



Perancangan Backend Portal Informasi Komite Olahraga Nasional Indonesia Denpasar Berbasis MVC

Hisyam Akmal Maulana*, Pratomo Setiaji

Fakultas Teknik, Program Studi Sistem Informasi, Universitas Muria Kudus, Kudus, Indonesia

Email: ^{1,*}202353086@std.umk.ac.id, ²pratomo.setiaji@umk.ac.id

Email Penulis Korespondensi: 202353086@std.umk.ac.id

Abstrak—Perkembangan teknologi informasi mendorong berbagai organisasi untuk memanfaatkan sistem informasi berbasis website guna meningkatkan efektivitas pengelolaan data dan distribusi informasi secara digital. KONI Denpasar sebagai organisasi yang bergerak di bidang pembinaan olahraga daerah membutuhkan sistem backend yang mampu mendukung pengelolaan informasi secara terintegrasi karena proses pengelolaan data sebelumnya masih dilakukan secara konvensional dan belum terpusat. Kondisi tersebut menyebabkan penyampaian informasi organisasi menjadi kurang optimal serta menyulitkan proses administrasi dan pengembangan sistem secara berkelanjutan. Beberapa penelitian sebelumnya telah membahas pengembangan sistem berbasis framework CodeIgniter, namun sebagian besar penelitian masih berfokus pada sistem akademik, layanan administrasi, dan sistem penjualan sehingga penelitian mengenai backend portal informasi organisasi olahraga masih tergolong terbatas. Penelitian ini bertujuan untuk merancang dan membangun backend portal informasi KONI Denpasar berbasis website menggunakan framework CodeIgniter 4 dengan pendekatan Model View Controller (MVC). Metode pengembangan sistem yang digunakan adalah Systems Development Life Cycle (SDLC) dengan model Waterfall yang meliputi tahapan analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan sistem. Sistem backend yang dikembangkan memiliki fitur pengelolaan berita, agenda kegiatan, galeri dokumentasi, data pengguna, serta pengaturan hak akses administrator guna mendukung pengelolaan informasi organisasi secara lebih terstruktur. Pengujian sistem dilakukan menggunakan metode Black Box Testing untuk memastikan seluruh fitur dapat berjalan sesuai kebutuhan pengguna. Hasil penelitian menunjukkan bahwa backend portal informasi KONI Denpasar berhasil dikembangkan dengan baik dan seluruh fitur sistem dapat berfungsi sesuai hasil pengujian. Penelitian ini memberikan kontribusi dalam pengembangan backend portal informasi organisasi olahraga berbasis MVC yang terintegrasi sehingga mampu mendukung pengelolaan informasi secara lebih efektif, terstruktur, dan mudah dikembangkan pada masa mendatang.

Kata Kunci: Sistem Informasi; Backend; Website; CodeIgniter 4; MVC; SDLC

Abstract—The rapid development of information technology has encouraged many organizations to implement website-based information systems to improve the effectiveness of data management and digital information distribution. KONI Denpasar, as an organization engaged in regional sports development, requires a backend system capable of supporting integrated information management because the previous data management process was still carried out conventionally and was not centralized. This condition caused the dissemination of organizational information to become less optimal and made administrative processes and long-term system development more difficult. Several previous studies have discussed the development of systems using the CodeIgniter framework; however, most of these studies focused on academic systems, administrative services, and sales systems, while research related to backend information portals for sports organizations remains limited. Therefore, this study aims to design and develop a website-based backend information portal for KONI Denpasar using the CodeIgniter 4 framework with the Model View Controller (MVC) architectural approach. The system development method applied in this research is the Systems Development Life Cycle (SDLC) using the Waterfall model, which consists of requirements analysis, system design, implementation, testing, and maintenance stages. The developed backend system includes features for managing news, event agendas, documentation galleries, user data, and administrator access rights to support more structured information management. System testing was conducted using the Black Box Testing method to ensure that all features functioned according to user requirements. The results of this study indicate that the backend information portal for KONI Denpasar was successfully developed and that all system features operated properly based on the testing results. This research contributes to the development of an integrated MVC-based backend information portal for sports organizations that can support information management in a more effective, structured, and easily maintainable manner in the future.

Keywords: Information System; Backend; Website; CodeIgniter 4; MVC; SDLC

1. PENDAHULUAN

Perkembangan teknologi informasi saat ini telah membawa perubahan yang cukup signifikan dalam berbagai aktivitas organisasi, terutama dalam proses pengelolaan dan penyampaian informasi secara digital. Salah satu bentuk pemanfaatan teknologi tersebut adalah penggunaan sistem informasi berbasis website yang dinilai mampu membantu organisasi dalam mengelola data serta menyampaikan informasi secara lebih cepat, efektif, dan efisien. Menurut Mario Maryanto Bani et al. (2025), penerapan sistem berbasis online dengan memanfaatkan framework CodeIgniter dan metode Model View Controller (MVC) dapat menjadi solusi yang efektif untuk mendukung pengelolaan layanan informasi secara terintegrasi [1]. Sejalan dengan pendapat tersebut, Mita Tri Murtiningsih menyatakan bahwa penggunaan framework CodeIgniter pada sistem informasi berbasis web mampu meningkatkan efektivitas pengelolaan data sekaligus mempercepat proses pengembangan aplikasi [2].

KONI Denpasar sebagai organisasi yang bergerak di bidang pembinaan olahraga daerah membutuhkan sistem pengelolaan informasi yang terorganisasi dan terintegrasi dengan baik. Akan tetapi, proses pengelolaan informasi yang masih dilakukan secara konvensional menyebabkan penyampaian informasi kepada masyarakat belum berjalan secara optimal. Berbagai informasi penting seperti berita organisasi, agenda kegiatan, dokumentasi, hingga data pengguna masih belum dikelola dalam satu sistem terpusat. Kondisi tersebut mengakibatkan proses administrasi menjadi kurang efektif, sulit dikontrol, serta menyulitkan pengembangan sistem secara berkelanjutan di masa mendatang.



Beberapa penelitian sebelumnya menunjukkan bahwa framework CodeIgniter banyak dimanfaatkan dalam pengembangan sistem informasi berbasis website karena memiliki struktur yang terorganisasi dan mudah dikembangkan. Penelitian yang dilakukan oleh Alif Rahmat Yudha Putra dan Dedi Gunawan (2025) mengenai sistem rekrutmen karyawan berbasis website menggunakan CodeIgniter 4 menunjukkan bahwa framework tersebut mampu menghasilkan sistem yang lebih terstruktur dan fleksibel dalam proses pengembangannya [3]. Selain itu, penelitian yang dilakukan oleh Putri Anjilis Rohmah et al. (2025) menjelaskan bahwa penggunaan framework CodeIgniter pada layanan aduan tindak kekerasan berbasis multi-platform dapat meningkatkan efektivitas integrasi sistem serta mempermudah pengelolaan data secara menyeluruh [4].

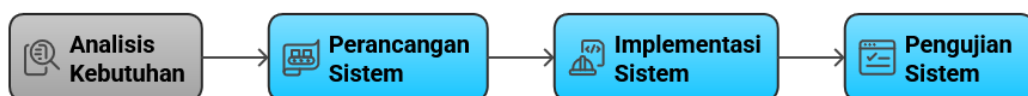
Penelitian lain yang dilakukan oleh Nelvi Mela Sari et al. (2025) mengenai pengembangan website PAUD berbasis CodeIgniter menunjukkan bahwa framework tersebut mampu mendukung pembangunan website yang lebih dinamis dan mudah diakses oleh pengguna [5]. Selanjutnya, Malik Maulana dan Verra Sofica (2024) menjelaskan bahwa penggunaan CodeIgniter dalam sistem informasi berbasis website dapat menghasilkan aplikasi yang lebih ringan, fleksibel, dan mudah dipelihara [6]. Penelitian yang dilakukan oleh Nova Selajar Da Costa et al. (2024) juga menunjukkan bahwa implementasi CodeIgniter 4 pada sistem penerimaan siswa baru berbasis web mampu meningkatkan efisiensi pengelolaan data serta mempermudah proses administrasi dalam sistem [7]. Walaupun berbagai penelitian terdahulu telah membahas penerapan framework CodeIgniter pada pengembangan sistem informasi berbasis website, sebagian besar penelitian masih berfokus pada bidang pendidikan, akademik, penjualan, maupun administrasi umum. Penelitian yang membahas pengembangan backend portal informasi pada organisasi olahraga, khususnya KONI Denpasar, masih sangat sedikit ditemukan. Selain itu, sebagian besar penelitian sebelumnya lebih banyak menekankan pada pengembangan sisi frontend, sedangkan pembahasan mengenai backend terintegrasi yang mendukung pengelolaan berita, agenda kegiatan, galeri dokumentasi, pengaturan hak akses pengguna, dan monitoring aktivitas administrator dalam satu platform masih belum banyak dilakukan.

Berdasarkan permasalahan tersebut, penelitian ini bertujuan untuk merancang dan membangun backend portal informasi KONI Denpasar berbasis website menggunakan framework CodeIgniter 4 dengan pendekatan arsitektur Model View Controller (MVC). Untuk itu, penelitian ini diharapkan mampu memberikan solusi dalam meningkatkan efektivitas pengelolaan informasi organisasi secara lebih terstruktur, terintegrasi, dan mudah dikembangkan pada masa mendatang. Selain itu, penelitian ini juga memberikan kontribusi pada pengembangan backend portal informasi organisasi olahraga berbasis MVC yang masih jarang dibahas pada penelitian sebelumnya.

