasan Hasil Pengujian *Black Box Testing*

Berdasarkan hasil pengujian pada Tabel 3, seluruh fungsionalitas sistem berjalan sesuai dengan yang diharapkan dan menghasilkan *output* yang valid. Tingkat keberhasilan 100% membuktikan bahwa sistem telah memenuhi seluruh kebutuhan fungsional yang telah dirancang.

4.5 Pembahasan

Sistem Informasi Manajemen Gym XYZ berhasil dibangun sebagai aplikasi berbasis *web* menggunakan PHP dan MySQL dengan menerapkan *Role-Based Access Control* (RBAC) yang membagi hak akses ke dalam dua peran pengguna, yaitu Admin dan Karyawan [6]. Admin memiliki hak akses untuk mengelola data pengguna dan data pelatih, sedangkan Karyawan mengelola data *member*, pembayaran, jadwal kelas, kehadiran, dan cetak kartu *member* berbasis QR code.

Jika dibandingkan dengan penelitian Hidayat & Permana (2022) tentang sistem manajemen gym berbasis *web* yang hanya mencakup pengelolaan *member* dan pembayaran tanpa fitur kehadiran berbasis QR code, sistem yang dibangun pada penelitian ini memiliki keunggulan berupa integrasi fitur QR code untuk verifikasi kehadiran *member* sehingga proses pencatatan menjadi lebih efisien dan akurat [7]. Selain itu, penelitian Kurniawan (2023) yang mengembangkan sistem serupa belum menerapkan *role-based access control* secara menyeluruh, sedangkan sistem ini telah membedakan hak akses secara spesifik antara

Admin dan Karyawan guna meningkatkan keamanan dan efisiensi pengelolaan data.

5. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan sebagai berikut.

- Sistem Informasi Manajemen Gym XYZ berhasil dirancang dan dibangun berbasis *web* menggunakan PHP dan MySQL dengan menerapkan metode pengembangan *Waterfall* dalam enam fase, yaitu *Requirements Analysis*, *System Design*, *Implementation*, *Testing*, *Deployment*, dan *Maintenance*.
- Sistem mencakup lima modul fungsional utama yang terintegrasi, meliputi *Login & Autentikasi*, *Pendaftaran Member*, *Pembayaran Iuran*, *Pengelolaan Jadwal Kelas*, serta *Laporan & Rekap*, yang seluruhnya dirancang untuk menyelesaikan permasalahan administrasi manual XYZ Fitness Center.
- Penerapan *Role-Based Access Control* (RBAC) berhasil membagi hak akses secara spesifik antara Admin dan Karyawan sehingga keamanan dan efisiensi pengelolaan data dapat terjaga dengan baik.
- Fitur unggulan berupa QR code pada kartu *member* menjadikan proses verifikasi kehadiran lebih cepat, akurat, dan modern dibandingkan sistem pencatatan manual sebelumnya.
- Berdasarkan hasil *Black Box Testing* terhadap 22 skenario uji pada 5 modul utama, seluruh fungsionalitas sistem berjalan sesuai yang diharapkan dengan tingkat keberhasilan 100%, membuktikan bahwa sistem layak digunakan dalam lingkungan operasional nyata.

Untuk penelitian selanjutnya, sistem ini dapat dikembangkan lebih lanjut dengan menambahkan fitur notifikasi otomatis bagi *member* yang masa keanggotaannya akan segera berakhir, integrasi *payment gateway* untuk mendukung pembayaran secara daring, serta pengembangan antarmuka berbasis *mobile* agar sistem dapat diakses lebih fleksibel melalui perangkat *smartphone*.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada seluruh pihak yang telah memberikan dukungan dalam penelitian ini, khususnya kepada Universitas Singaperbangsa Karawang dan pengelola XYZ Fitness Center yang telah memberikan izin serta kemudahan akses selama proses penelitian berlangsung.

