# LAPORAN

**PENGEMBANGAN SISTEM MANAJEMEN DATA DOSEN BERBASIS WEB MENGGUNAKAN CODEIGNITER 4**

DISUSUN OLEH:

SALMAN AL FARIZI

MUCHLIS



**PROGRAM STUDI ILMU KOMPUTER  
FAKULTAS TEKNIK DAN ILMU KOMPUTER  
UNIVERSITAS MUHAMMADIYAH BIMA   
TAHUN 2025**

# DAFTAR ISI

[LAPORAN i](#_Toc201117172)

[DAFTAR ISI ii](#_Toc201117173)

[BAB I PENDAHULUAN 4](#_Toc201117174)

[1.1. Latar Belakang 4](#_Toc201117175)

[1.2. Rumusan Masalah 4](#_Toc201117176)

[1.3. Tujuan 4](#_Toc201117177)

[1.4. Manfaat 5](#_Toc201117178)

[BAB II TINJAUAN PUSTAKA 6](#_Toc201117179)

[2.1 Sistem Informasi 6](#_Toc201117180)

[2.2 Sistem Manajemen Data 6](#_Toc201117181)

[2.3 Metode Waterfall 7](#_Toc201117182)

[2.4 CodeIgniter 4 (CI4) 8](#_Toc201117183)

[2.5 MySQL 9](#_Toc201117184)

[2.6 Konsep CRUD 10](#_Toc201117185)

[2.7 Role-Based Access Control (RBAC) 10](#_Toc201117186)

[2.8 Antarmuka Pengguna (User Interface) 10](#_Toc201117187)

[2.9 Metode Pengujian Sistem (Black Box Testing) 11](#_Toc201117188)

[BAB III METODOLOGI PENGEMBANGAN SISTEM 12](#_Toc201117189)

[3.1 Model Pengembangan Sistem 12](#_Toc201117190)

[3.1.1 Analisis Kebutuhan 12](#_Toc201117191)

[3.1.2 Perancangan Sistem 12](#_Toc201117192)

[3.1.3 Implementasi 12](#_Toc201117193)

[3.1.4 Pengujian 12](#_Toc201117194)

[3.1.5 Pemeliharaan 12](#_Toc201117195)

[3.2 Alat dan Teknologi yang Digunakan 13](#_Toc201117196)

[3.2.1 Bahasa Pemrograman 13](#_Toc201117197)

[3.2.2 Framework 13](#_Toc201117198)

[3.2.3 Database 13](#_Toc201117199)

[3.2.4 Server Lokal 13](#_Toc201117200)

[3.2.5 Text Editor 13](#_Toc201117201)

[3.2.6 Peramban Web 13](#_Toc201117202)

[3.3 Spesifikasi Sistem 14](#_Toc201117203)

[3.3.1 Spesifikasi Perangkat Keras 14](#_Toc201117204)

[3.3.2 Spesifikasi Perangkat Lunak 14](#_Toc201117205)

[3.3.3 Peran Pengguna Sistem 14](#_Toc201117206)

[BAB IV PERANCANGAN DAN IMPLEMENTASI SISTEM 15](#_Toc201117207)

[4.1 Diagram Perancangan Sistem 15](#_Toc201117208)

[4.1.1 Use Case Diagram 15](#_Toc201117209)

[4.1.2 Entity Relationship Diagram (ERD) 16](#_Toc201117210)

[4.1.3 Diagram Alur Sistem 18](#_Toc201117211)

[4.2 Struktur Folder Aplikasi CodeIgniter 4 20](#_Toc201117212)

[4.3 Desain Antarmuka Pengguna 21](#_Toc201117213)

[4.3.1 Halaman Dashboard (Pengguna Umum) 21](#_Toc201117214)

[4.3.2 Halaman Registrasi 22](#_Toc201117215)

[4.3.3 Halaman Login 22](#_Toc201117216)