2. METODOLOGI PENELITIAN

Metode penelitian yang digunakan dalam pengembangan sistem ini mengacu pada Software Development Life Cycle (SDLC) dengan model Waterfall sebagai metode pengembangan sistem [8]. Model Waterfall dipilih karena memiliki alur kerja yang terstruktur dan sistematis sehingga setiap tahapan pengembangan dapat dilakukan secara berurutan sesuai kebutuhan sistem yang telah ditentukan sebelumnya [9]. Pendekatan ini dinilai sesuai untuk pengembangan backend portal informasi karena proses pengembangannya memerlukan tahapan yang jelas mulai dari analisis hingga pengujian sistem. Dalam penelitian ini, tahapan pengembangan sistem meliputi analisis kebutuhan, perancangan sistem, implementasi, dan pengujian. Tahap analisis kebutuhan dilakukan untuk mengidentifikasi kebutuhan pengguna serta menentukan spesifikasi sistem yang akan dibangun agar pengembangan berjalan sesuai tujuan penelitian [4]. Selanjutnya, tahap perancangan sistem dilakukan untuk menyusun struktur sistem backend, alur proses, serta rancangan basis data yang akan digunakan. Tahap implementasi dilakukan dengan menerapkan framework CodeIgniter 4 menggunakan pendekatan Model View Controller (MVC) agar sistem lebih terstruktur dan mudah dikembangkan. Setelah implementasi selesai, tahap pengujian dilakukan untuk memastikan seluruh fitur sistem dapat berjalan dengan baik sesuai kebutuhan pengguna [10].

Setelah proses analisis selesai dilakukan, tahap berikutnya adalah perancangan sistem yang mencakup penyusunan arsitektur aplikasi, desain antarmuka pengguna, serta perancangan basis data sesuai kebutuhan sistem [6]. Tahap selanjutnya adalah implementasi, yaitu proses menerjemahkan rancangan sistem ke dalam bentuk kode program menggunakan bahasa pemrograman yang telah ditentukan. Pada tahap ini, seluruh komponen sistem mulai dikembangkan dan diintegrasikan berdasarkan desain yang telah dibuat sebelumnya [11]. Setelah sistem selesai dibangun, dilakukan tahap pengujian untuk memastikan bahwa aplikasi dapat berjalan dengan baik, meminimalkan kesalahan sistem, serta mampu memenuhi kebutuhan pengguna sesuai tujuan pengembangan yang telah ditetapkan sejak awal. Pada Gambar 1 di bawah ini merupakan alur atau tahapan dalam membangun sistem ini.



Gambar 1. Model Waterfall Sistem

Penelitian ini mengadopsi empat tahap utama dalam model Waterfall, yang divisualisasikan pada Gambar 1, dan akan dijelaskan secara lebih rinci untuk memberikan gambaran proses pengembangan sistem.



2.1 Analisis Kebutuhan

Tahap analisis kebutuhan merupakan tahapan awal dalam proses pengembangan sistem yang bertujuan mengidentifikasi kebutuhan pengguna terhadap sistem yang akan dibangun. Pada tahap ini dilakukan pengumpulan informasi terkait alur bisnis yang sedang berjalan, permasalahan yang ditemukan, serta kebutuhan yang perlu dipenuhi oleh sistem [12]. Analisis kebutuhan mencakup kebutuhan fungsional yang berkaitan dengan fitur utama sistem, serta kebutuhan nonfungsional seperti aspek keamanan, performa, dan kemudahan penggunaan. Hasil dari proses analisis tersebut selanjutnya dijadikan dasar dalam penyusunan spesifikasi sistem yang akan dikembangkan [13].

2.2 Perancangan Sistem

Tahap perancangan sistem dilakukan setelah proses analisis kebutuhan selesai dilakukan. Pada tahap ini, perancangan difokuskan pada penyusunan struktur dan alur kerja sistem yang akan dikembangkan. Untuk mempermudah visualisasi rancangan sistem, digunakan beberapa model Unified Modeling Language (UML) agar desain sistem dapat dipahami secara lebih jelas dan terstruktur [7]. Perancangan sistem bertujuan untuk memberikan gambaran mengenai mekanisme kerja aplikasi yang akan dikembangkan sehingga proses implementasi dapat dilakukan secara lebih terarah dan sistematis.

a. Use Case Diagram

Use Case Diagram digunakan untuk menggambarkan interaksi antara pengguna dengan sistem yang dikembangkan.

Diagram ini menunjukkan hak akses pengguna terhadap fitur-fitur utama sistem backend seperti login, pengelolaan berita, agenda kegiatan, galeri dokumentasi, data cabang olahraga, serta manajemen pengguna. Melalui use case diagram, alur interaksi pengguna dengan sistem dapat dipahami secara lebih jelas [14].

b. Activity Diagram dan Entity Relationship Diagram (ERD)

Activity Diagram digunakan untuk menggambarkan alur aktivitas maupun proses kerja yang berlangsung pada sistem secara berurutan sehingga mempermudah pemahaman terhadap proses bisnis yang berjalan pada aplikasi [15]. Selain itu, Entity Relationship Diagram (ERD) digunakan dalam proses perancangan basis data untuk menunjukkan struktur data serta hubungan antarentitas yang terdapat pada sistem [16]. Melalui Activity Diagram dan ERD, pengembang dapat memahami alur sistem serta struktur penyimpanan data secara lebih terorganisasi sehingga proses implementasi sistem dapat dilakukan dengan lebih efektif dan terarah.

c. Flowchart Sistem

Flowchart sistem digunakan untuk menggambarkan alur proses kerja aplikasi mulai dari proses autentikasi pengguna hingga pengelolaan data pada sistem backend. Diagram ini membantu menjelaskan urutan proses yang terjadi pada sistem sehingga mempermudah pengembang dalam memahami alur logika aplikasi [15].

d. Perancangan Basis Data

Perancangan basis data dilakukan untuk menentukan struktur tabel, relasi antar data, serta pengelolaan penyimpanan informasi pada sistem. Basis data digunakan untuk menyimpan seluruh data yang berkaitan dengan berita, agenda kegiatan, galeri dokumentasi, data pengguna, serta informasi organisasi lainnya.

e. Perancangan Antarmuka

Perancangan antarmuka dilakukan untuk menghasilkan tampilan sistem yang mudah digunakan dan dipahami oleh administrator. Antarmuka sistem dirancang secara sederhana, terstruktur, dan responsif agar proses pengelolaan informasi dapat dilakukan secara lebih efektif.

Melalui proses perancangan ini, pengembang dapat memperoleh gambaran yang jelas mengenai struktur sistem, hubungan antar komponen, serta mekanisme kerja sistem yang akan dibangun [17].

2.2.1 Model View Controller (MVC)

Model View Controller (MVC) merupakan pola arsitektur perangkat lunak yang digunakan untuk memisahkan komponen aplikasi menjadi tiga bagian utama, yaitu Model, View, dan Controller. Menurut Mario Maryanto Bani et al. (2025), penerapan MVC bertujuan untuk menghasilkan struktur aplikasi yang lebih modular, terorganisasi, dan mudah dikembangkan [1]. Model berfungsi untuk mengelola data dan berinteraksi langsung dengan basis data. Komponen ini bertanggung jawab terhadap proses penyimpanan, pengambilan, pengubahan, dan penghapusan data pada sistem. View berfungsi sebagai bagian antarmuka pengguna yang digunakan untuk menampilkan informasi kepada pengguna sistem. Sementara itu, Controller bertugas mengatur alur logika aplikasi dan menjadi penghubung antara Model dan View. Pada penelitian ini, konsep MVC diterapkan menggunakan framework CodeIgniter 4 untuk menghasilkan struktur backend yang lebih terorganisasi dan mudah dipelihara. Dengan adanya pemisahan fungsi pada setiap komponen, proses pengembangan fitur baru dapat dilakukan tanpa memengaruhi bagian sistem lainnya. Selain itu, penerapan MVC juga membantu meningkatkan efisiensi pemeliharaan sistem dan mempermudah proses debugging aplikasi.

2.3 Implementasi Sistem

Tahap implementasi menjadi proses lanjutan setelah rancangan sistem selesai disusun, yaitu dengan mengubah desain yang telah dibuat ke dalam bentuk aplikasi yang dapat digunakan secara langsung. Pada penelitian ini, proses implementasi dilakukan melalui pengembangan aplikasi berbasis web dengan memanfaatkan beberapa teknologi pendukung, seperti PHP sebagai bahasa pemrograman utama, CodeIgniter 4 sebagai framework pengembangan, MySQL untuk pengelolaan basis data, serta HTML, CSS, dan JavaScript sebagai komponen dalam membangun tampilan antarmuka pengguna [18]. Kombinasi teknologi tersebut digunakan untuk menghasilkan sistem yang mampu berjalan



secara optimal serta mendukung kebutuhan pengelolaan informasi pada website portal KONI Denpasar. Pemanfaatan framework CodeIgniter 4 dipilih karena framework ini memiliki performa yang ringan dengan struktur pengembangan yang lebih sederhana dan mudah dipahami [4]. Selain itu, CodeIgniter 4 mendukung penerapan konsep Model View Controller (MVC) yang memungkinkan pembagian fungsi sistem menjadi lebih terstruktur. Dalam konsep tersebut, bagian model berfungsi untuk mengelola data, controller bertugas mengatur logika aplikasi, sedangkan view digunakan untuk menangani tampilan antarmuka pengguna. Adanya pemisahan fungsi tersebut membuat proses pengembangan aplikasi menjadi lebih teratur, memudahkan pengembang dalam melakukan pemeliharaan sistem, serta mendukung pengembangan fitur tambahan di masa mendatang tanpa mengganggu komponen sistem lainnya.