DAFTAR PUSTAKA

- [1] F. A. Trinanda, N. C. Wibowo, and R. Permatasari, "PERANCANGAN WEBSITE SISTEM INFORMASI SI-PEGANG MENGGUNAKAN METODE WATERFALL PADA UPT DINAS LINGKUNGAN HIDUP JAWA TIMUR," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13,
- [2] A. Dennis and B. H. Wixom, "Tegarden, Systems Analysis and Design: An Object-Oriented Approach with UML," 2015.
- [3] A. Ginting, E. Prasista Bangun, K. Br Purba Universitas Mandiri Bina Prestasi Sri Wahyuni, E. Nurmalasari Pasaribu, and O. Bangun, "STUDI ANALISIS MODEL WATERFALL PADA PENGEMBANGAN SISTEM INFORMASI," vol. 5, p. 2026,
- [4] M. Ridwan and Z. Halim, "Perancangan Sistem Informasi Fasilitas Fitness Gym Berbasis Website Menggunakan Codeigniter," *Media Online*, vol. 4, no. 1, pp. 601–609, 2023, doi: 10.30865/klik.v4i1.1039.
- [5] V. M. Fastiana and D. Irfan, "Sistem Informasi Manajemen Data Berbasis Web di Start Gym & Fitness Lubuk Buaya Padang Sumatera Barat," 2021, [Online]. Available: <http://ejournal.unp.ac.id/index.php/voteknika/index>
- [6] V.] Rosalina and I. P. Sari, "Implementasi Role Based Access Control pada Sistem Informasi Manajemen Berbasis Web," 2022.
- [7] M. M. Fakhri *et al.*, "Perancangan Sistem Informasi Manajemen Karyawan Berbasis Website dengan Metode Waterfall," 2023.
- [8] Yunus Anisa, Eko Nur Wahyudib, and Harnanda Cahya Kurniawanc, "Metode Waterfall dalam Pengembangan Sistem Inventaris Guna Meningkatkan Efisiensi Manajemen Stok Barang," *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 6, no. 2, pp. 322–328, Apr. 2024, doi: 10.47233/jteksis.v6i2.1350.
- [9] Y. Sri Rahayu *et al.*, "IMPLEMENTASI METODE WATERFALL PADA PENGEMBANGAN SISTEM INFORMASI MOBILE E-DISARPUS," 2024.
- [10] N. Rizqiananda and C. Anwar, "Analisis Pengujian Perangkat Lunak pada Aplikasi CRUD Data Mahasiswa Berbasis PHP dan MySQL Menggunakan Black Box Testing 1," *journal.jci.co.id/jisbt*, vol. 1, no. 4, pp. 40–43, 2025, [Online]. Available: <https://journal.jci.co.id/jisbt>
- [11] "Rancang Bangun Sistem Informasi Pendaftaran Calon Member Komara Fitness Berbasis Web".
- [12] Siska Narulita, Ahmad Nugroho, and M. Zakki Abdillah, "Diagram Unified Modelling Language (UML) untuk Perancangan Sistem Informasi Manajemen Penelitian dan Pengabdian Masyarakat (SIMLITABMAS)," *Bridge : Jurnal publikasi Sistem Informasi dan Telekomunikasi*, vol. 2, no. 3, pp. 244–256, Aug. 2024, doi: 10.62951/bridge.v2i3.174.
- [13] K. Nistrina and L. Sahidah, "UNIFIED MODELLING LANGUAGE (UML) UNTUK PERANCANGAN SISTEM INFORMASI PENERIMAAN SISWA BARU DI SMK MARGA INSAN KAMIL."
- [14] S. M. Pulungan, R. Febrianti, T. Lestari, N. Gurning, and N. Fitriana, "Analisis Teknik Entity-Relationship Diagram Dalam Perancangan Database," vol. 01, no. 2, pp. 143–147, doi: 10.47233/jemb.v2i1.533.
- [15] M. Zen, Irwan, Hafni, and M. D. P. Ananda, "Implementasi dan Pengujian Menggunakan Metode BlackBox Testing Pada Sistem Informasi Tracer Study," *Bulletin of Computer Science Research*, vol. 4, no. 4, pp. 327–340, Jun. 2024, doi: 10.47065/bulletincsr.v4i4.359.