[4.3.4 Halaman Dashboard Dosen 23](#_Toc201117217)

[4.3.5 Pengelolaan Portofolio Dosen 24](#_Toc201117218)

[4.3.6 Pengelolaan Profil Dosen 26](#_Toc201117219)

[4.4 Implementasi Fitur Sistem 28](#_Toc201117220)

[4.4.1 Pengelolaan Data Dosen 28](#_Toc201117221)

[4.4.2 Fitur bidang keahlian dan jadwal mengajar 29](#_Toc201117222)

[4.4.3 Tampilan Grafik Evaluasi Kinerja Dosen 30](#_Toc201117223)

[4.5 Pengujian Sistem 31](#_Toc201117224)

[4.5.1 Tujuan Pengujian 31](#_Toc201117225)

[4.5.2 Metode Pengujian 31](#_Toc201117226)

[4.5.3 Hasil Pengujian 32](#_Toc201117227)

[4.5.4 Evaluasi Hasil Pengujian 32](#_Toc201117228)

[4.6 Evaluasi istem 33](#_Toc201117229)

[BAB V PENUTUP 34](#_Toc201117230)

[5.1 Kesimpulan 34](#_Toc201117231)

[5.2 Saran Pengembangan 34](#_Toc201117232)

[DAFTAR PUSTAKA 36](#_Toc201117233)

# BAB I PENDAHULUAN

## Latar Belakang

Sistem manajemen data dosen adalah suatu sistem yang dirancang untuk mencatat, menyimpan, dan mengelola informasi dosen secara terstruktur dan terpusat. Informasi ini mencakup data pribadi, riwayat pendidikan, jabatan akademik, publikasi ilmiah, dan berbagai hal lain yang berkaitan dengan aktivitas dosen di lingkungan perguruan tinggi. Dengan adanya sistem ini, institusi pendidikan dapat dengan mudah melakukan pelacakan dan pembaruan data dosen secara real-time.

Namun, dalam praktiknya, masih banyak perguruan tinggi yang mengelola data dosen secara manual, baik menggunakan dokumen fisik maupun spreadsheet sederhana yang tidak terintegrasi. Kondisi ini menimbulkan berbagai permasalahan seperti duplikasi data, kesalahan input, kehilangan informasi penting, hingga keterlambatan dalam proses pelaporan atau pengambilan keputusan. Tidak hanya menyulitkan staf administrasi, hal ini juga dapat berdampak pada penilaian akreditasi institusi.

Untuk mengatasi masalah tersebut, dibutuhkan sistem manajemen data dosen berbasis web yang modern dan responsif. Sistem ini dirancang menggunakan framework CodeIgniter 4 dengan dukungan database MySQL serta antarmuka yang ramah pengguna. Tujuannya adalah untuk menciptakan platform terpusat yang aman, efisien, dan dapat diakses oleh berbagai peran pengguna seperti admin dan dosen. Dengan demikian, proses pengelolaan data dosen dapat berjalan lebih efektif, akurat, dan mendukung transformasi digital di lingkungan perguruan tinggi.

## Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, maka rumusan masalah dalam pengembangan sistem ini adalah sebagai berikut:

1. Bagaimana membangun sistem informasi yang dapat mengelola data dosen secara terpusat dan efisien?
2. Bagaimana merancang sistem perizinan perubahan data antara dosen dan admin agar lebih transparan dan aman?
3. Bagaimana menyajikan data dosen dalam bentuk laporan yang mudah diakses dan dapat diekspor?
4. Bagaimana mengimplementasikan sistem berbasis web yang user-friendly dan responsif menggunakan CodeIgniter 4?