Metode Model View Controller (MVC) digunakan sebagai pendekatan arsitektur dalam pengembangan sistem backend portal informasi. MVC membagi struktur aplikasi menjadi tiga komponen utama, yaitu Model yang bertugas mengelola data dan basis data, View yang berfungsi menampilkan antarmuka pengguna, serta Controller yang bertanggung jawab mengatur logika aplikasi dan menghubungkan Model dengan View. Penerapan MVC pada penelitian ini bertujuan untuk menghasilkan struktur kode program yang lebih terorganisasi, mudah dikembangkan, serta mempermudah proses pemeliharaan sistem. Dengan adanya pemisahan fungsi tersebut, proses pengembangan fitur baru dapat dilakukan tanpa memengaruhi komponen sistem lainnya.

2.4 Pengujian Sistem

Tahap pengujian sistem dilakukan sebagai langkah untuk mengetahui apakah aplikasi yang telah dikembangkan sudah mampu berjalan sesuai dengan kebutuhan pengguna dan spesifikasi sistem yang telah ditentukan sebelumnya. Dalam penelitian ini, metode pengujian yang digunakan adalah Blackbox Testing, yaitu teknik pengujian yang berfokus pada fungsi dan hasil keluaran sistem tanpa melakukan pemeriksaan terhadap struktur maupun kode program secara langsung [19]. Dengan metode ini, pengujian dilakukan dari sisi pengguna untuk memastikan setiap fitur dapat berfungsi sebagaimana mestinya. Proses pengujian dilakukan dengan menjalankan berbagai skenario penggunaan pada setiap fitur sistem untuk memastikan respons dan keluaran yang sesuai. Pengujian juga mengidentifikasi kesalahan atau kekurangan, sehingga kualitas dan kinerja aplikasi dapat diketahui dan sistem siap untuk digunakan dalam kegiatan operasional [20].

3. HASIL DAN PEMBAHASAN

3.1 Perancangan Sistem

Perancangan sistem menjelaskan struktur dan alur proses kerja portal informasi untuk pengembangan terarah.

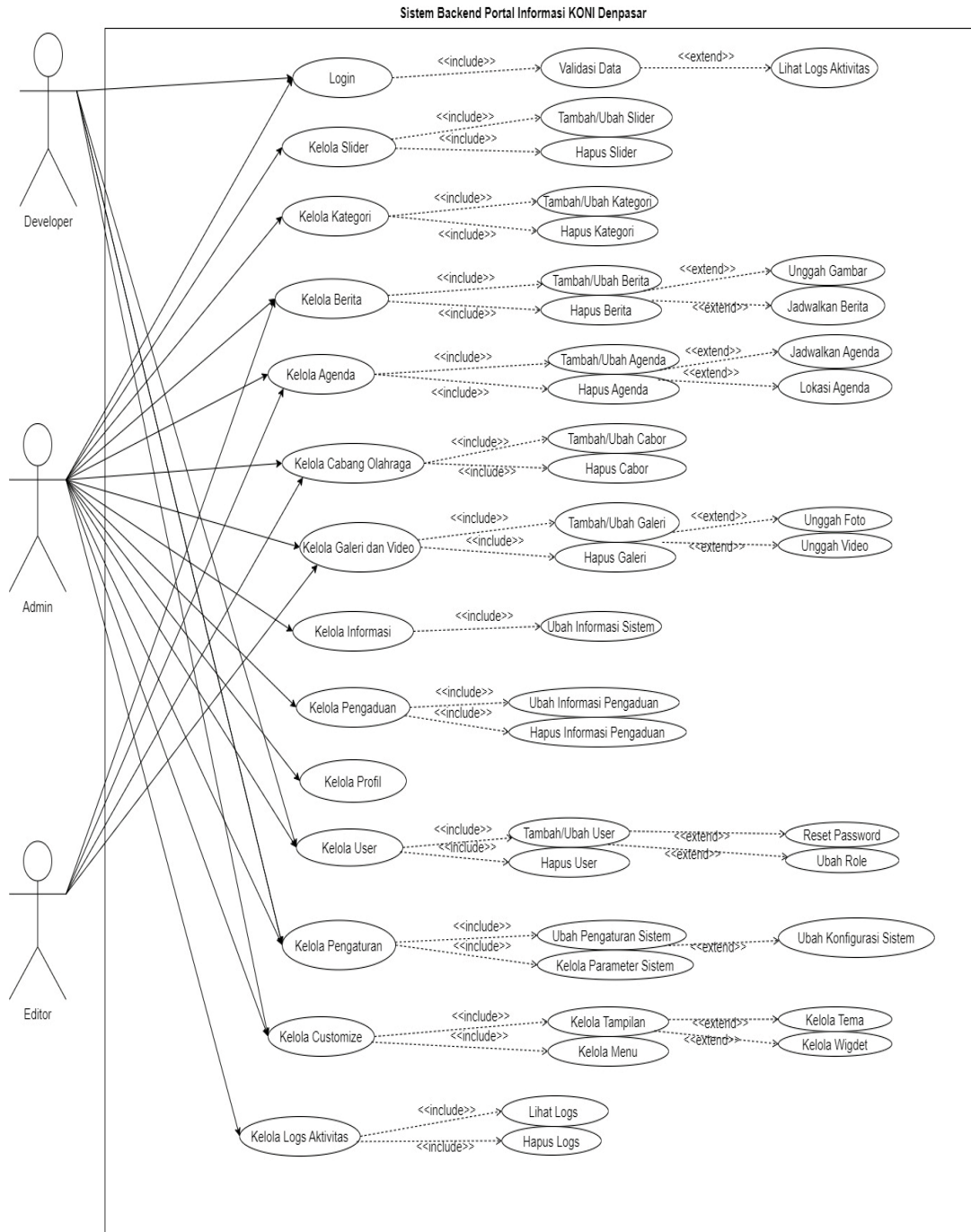
3.1.1 Use Case Diagram

Sebagai bagian dari perancangan sistem, Use Case Diagram digunakan untuk menggambarkan interaksi antara aktor dengan sistem yang dikembangkan. Adapun Use Case Diagram pada penelitian ini ditunjukkan pada Gambar 2.

Berdasarkan *Use Case Diagram* yang telah dirancang seperti pada Gambar 2, sistem *backend* Portal Informasi KONI Denpasar melibatkan tiga aktor utama, yakni Developer, Admin, dan Editor, yang masing-masing memiliki fungsi, tanggung jawab, serta tingkat hak akses yang berbeda sesuai dengan kebutuhan operasional sistem. Pembagian peran ini tidak semata-mata untuk membatasi akses pengguna terhadap fitur tertentu, tetapi juga untuk membangun alur kerja yang lebih sistematis, efisien, dan aman melalui penerapan konsep *role-based access control*. Dengan pendekatan tersebut, setiap aktor hanya diberikan akses terhadap bagian sistem yang relevan dengan tugasnya, sehingga potensi kesalahan penggunaan maupun penyalahgunaan hak akses dapat ditekan seminimal mungkin. Developer bertindak sebagai pihak yang memiliki kendali penuh terhadap aspek teknis sistem, meliputi pengelolaan data pengguna, konfigurasi sistem, penyesuaian tampilan antarmuka, pengelolaan laporan, serta pemantauan aktivitas melalui logs yang berfungsi untuk mendeteksi kesalahan dan aktivitas yang tidak wajar.

Sementara itu, Admin memiliki cakupan kewenangan yang luas dalam pengelolaan data dan konten sistem. Tanggung jawab Admin mencakup pengelolaan berbagai komponen utama seperti slider, kategori, berita, agenda kegiatan, cabang olahraga, galeri (foto dan video), informasi publik, pengaduan masyarakat, profil organisasi, data pengguna, hingga pengaturan sistem, laporan, penyesuaian tampilan, dan akses terhadap logs aktivitas. Di sisi lain, Editor memiliki ruang lingkup tugas yang lebih terbatas dan terfokus pada pengelolaan konten yang akan dipublikasikan. Peran Editor mencakup pengelolaan berita, agenda, galeri, informasi, serta pengaduan, tanpa memiliki akses terhadap pengaturan sistem maupun manajemen pengguna, sehingga fungsi utamanya lebih diarahkan pada operasional konten secara langsung.