## Tujuan

Tujuan dari penyusunan makalah dan pengembangan awal sistem manajemen data dosen ini adalah sebagai berikut:

1. Merancang sistem informasi berbasis web yang dapat digunakan untuk mengelola data dosen secara terpusat dan terstruktur.
2. Menyusun konsep sistem perizinan perubahan data antara dosen dan admin secara aman dan transparan.
3. Membuat simulasi atau prototipe awal dari fitur laporan data dosen yang dapat diekspor ke dalam format PDF atau Excel.
4. Mengembangkan tampilan awal antarmuka sistem dengan pendekatan desain yang user-friendly dan responsif menggunakan framework CodeIgniter 4.

## Manfaat

Pengembangan sistem manajemen data dosen ini diharapkan dapat memberikan manfaat sebagai berikut:

1. **Bagi institusi:** Membantu dalam pengelolaan data dosen secara terpusat dan terstruktur, sehingga proses administrasi akademik menjadi lebih efisien dan akurat.
2. **Bagi admin/operator:** Memudahkan dalam pencatatan, pembaruan, dan pelacakan riwayat perubahan data dosen dengan sistem yang jelas dan terdokumentasi.
3. **Bagi dosen:** Memberikan kontrol terhadap data pribadi melalui fitur perizinan, sehingga data yang ditampilkan lebih relevan dan sesuai dengan kenyataan.
4. **Bagi pengembang sistem:** Sebagai media pembelajaran dalam merancang dan mengimplementasikan sistem informasi berbasis web menggunakan CodeIgniter 4.
5. **Bagi peneliti selanjutnya:** Menjadi referensi awal untuk pengembangan sistem serupa atau sistem informasi lainnya di lingkungan akademik.

# BAB II TINJAUAN PUSTAKA

## Sistem Informasi

Sistem informasi merupakan kombinasi dari teknologi informasi dan aktivitas manusia yang menggunakan teknologi tersebut untuk mendukung operasi dan manajemen. Menurut Laudon dan Laudon (2018), sistem informasi adalah suatu sistem yang mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi untuk mendukung pengambilan keputusan, koordinasi, analisis, dan visualisasi dalam suatu organisasi.

Dalam konteks akademik, sistem informasi berperan penting dalam pengelolaan berbagai data dan proses, termasuk data mahasiswa, dosen, kurikulum, dan sebagainya. Penggunaan sistem informasi dapat meningkatkan efisiensi, transparansi, serta integritas data yang dibutuhkan oleh institusi pendidikan tinggi dalam menunjang kegiatan administratif maupun akademik.

## Sistem Manajemen Data

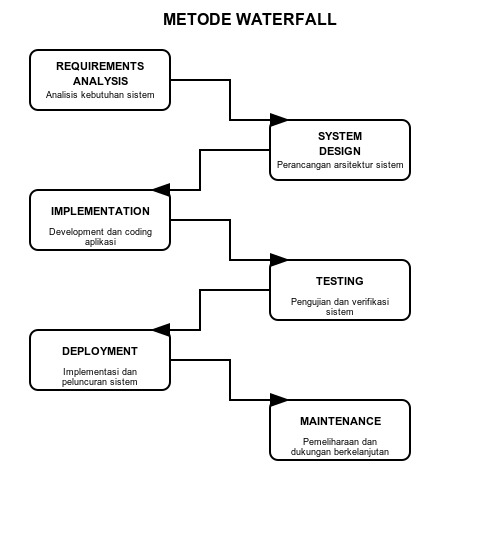
Sistem manajemen data merupakan serangkaian proses dan teknologi yang digunakan untuk mengumpulkan, menyimpan, mengatur, dan mengakses data secara efisien dan terstruktur. Menurut *Stair & Reynolds* (2016), manajemen data adalah metode sistematis untuk mengelola aset data organisasi sebagai sumber daya strategis. Dalam konteks institusi pendidikan, data yang dikelola meliputi data mahasiswa, dosen, mata kuliah, penelitian, dan aktivitas akademik lainnya.

Manajemen data yang baik memungkinkan institusi untuk menjaga konsistensi, integritas, dan keamanan informasi. Tanpa sistem yang terorganisir, pengelolaan data dapat menjadi tidak efisien, menyebabkan duplikasi, kehilangan data, atau kesalahan input yang dapat berdampak langsung pada pengambilan keputusan manajerial. Oleh karena itu, dibutuhkan sistem yang tidak hanya mampu menyimpan data, tetapi juga mendukung pelacakan riwayat perubahan, kontrol akses, serta pemrosesan data secara cepat dan akurat.

Dalam pengembangan sistem manajemen data dosen ini, pendekatan manajemen data diterapkan melalui penggunaan database relasional (MySQL) yang terintegrasi dengan sistem antarmuka web. Data dosen yang dikelola mencakup identitas pribadi, riwayat pendidikan, jabatan akademik, hingga aktivitas profesional. Sistem ini juga dirancang untuk memudahkan admin dalam memperbarui data, serta memberi akses terbatas kepada dosen untuk melihat atau mengajukan perubahan data. Dengan demikian, sistem manajemen data ini tidak hanya bertindak sebagai penyimpan informasi, tetapi juga sebagai alat bantu administrasi akademik yang modern dan efisien.

## Metode Waterfall

Metode Waterfall merupakan model pengembangan perangkat lunak tradisional yang bersifat sistematis dan terstruktur. Dalam model ini, setiap tahapan dilakukan secara berurutan dan harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahapan berikutnya. Model ini sangat cocok digunakan untuk proyek dengan kebutuhan yang sudah terdefinisi dengan jelas sejak awal.



**Gambar 2.2 Alur pengembangan metode waterfall**

Sebagaimana terlihat pada Gambar 2.1, alur pengembangan dengan metode Waterfall dimulai dari tahap Analisis Kebutuhan, dilanjutkan dengan Perancangan Sistem, kemudian masuk ke tahap Implementasi, Pengujian, dan diakhiri dengan Pemeliharaan.

1. **Analisis Kebutuhan:** Pada tahap ini dilakukan proses identifikasi dan dokumentasi terhadap kebutuhan pengguna. Informasi yang dikumpulkan berasal dari hasil observasi, wawancara, atau studi literatur terkait proses pengelolaan data dosen secara manual yang selama ini digunakan. Hasil dari tahap ini berupa dokumen kebutuhan sistem yang menjadi acuan pengembangan.
2. **Perancangan Sistem:** Setelah kebutuhan dikumpulkan, tahap selanjutnya adalah perancangan sistem yang mencakup struktur database, arsitektur sistem, dan antarmuka pengguna (UI). Tujuannya adalah merancang solusi teknis yang mampu memenuhi kebutuhan pengguna secara efisien.
3. **Implementasi:** Tahap ini merupakan proses pengkodean atau pembangunan sistem berdasarkan desain yang telah dibuat. Dalam konteks proyek ini, implementasi dilakukan menggunakan framework CodeIgniter 4, dengan bahasa pemrograman PHP serta database MySQL. Fokus utama pada tahap ini adalah membuat modul-modul inti seperti login, dashboard, dan pengelolaan data dosen.
4. **Pengujian:** Setelah sistem diimplementasikan, dilakukan pengujian untuk memastikan bahwa sistem berjalan sesuai dengan spesifikasi. Pengujian dilakukan secara manual menggunakan metode black-box untuk menguji setiap fitur tanpa melihat struktur kode program.
5. **Pemeliharaan:** Tahap terakhir adalah pemeliharaan sistem. Tahap ini mencakup perbaikan bug, peningkatan performa, serta pengembangan fitur tambahan berdasarkan masukan pengguna di masa depan.