Dari perspektif analisis sistem, alur proses dimulai dari mekanisme login yang terintegrasi dengan proses validasi data sebagai bagian dari relasi `<<include>>`, yang menandakan bahwa setiap pengguna harus melewati tahap autentikasi sebelum dapat mengakses fitur sistem. Selanjutnya, seluruh aktivitas pengelolaan data, seperti pengelolaan berita, agenda, kategori, galeri, dan pengguna, didasarkan pada operasi CRUD (*Create, Read, Update, Delete*) sebagai fondasi utama dalam pengolahan data. Hal ini menunjukkan bahwa sistem dirancang dengan pendekatan yang terstandarisasi dan konsisten dalam menangani berbagai jenis data. Selain itu, terdapat relasi `<<extend>>` yang merepresentasikan fitur tambahan yang bersifat opsional dan hanya digunakan dalam kondisi tertentu, seperti unggah gambar dan penjadwalan publikasi pada berita, pengaturan jadwal dan lokasi pada agenda, unggah media pada galeri, serta fitur reset *password* dan pengaturan peran dalam manajemen pengguna.

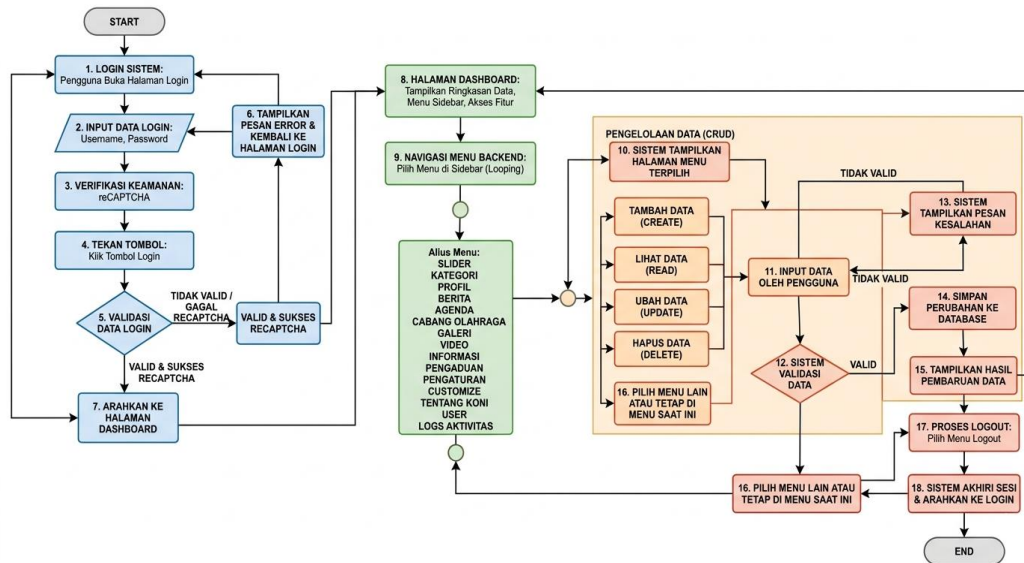


Gambar 2. Use Case Diagram Sistem Backend

Pada aspek pengelolaan sistem, disediakan pula fitur tambahan seperti konfigurasi lanjutan, penyaringan data, serta ekspor laporan untuk mendukung kebutuhan analisis dan pelaporan. Kehadiran relasi `<<extend>>` ini menunjukkan bahwa sistem memiliki fleksibilitas yang tinggi dalam mengakomodasi berbagai kebutuhan pengguna tanpa mengganggu fungsi utama yang sudah ada. Secara keseluruhan, struktur *Use Case Diagram* tersebut mencerminkan bahwa sistem *backend* dikembangkan dengan pendekatan modular, di mana setiap komponen memiliki fungsi yang terpisah namun tetap saling terintegrasi dalam satu kesatuan sistem. Dengan perancangan yang terstruktur dan sistematis ini, sistem tidak hanya mampu mendukung pengelolaan data dan konten secara optimal, tetapi juga meningkatkan keamanan melalui pembagian hak akses yang jelas, serta mempermudah proses pengembangan, pengelolaan, dan pemeliharaan sistem di masa mendatang.

3.1.2 Flowchart Sistem

Flowchart digunakan untuk memvisualisasikan alur kerja pada sistem *backend* yang dioperasikan oleh administrator dalam mengelola data pada website portal informasi KONI Denpasar. Gambar 5 memperlihatkan rangkaian proses dalam sistem yang diawali dengan autentikasi pengguna dan dilanjutkan dengan pengelolaan berbagai data serta fitur pada sistem backend.



Gambar 3. Flowchart Sistem Backend

Berdasarkan flowchart backend yang ditunjukkan pada Gambar 3, proses sistem dimulai dari tahap login yang dilakukan oleh pengguna dengan hak akses sebagai administrator, developer, maupun editor. Pada tahap ini, pengguna terlebih dahulu membuka halaman backend dengan memasukkan username dan password yang sesuai, kemudian melakukan verifikasi melalui reCAPTCHA sebelum menekan tombol login. Setelah itu, sistem akan memeriksa dan memvalidasi data yang telah dimasukkan. Jika data dinyatakan benar, pengguna akan diarahkan menuju halaman dashboard. Namun, apabila data tidak valid, sistem akan menampilkan notifikasi kesalahan dan mengembalikan pengguna ke halaman login untuk melakukan input ulang.

Setelah proses autentikasi berhasil dilakukan, pengguna akan masuk ke halaman dashboard yang berfungsi sebagai pusat pengelolaan sistem. Halaman tersebut menampilkan ringkasan informasi serta menyediakan berbagai menu navigasi untuk mengakses fitur-fitur yang tersedia. Navigasi sistem dilakukan melalui sidebar yang memuat beberapa menu utama, seperti slider, berita, agenda, galeri, dan fitur lainnya. Melalui menu-menu tersebut, pengguna dapat melakukan pengelolaan data menggunakan fungsi Create, Read, Update, dan Delete (CRUD) yang menjadi bagian utama dalam operasional sistem informasi.

Setiap halaman dalam sistem dirancang dengan fungsi yang berbeda sesuai kebutuhan pengelolaan data. Halaman slider digunakan untuk mengatur tampilan banner pada website, halaman berita untuk mengelola konten informasi, halaman agenda untuk mengatur jadwal kegiatan, halaman user untuk manajemen pengguna, serta halaman logs yang berfungsi memantau aktivitas sistem. Pembagian fungsi ini menunjukkan bahwa sistem dikembangkan dengan pendekatan modular, sehingga setiap komponen memiliki peran yang jelas.

Dalam proses pengolahan data, setiap input yang dimasukkan pengguna akan melalui tahap validasi sebelum disimpan ke dalam basis data. Jika data telah memenuhi kriteria yang ditentukan, maka sistem akan menyimpannya ke dalam database. Namun, apabila data tidak sesuai, sistem akan memberikan pesan kesalahan sebagai bentuk umpan balik kepada pengguna. Setelah seluruh aktivitas selesai, pengguna dapat mengakhiri sesi dengan melakukan logout, di mana sistem akan menutup sesi dan mengarahkan kembali ke halaman login.

Berdasarkan alur tersebut, dapat disimpulkan bahwa sistem *backend* yang dirancang memiliki mekanisme kerja yang terstruktur, mulai dari proses autentikasi, pengelolaan data, hingga pengakhiran sesi. Alur ini menunjukkan bahwa sistem telah dirancang untuk mendukung aspek keamanan, kemudahan dalam pengelolaan, serta integrasi data yang efektif dalam sistem informasi berbasis web.

3.2 Implementasi Sistem

Tahapan implementasi sistem dilakukan sebagai bentuk realisasi dari hasil analisis dan perancangan yang telah disusun sebelumnya. Pada penelitian ini, implementasi diwujudkan melalui pembangunan Website Portal Informasi KONI Denpasar berbasis web dengan memanfaatkan framework CodeIgniter 4 sebagai dasar pengembangan aplikasi. Framework tersebut dikembangkan dengan pendekatan arsitektur Model View Controller (MVC) yang memungkinkan proses pengembangan sistem dilakukan secara lebih terstruktur dan terorganisasi.

Pemilihan CodeIgniter 4 didasarkan pada kemampuannya dalam mendukung proses pengembangan aplikasi web secara lebih cepat, ringan, dan efisien. Selain itu, framework ini menyediakan struktur kerja yang memudahkan pengembang dalam mengatur alur program sehingga kode yang dihasilkan menjadi lebih rapi dan mudah dipahami. Penerapan konsep MVC juga memberikan pembagian tugas yang jelas antara bagian pengelolaan data, pengaturan logika aplikasi, serta tampilan antarmuka pengguna. Dengan struktur tersebut, proses pemeliharaan sistem maupun pengembangan fitur tambahan pada tahap selanjutnya dapat dilakukan dengan lebih mudah tanpa mengganggu fungsi utama aplikasi yang telah berjalan.



3.2.1 Lingkungan Implementasi

Sistem yang dikembangkan dalam penelitian ini memanfaatkan berbagai perangkat lunak dan teknologi yang saling terintegrasi guna menunjang proses pembangunan aplikasi secara optimal. Dari sisi bahasa pemrograman, sistem dibangun menggunakan PHP yang berperan sebagai fondasi dalam pengembangan logika aplikasi serta pengelolaan proses di sisi server. Pemilihan PHP didasarkan pada kemampuannya dalam mengembangkan aplikasi web yang bersifat dinamis serta tingkat kompatibilitasnya yang tinggi dengan berbagai teknologi pendukung.

Dalam tahap pengembangan, digunakan framework CodeIgniter 4 yang mengimplementasikan arsitektur Model View Controller (MVC). Framework ini dipilih karena memiliki struktur yang sederhana namun terorganisasi dengan baik, sehingga memudahkan dalam pengelolaan kode serta meningkatkan efisiensi proses pengembangan aplikasi. Selain itu, penerapan framework ini memungkinkan adanya pemisahan yang jelas antara komponen logika aplikasi, pengolahan data, dan tampilan antarmuka.

Pada aspek pengelolaan data, sistem menggunakan MySQL sebagai basis data yang berfungsi untuk menyimpan data secara terstruktur. MySQL dipilih karena kemampuannya dalam menangani data dalam jumlah besar serta mendukung proses manipulasi data secara efisien. Untuk menjalankan aplikasi selama tahap pengembangan, digunakan web server Apache yang tersedia dalam paket seperti Laragon maupun XAMPP. Penggunaan paket tersebut memberikan kemudahan dalam proses konfigurasi serta pengujian sistem secara lokal sebelum diterapkan secara penuh.

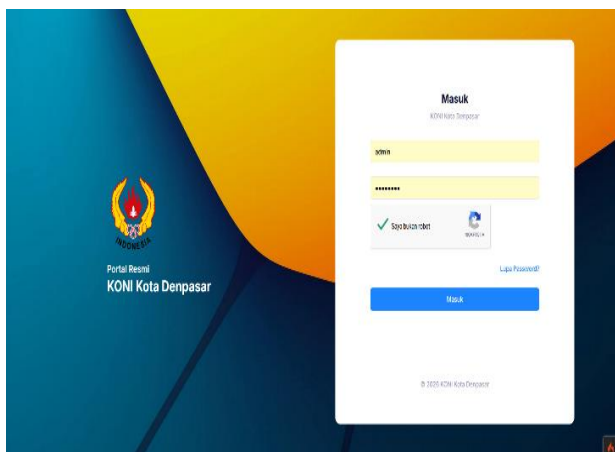
Dalam proses pengembangan, Visual Studio Code digunakan sebagai alat utama untuk penulisan kode. Editor ini dipilih karena menyediakan berbagai fitur pendukung, seperti ekstensi tambahan, fasilitas debugging, serta kemudahan dalam pengelolaan proyek yang dapat meningkatkan produktivitas pengembang. Secara keseluruhan, pemilihan kombinasi teknologi tersebut didasarkan pada pertimbangan bahwa seluruhnya bersifat *open source*, sehingga tidak memerlukan biaya lisensi, mudah dipelajari, serta didukung oleh komunitas yang luas. Selain itu, teknologi yang digunakan juga mampu mendukung pengembangan aplikasi web yang dinamis, fleksibel, dan mudah dikembangkan di masa mendatang.

3.2.2 Implementasi antarmuka Sistem

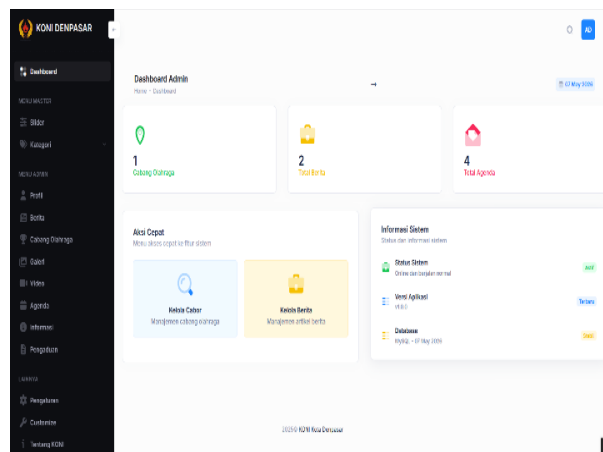
Penerapan arsitektur Model View Controller (MVC) pada sistem backend Portal Informasi Komite Olahraga Nasional Indonesia Denpasar dilakukan untuk membagi pengelolaan data, proses logika aplikasi, serta tampilan antarmuka pengguna secara lebih terstruktur. Dalam implementasinya, bagian Model berfungsi untuk menangani seluruh proses yang berkaitan dengan basis data, seperti menambah, mengubah, menghapus, dan mengambil data berita, agenda kegiatan, galeri, pengguna, serta berbagai data lainnya. Pengelolaan data yang dipusatkan pada model membuat struktur basis data menjadi lebih rapi, terorganisasi, dan memudahkan proses pemeliharaan sistem.

Pada bagian Controller, komponen ini berperan sebagai penghubung antara Model dan View dengan mengatur alur logika aplikasi serta proses validasi data yang diberikan pengguna. Controller menerima permintaan dari pengguna, kemudian memproses data melalui model sebelum hasilnya ditampilkan kembali pada antarmuka sistem. Dalam penelitian ini, controller digunakan pada proses login pengguna, pengelolaan berita, pengaturan hak akses, pengelolaan agenda kegiatan, serta konfigurasi sistem backend lainnya.

Sementara itu, komponen View digunakan untuk menampilkan antarmuka backend dalam bentuk halaman website yang dapat diakses oleh administrator. Tampilan sistem dirancang secara sederhana dan mudah dipahami agar proses pengelolaan data dapat dilakukan dengan lebih efektif. Melalui penerapan konsep MVC, struktur sistem menjadi lebih modular dan terorganisasi sehingga memudahkan proses pengembangan maupun penambahan fitur baru di masa mendatang tanpa mengganggu bagian utama aplikasi.



Gambar 4. Halaman Login



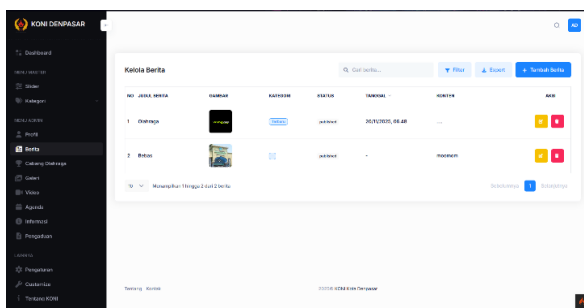
Gambar 5. Halaman Dashboard

Gambar 4 menunjukkan halaman login sebagai tampilan awal pada sistem backend Portal Informasi Komite Olahraga Nasional Indonesia Denpasar yang digunakan untuk proses autentikasi pengguna sebelum mengakses sistem. Pengguna diwajibkan memasukkan username dan password yang telah terdaftar serta melakukan verifikasi reCAPTCHA

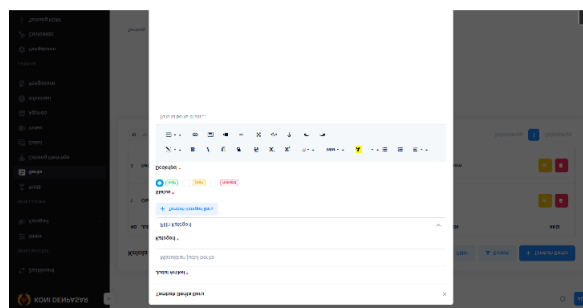
sebagai bentuk keamanan tambahan. Setelah berhasil login, pengguna akan diarahkan ke halaman dashboard seperti yang ditunjukkan pada Gambar 5. Halaman dashboard berfungsi sebagai pusat pengelolaan sistem yang menampilkan ringkasan informasi penting seperti jumlah cabang olahraga, berita, dan agenda, serta menyediakan akses cepat menuju fitur pengelolaan data melalui menu navigasi yang tersedia. Tampilan dashboard dirancang sederhana dan terstruktur agar memudahkan administrator dalam memantau dan mengelola sistem secara lebih efektif.

Pada sistem backend Portal Informasi Komite Olahraga Nasional Indonesia Denpasar terdapat sidebar navigasi yang digunakan sebagai menu utama untuk mengakses berbagai fitur pengelolaan data. Sidebar dirancang secara terstruktur agar memudahkan pengguna dalam berpindah halaman dan mengelola informasi pada sistem. Menu yang tersedia meliputi dashboard, berita, agenda, galeri, video, pengaduan, pengaturan sistem, manajemen pengguna, dan monitoring aktivitas. Namun, pada penelitian ini pembahasan difokuskan pada fitur utama yang mendukung pengelolaan informasi portal, seperti dashboard, berita, agenda, galeri, dan manajemen pengguna.

Berdasarkan Gambar 6, halaman CRUD berita digunakan untuk mengelola data berita melalui proses Create, Read, Update, dan Delete (CRUD). Halaman ini menampilkan tabel berita yang memuat informasi seperti judul, kategori, status publikasi, tanggal, dan penulis berita, serta dilengkapi fitur pencarian dan tombol aksi edit maupun hapus data. Selanjutnya, pada Gambar 7 ditunjukkan form tambah berita yang digunakan untuk memasukkan data berita baru ke dalam sistem. Form tersebut menyediakan input judul, kategori, isi berita, status publikasi, tanggal publikasi, dan unggahan gambar thumbnail sehingga administrator dapat mengelola dan mempublikasikan informasi secara lebih mudah, cepat, dan terorganisasi. Selain itu, pada halaman cabang olahraga dan halaman lainnya juga mempunyai proses Create, Read, Update, dan Delete (CRUD) punya form inputan tersendiri.

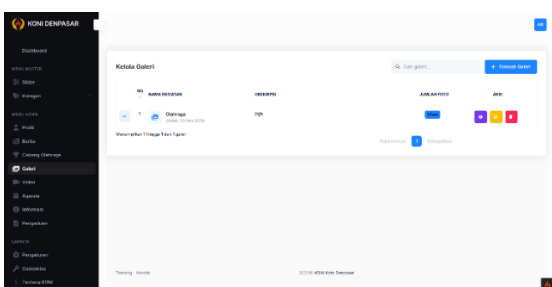


Gambar 6. Halaman CRUD Berita

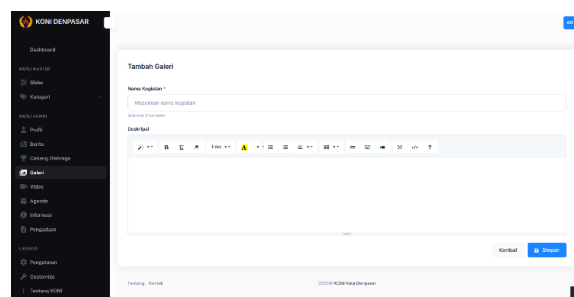


Gambar 7. Form Tambah Berita

Berdasarkan Gambar 8, halaman CRUD galeri digunakan untuk mengelola dokumentasi kegiatan pada sistem backend Portal Informasi Komite Olahraga Nasional Indonesia Denpasar melalui proses Create, Read, Update, dan Delete (CRUD). Halaman ini mendukung pengelolaan galeri foto dan galeri video yang akan ditampilkan pada website frontend sebagai media publikasi dokumentasi organisasi. Informasi yang ditampilkan meliputi judul galeri, jenis media, thumbnail, tanggal unggah, dan status publikasi, serta dilengkapi fitur pencarian untuk mempermudah pengelolaan data. Selanjutnya, pada Gambar 9 ditunjukkan form tambah galeri yang digunakan untuk menambahkan dokumentasi baru berupa foto maupun video ke dalam sistem dengan melengkapi data seperti judul, kategori, deskripsi, serta unggahan file atau tautan video. Implementasi fitur galeri ini bertujuan untuk mempermudah pengelolaan dokumentasi multimedia agar lebih terstruktur, terintegrasi, dan mudah diakses oleh masyarakat melalui website organisasi.



Gambar 8. Halaman CRUD Galeri



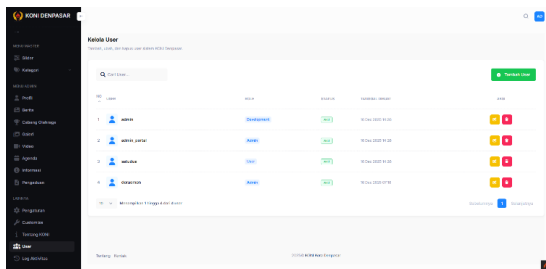
Gambar 9. Form Tambah Galeri Foto

Selain fitur utama seperti dashboard, berita, agenda, dan galeri, sistem backend Portal Informasi Komite Olahraga Nasional Indonesia Denpasar juga dilengkapi dengan beberapa fitur pendukung untuk menunjang pengelolaan informasi dan administrasi sistem. Fitur tersebut meliputi pengelolaan profil organisasi, informasi publik, pengaduan masyarakat, customize tampilan website, serta halaman tentang aplikasi. Kehadiran fitur-fitur ini bertujuan untuk mempermudah pengelolaan identitas organisasi, penyampaian informasi kepada masyarakat, serta penyesuaian tampilan website sesuai kebutuhan organisasi agar sistem dapat berjalan lebih fleksibel dan terintegrasi.

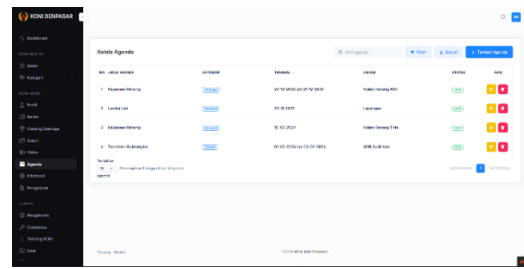
Berdasarkan Gambar 10, halaman manajemen user digunakan untuk mengelola data pengguna yang memiliki akses ke sistem backend melalui proses Create, Read, Update, dan Delete (CRUD). Administrator dapat mengatur data pengguna, hak akses berdasarkan peran, status akun, hingga penggantian password untuk mendukung keamanan sistem

berbasis role management. Selanjutnya, pada Gambar 11 ditunjukkan halaman agenda yang digunakan untuk mengelola informasi kegiatan organisasi, seperti judul kegiatan, tanggal, lokasi, dan deskripsi agenda yang akan ditampilkan pada website frontend sehingga informasi kegiatan dapat diperbarui secara cepat dan terstruktur.

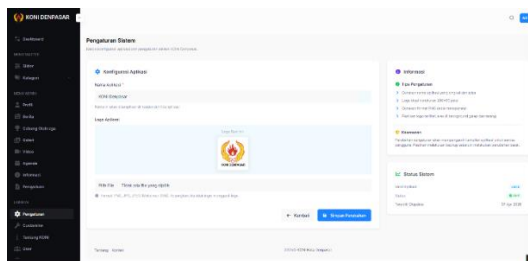
Pada Gambar 12, halaman pengaturan digunakan sebagai pusat konfigurasi sistem yang memungkinkan administrator melakukan penyesuaian identitas website, tampilan sistem, dan konfigurasi fitur lainnya tanpa mengubah kode program secara langsung. Selain itu, Gambar 13 menunjukkan halaman logs aktivitas yang berfungsi untuk memantau dan merekam seluruh aktivitas pengguna pada sistem, seperti login, penambahan data, perubahan data, dan penghapusan data. Implementasi fitur logs aktivitas membantu administrator dalam melakukan monitoring penggunaan sistem serta meningkatkan keamanan dan pengelolaan backend secara lebih terorganisasi.



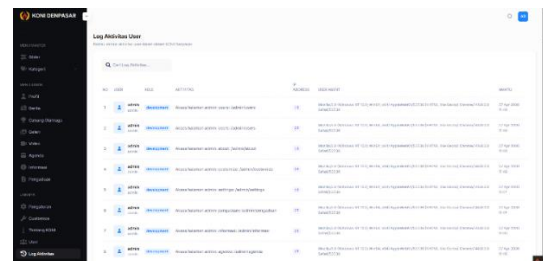
Gambar 10. Halaman Manajemen User



Gambar 11. Halaman Agenda(Konten)



Gambar 12. Halaman Pengaturan



Gambar 13. Halaman Logs Aktivitas

3.2.3 Implementasi Fungsional Sistem

Implementasi fungsional sistem mencakup proses utama yang berjalan dalam sistem, antara lain:

- Autentikasi pengguna (login sistem)
Sistem melakukan validasi username dan password sebelum memberikan akses ke halaman administrator.
- Pengelolaan data (CRUD)
Sistem mendukung operasi Create, Read, Update, dan Delete terhadap data berita, agenda, galeri, dan cabang olahraga.
- Manajemen konten website
Seluruh konten yang ditampilkan pada frontend dapat dikelola melalui *backend* oleh administrator.
- Integrasi database
Sistem menggunakan database MySQL untuk menyimpan seluruh data yang dikelola dalam sistem.

3.2.4 Analisis Implementasi

Berdasarkan hasil implementasi yang telah dilakukan, sistem portal informasi yang dikembangkan menunjukkan berbagai capaian penting dalam mendukung kebutuhan pengelolaan informasi pada organisasi. Sistem ini mampu berfungsi sebagai media informasi terpusat yang memungkinkan seluruh data dan informasi organisasi dikelola dalam satu platform yang saling terintegrasi. Dengan adanya integrasi tersebut, proses pengelolaan dan penyampaian informasi menjadi lebih terarah, terstruktur, dan mudah dikendalikan. Seluruh informasi yang berkaitan dengan kegiatan organisasi dapat disimpan dan diakses melalui satu sistem yang sama, sehingga mampu meningkatkan efektivitas penyampaian informasi kepada masyarakat maupun pihak internal organisasi. Selain itu, penggunaan sistem berbasis web juga memberikan kemudahan dalam proses distribusi informasi karena dapat diakses secara online melalui jaringan internet.

Sistem yang dikembangkan memudahkan administrator dalam mengelola data organisasi. Fitur yang ada memungkinkan pengelolaan informasi seperti berita, agenda, galeri dokumentasi, slider, dan data pendukung lainnya secara efisien melalui halaman backend yang sistematis. Antarmuka backend yang terstruktur membuat pengelolaan data lebih mudah dipahami dan dioperasikan. Kondisi ini meningkatkan efisiensi kerja karena proses input, pengeditan, dan penghapusan data lebih cepat dan terorganisir, serta mengurangi risiko kesalahan pengelolaan data dalam proses manual.

Dari sisi pengguna, sistem portal informasi ini memudahkan mereka dalam mendapat informasi yang dibutuhkan. Pengguna dapat mengakses informasi organisasi langsung melalui website tanpa prosedur autentikasi yang rumit. Informasi di website diatur secara terstruktur berdasarkan kategori, sehingga memudahkan pencarian dan pemahaman. Penyajian informasi yang rapi dan sistematis meningkatkan pengalaman penggunaan serta kenyamanan saat mengakses website, menjadikannya sebagai media publikasi dan sarana komunikasi digital untuk keterbukaan informasi organisasi kepada masyarakat.



Penggunaan framework CodeIgniter 4 dengan arsitektur Model View Controller (MVC) meningkatkan kualitas pengembangan sistem. MVC memisahkan fungsi antara model untuk mengelola data, controller untuk logika aplikasi, dan view untuk antarmuka pengguna. Ini membuat kode lebih rapi, terorganisasi, dan mudah dipahami. Penerapan arsitektur MVC juga mendukung pemeliharaan dan pengembangan sistem yang lebih efektif dan efisien di masa depan. Dengan sistem yang terorganisasi baik, pengembangan fitur dan perbaikan dapat dilakukan tanpa mengganggu fungsi aplikasi yang sudah ada.

3.3 Hasil Pengujian Sistem

Untuk mengetahui tingkat keberhasilan fungsi sistem yang telah dikembangkan, dilakukan pengujian menggunakan metode Blackbox Testing terhadap seluruh fitur utama sistem backend. Hasil pengujian sistem dapat dilihat pada Tabel 1 berikut.

Tabel 1. Blackox Testing Sistem Backend

Skenario Pengujian	Aksi Pengujian	Output yang Diharapkan	Hasil
Login dengan data valid	Input username & password benar serta reCAPTCHA valid	Sistem berhasil masuk ke halaman dashboard	Sesuai
Login dengan data tidak valid	Input username atau password salah	Sistem menampilkan pesan error dan menolak akses	Sesuai
Login tanpa reCAPTCHA	Input data benar tanpa mencentang reCAPTCHA	Sistem menolak proses login	Sesuai
Akses dashboard	Klik menu dashboard setelah login	Sistem menampilkan halaman dashboard	Sesuai
Tambah data berita	Input data berita lengkap dan valid	Data berhasil disimpan dan ditampilkan	Sesuai
Tambah data berita tidak valid	Input data berita tidak lengkap/kosong	Sistem menolak penyimpanan dan menampilkan pesan error	Sesuai
Edit data berita	Mengubah data berita yang sudah ada	Data berhasil diperbarui	Sesuai
Hapus data berita	Klik tombol hapus pada data berita	Data berhasil dihapus dari sistem	Sesuai
Upload file galeri valid	Upload file gambar/video dengan format sesuai	File berhasil diunggah dan ditampilkan	Sesuai
Upload file tidak valid	Upload file dengan format tidak didukung	Sistem menolak upload dan menampilkan pesan error	Sesuai
Tambah data user	Input data user baru dengan benar	User berhasil ditambahkan ke sistem	Sesuai
Edit data user	Mengubah data user	Data user berhasil diperbarui	Sesuai
Akses tanpa login	Mengakses halaman backend tanpa login	Sistem mengarahkan ke halaman login	Sesuai
Ubah pengaturan sistem	Mengubah konfigurasi sistem	Sistem menyimpan dan menerapkan perubahan	Sesuai
Melihat logs aktivitas	Mengakses menu logs	Sistem menampilkan riwayat aktivitas pengguna	Sesuai
Logout sistem	Klik tombol logout	Sistem keluar dan kembali ke halaman login	Sesuai

Berdasarkan Tabel 1, seluruh fitur utama pada sistem backend berhasil berjalan sesuai dengan skenario pengujian yang telah ditentukan. Seluruh proses pengelolaan data, autentikasi pengguna, validasi sistem, serta pengaturan konfigurasi dapat berjalan dengan baik tanpa ditemukan kesalahan fungsional yang signifikan. Hasil pengujian tersebut menunjukkan bahwa sistem backend yang dikembangkan memiliki tingkat stabilitas dan keandalan yang baik untuk digunakan dalam mendukung pengelolaan portal informasi organisasi. Dalam proses autentikasi, sistem terbukti mampu melakukan verifikasi data pengguna secara tepat, di mana kombinasi data login yang valid serta verifikasi reCAPTCHA yang benar akan mengarahkan pengguna ke halaman dashboard. Sebaliknya, apabila data yang dimasukkan tidak valid atau proses verifikasi reCAPTCHA tidak dilakukan, maka sistem secara otomatis menolak akses dan menampilkan pesan kesalahan yang relevan.

Pada tahap pengujian fitur pengelolaan data, sistem menunjukkan performa yang konsisten dalam menjalankan operasi dasar Create, Read, Update, dan Delete (CRUD). Hal ini terlihat dari pengujian pada modul berita yang digunakan sebagai sampel, di mana sistem mampu melakukan penyimpanan, pembaruan, serta penghapusan data secara akurat dan berkelanjutan. Selain itu, sistem juga memiliki mekanisme penanganan kesalahan yang cukup baik, seperti ketika pengguna memasukkan data yang tidak lengkap, sistem akan menolak proses penyimpanan serta memberikan notifikasi kesalahan yang informatif. Pengujian pada fitur pengelolaan media juga memperlihatkan bahwa sistem mampu memvalidasi file yang diunggah, dengan menerima format yang sesuai serta menolak file yang tidak memenuhi kriteria, sehingga aspek validasi input dapat dikatakan berjalan secara optimal.

Dari sisi manajemen sistem, fitur pengelolaan pengguna serta konfigurasi sistem juga telah berjalan sesuai dengan kebutuhan yang ditetapkan. Setiap perubahan data yang dilakukan dapat langsung disimpan dan diterapkan tanpa kendala



yang berarti. Selain itu, sistem juga menunjukkan tingkat keamanan akses yang baik, yang dibuktikan melalui pengujian akses tanpa autentikasi, di mana pengguna secara otomatis diarahkan ke halaman login. Fitur pencatatan aktivitas (*activity logs*) juga berfungsi secara efektif dalam merekam serta menampilkan riwayat aktivitas pengguna, sehingga dapat mendukung proses pemantauan dan evaluasi sistem secara menyeluruh.

Secara umum, hasil pengujian yang telah dilakukan menunjukkan bahwa sistem backend yang dikembangkan telah mampu memenuhi seluruh kebutuhan fungsional yang dirancang sebelumnya. Sistem tidak hanya mampu memproses berbagai jenis input, baik yang valid maupun tidak valid, tetapi juga dapat menghasilkan respons yang sesuai dengan skenario pengujian yang telah ditentukan. Dengan demikian, dapat disimpulkan bahwa sistem memiliki tingkat stabilitas dan keandalan yang baik, sehingga layak untuk diimplementasikan dalam mendukung pengelolaan data serta konten pada Portal Informasi KONI Denpasar secara efektif dan efisien.

3.4 Pembahasan

Hasil penelitian menunjukkan bahwa backend portal informasi KONI Denpasar berhasil dikembangkan menggunakan framework CodeIgniter 4 dengan pendekatan Model View Controller (MVC). Sistem yang dibangun mampu mendukung pengelolaan berita, agenda kegiatan, galeri dokumentasi, data pengguna, serta pengaturan hak akses administrator secara terintegrasi dalam satu platform. Berdasarkan hasil pengujian menggunakan metode Black Box Testing, seluruh fitur utama pada sistem dapat berjalan dengan baik sesuai kebutuhan pengguna tanpa ditemukan kesalahan fungsional yang signifikan.

Temuan penelitian ini sejalan dengan penelitian yang dilakukan oleh Mita Tri Murtiningsih (2023) yang menyatakan bahwa framework CodeIgniter mampu meningkatkan efektivitas pengelolaan data dan mempercepat proses pengembangan aplikasi berbasis web [2]. Selain itu, penelitian yang dilakukan oleh Malik Maulana dan Verra Sofica (2024) juga menunjukkan bahwa penggunaan CodeIgniter dapat menghasilkan sistem yang lebih ringan, fleksibel, dan mudah dikembangkan sesuai kebutuhan pengguna [6]. Kesamaan penelitian ini dengan penelitian terdahulu terletak pada penggunaan framework CodeIgniter sebagai dasar pengembangan aplikasi berbasis website.

Meskipun demikian, penelitian ini memiliki beberapa perbedaan dibandingkan penelitian sebelumnya. Sebagian besar penelitian terdahulu lebih berfokus pada pengembangan sistem akademik, layanan administrasi, dan sistem penjualan, sedangkan penelitian ini lebih menitikberatkan pada pengembangan backend portal informasi organisasi olahraga. Selain itu, penelitian ini tidak hanya membangun sistem informasi berbasis website, tetapi juga mengembangkan backend terintegrasi yang mendukung pengelolaan berita, agenda kegiatan, galeri dokumentasi, serta pengaturan hak akses administrator dalam satu platform terpusat.

Penerapan arsitektur MVC dalam penelitian ini memberikan dampak positif terhadap struktur pengembangan sistem. Pemisahan komponen Model, View, dan Controller membuat kode program menjadi lebih terorganisasi, modular, dan mempermudah proses pemeliharaan sistem di masa mendatang. Dengan demikian, backend portal informasi KONI Denpasar yang dikembangkan dinilai mampu menjadi solusi dalam mendukung pengelolaan informasi organisasi secara lebih efektif, terstruktur, dan mudah dikembangkan sesuai kebutuhan organisasi.

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa pengembangan backend Portal Informasi Komite Olahraga Nasional Indonesia (KONI) Denpasar menggunakan framework CodeIgniter 4 dengan pendekatan Model View Controller (MVC) berhasil menghasilkan sistem informasi berbasis web yang terintegrasi dan mampu mendukung proses pengelolaan administrasi organisasi secara lebih efektif. Sistem yang dibangun dapat mempermudah administrator dalam mengelola berbagai informasi seperti berita, agenda kegiatan, galeri dokumentasi, data pengguna, serta pengaturan hak akses melalui satu platform yang terpusat. Kontribusi utama dari penelitian ini terletak pada penerapan arsitektur MVC dalam pengembangan backend portal informasi organisasi olahraga sehingga menghasilkan sistem yang lebih terstruktur, modular, mudah dipelihara, dan mudah dikembangkan untuk kebutuhan di masa mendatang. Berbeda dengan penelitian sebelumnya yang umumnya berfokus pada sistem akademik dan administrasi umum, penelitian ini lebih menitikberatkan pada pengembangan backend terintegrasi yang mendukung pengelolaan informasi organisasi olahraga secara menyeluruh. Hasil pengujian menggunakan metode Blackbox Testing menunjukkan bahwa seluruh fitur pada sistem dapat berjalan sesuai kebutuhan pengguna tanpa ditemukan kendala fungsional yang signifikan. Dengan demikian, backend portal informasi yang dikembangkan dinilai layak untuk diterapkan sebagai media pengelolaan informasi KONI Denpasar agar proses distribusi dan pengelolaan informasi dapat berjalan lebih efektif, terorganisasi, dan efisien.

REFERENCES

- [1] M. M. Bani, S. Dwi, I. Mau, and S. G. Balimema, "Implementasi Sistem Pelayanan Kesehatan Berbasis Online Menggunakan Codeigniter 3 dan Metode MVC CodeIgniter 3 , diharapkan dapat tercipta solusi yang efektif dalam menunjang layanan Delone And Mclean Website Sistem Informasi Akademik STIKES Sukabumi , de," *J. Inform. dan Sains Teknol.*, vol. 3, no. 4, pp. 28–38, 2025, doi: <https://doi.org/10.62951/modem.v3i4.639>.
- [2] M. T. Murtiningsih, "Penerapan Framework CodeIgniter dalam Sistem Informasi Pengelolaan Jadwal Video Conference (VICON) Polrestabes Palembang Implementation of the Framework CodeIgniter in the Schedules Management Information System of Video Conference (VICON) Polrestabes," *SMATIKA STIKI Inform. J.*, vol. 13, no. 1, pp. 83–92, 2023, doi:



- <https://doi.org/10.32664/smatika.v13i01.724>.
- [3] A. R. Y. Putra and D. Gunawan, "Rancang Bangun Sistem Rekrutmen Karyawan Berbasis Website Menggunakan Codeigniter 4 Studi Kasus Cv . Amins Project Teknologi Indonesia," *Asos. Progr. Stud. Infokom*, vol. 1, no. 1, pp. 212–215, 2025, doi: <https://infocoding.apsi-ptma.org/index.php/proceeding/article/view/27>.
 - [4] P. A. Rohmah, P. Setiaji, and S. Muzid, "Pengembangan Multi - Platform System Development for Violence Complaint Services using the CodeIgniter Framework," *Sist. J. Sist. Inf.*, vol. 14, no. 6, pp. 2668–2681, 2025, doi: <https://doi.org/10.32520/stmsi.v14i6.5528>.
 - [5] N. M. Sari and D. Sunardi, "Website Paud AI-Muchlis Kelurahan Tanjung Jaya Menggunakan Framework Codeigniter," *J. Komputer, Inf. dan Teknol.*, vol. 5, no. 2, pp. 1–14, 2025, doi: <https://doi.org/10.53697/jkomitek.v5i2.3826>.
 - [6] M. Maulana and V. Sofica, "Perancangan Sistem Informasi Berbasis Website Codeigniter Pada Toko Agama Aceh," *Informatics Educ. Prof. J. Informatics*, vol. 7, no. 2, pp. 165–173, 2024, doi: <https://doi.org/10.51211/itbi.v8i1.2184>.
 - [7] N. Selajar, D. Costa, P. K. Karo, and N. M. Faizah, "Perancangan Sistem Informasi Penerimaan Siswa Baru di SMK Harapan Bangsa Berbasis Web Menggunakan CodeIgniter 4," *Comput. J.*, vol. 2, no. 2, pp. 83–90, 2024, doi: <https://doi.org/10.58477/cj.v2i2.172>.
 - [8] M. Farid, A. Prasetyo, M. R. Ardiansyah, A. A. Ashari, T. Putro, and E. Rahmawati, "Rancang Bangun Tracking Pengiriman Berbasis Website Menggunakan Metode Systems Development Life Cycle (SDLC) dengan Model Waterfall," *J. Teknol. Inf. dan Komun.*, vol. 8, no. 2, pp. 307–315, 2024, doi: <https://doi.org/10.35870/jtik.v8i2.1387>.
 - [9] V. A. Kurniyanti and D. Murdiani, "Perbandingan Model Waterfall Dengan Prototype Pada Pengembangan System Informasi Berbasis Website," *J. Syntax FUSION*, vol. 2, no. 08, 2022.
 - [10] R. W. Saputra, C. Q. Pirera, and V. V. Verdana, "Analisis Resiko Penggunaan Metode Waterfall dan Prototyping dalam Pengembangan Website," *J. Mhs. Tek. Inform.*, vol. 8, no. 4, pp. 4405–4410, 2024, doi: <https://doi.org/10.36040/jati.v8i4.9943>.
 - [11] A. Ibrahim and M. Sutrisno, "Implementasi Framework CodeIgniter dalam Sistem Informasi Akademik Berbasis Web," 2024, [Online]. Available: <https://example.com>
 - [12] A. Yulia and R. A. Aziz, "Sistem Informasi Penjualan Sayur Berbasis Website Menggunakan Framework Codeigniter," *J. Penelit. Sist. Inf.*, vol. 1, no. 3, pp. 211–220, 2023, doi: <https://doi.org/10.54066/jpsi.v1i3.687>.
 - [13] H. Idwan, "Perancangan Sistem Informasi Website Kantor Desa Beurawe Berbasis Framework CodeIgniter," *J. Manaj. Sist. Inf.*, vol. 4, no. 2, pp. 93–101, 2025, doi: [10.59431/jmasif.v4i2.605](https://doi.org/10.59431/jmasif.v4i2.605).
 - [14] M. S. Pebriadi, P. Salman, and T. K. Fattah, "Implementasi Use Case Diagram dan Activity Diagram dalam Perancangan Aplikasi Kalkulator Pajak bagi UMKM," *Multitek Indones. J. Ilm.*, vol. 6223, no. 1, pp. 44–58, 2023, doi: <https://doi.org/10.24269/mtkind.v17i1.6024>.
 - [15] H. P. Putra, Sugiarto, H. Maulana, E. Triandini, and P. F. Nuryanda, "Relasional Desain Activity Diagram Sistem Informasi Agen Travel," *Pros. Semin. Nas. Teknol. dan Sist. Inf.*, vol. 2, no. 1, pp. 10–11, 2022, doi: <https://doi.org/10.33005/sitasi.v2i1.303>.
 - [16] Z. M. Anggraini, "Pengujian Crud Link Pada Aplikasi," *J. Inf. Syst. Bus. Technol.*, vol. 1, no. 1, pp. 83–90, 2025, doi: <https://journal.jci.co.id/jisbt/article/view/172>.
 - [17] M. Muthohir, *Sistem Informasi CRUD, REPORTING, dan GRAFIS dengan CODEIGNITER 4*. Semarang: Yayasan Prima Agus Teknik, 2023.
 - [18] R. Noviana, "Pembuatan Aplikasi Penjualan Berbasis Web Monja Store Menggunakan PHP dan MySQL," *J. Tek. dan Sci.*, vol. 1, no. 2, pp. 112–124, 2022, doi: <https://doi.org/10.56127/jts.v1i2.128>.
 - [19] Y. Saputra and D. Mardiyati, "Implementasi Sistem Informasi Manajemen Klinik menggunakan Metode Black Box Testing," *J. Inform. dan Tek. Elektro Terap.*, vol. 13, no. 1, pp. 1611–1618, 2025, doi: <https://doi.org/10.23960/jitet.v13i1.6015>.
 - [20] A. W. Syahrani, N. P. Dewi, N. Ramadhani, and B. Said, "Uji Keamanan Back end Aplikasi Berbasis Website Menggunakan Metode Black Box Testing," *J. Ilm. Sist. Informasi, Teknol. Inf. dan Sist. Komput.*, vol. 19, no. 2, pp. 215–226, 2024, doi: [10.33998/processor.2024.19.19.1752](https://doi.org/10.33998/processor.2024.19.19.1752).