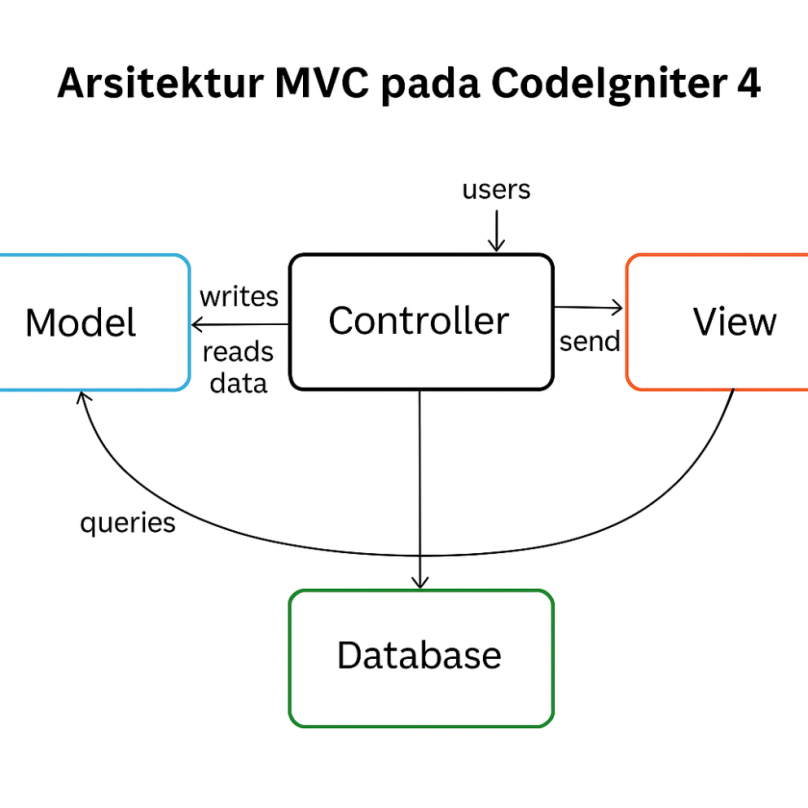
Dengan mengikuti alur pada Gambar 2.1, proses pengembangan sistem menjadi lebih terstruktur, terdokumentasi, dan mudah dievaluasi. Meskipun model Waterfall kurang fleksibel terhadap perubahan kebutuhan, namun dalam proyek ini model ini cukup efektif karena kebutuhan sistem sudah dirancang sejak awal dan tidak mengalami banyak perubahan.

## CodeIgniter 4 (CI4)

CodeIgniter 4 adalah salah satu framework PHP yang digunakan untuk membangun aplikasi web dengan arsitektur Model-View-Controller (MVC). Framework ini dikembangkan untuk membantu developer dalam menyusun kode program secara terstruktur, memisahkan antara logika bisnis, tampilan, dan pengelolaan data.

CI4 hadir dengan berbagai fitur modern seperti dukungan namespace, routing berbasis controller, RESTful API, validasi data, keamanan CSRF/XSS, dan ORM (Object Relational Mapping) ringan. Keunggulan lainnya adalah performa yang cepat dan dokumentasi yang lengkap, sehingga sangat cocok digunakan oleh pemula maupun pengembang tingkat lanjut.

Dalam pengembangan sistem manajemen data dosen ini, CodeIgniter 4 dipilih karena sifatnya yang ringan, mudah dipelajari, dan fleksibel untuk dikembangkan sesuai kebutuhan. Dengan menerapkan pola MVC, CI4 memudahkan dalam pemisahan kode backend (controller dan model) dengan frontend (view), sehingga proses pengembangan menjadi lebih terorganisir dan mudah dipelihara.



**Gambar 2.2 Arsitektur MVC pada CodeIgniter 4**

Seperti terlihat pada Gambar 2.2, *Controller* bertugas menerima permintaan dari pengguna (*user*) dan menjadi penghubung antara *View* dan *Model*. *Model* bertanggung jawab dalam mengelola logika bisnis serta interaksi dengan database, sedangkan *View* menyajikan data ke pengguna dalam bentuk antarmuka yang dapat dilihat dan digunakan. Data yang dibutuhkan oleh sistem diambil oleh *Model* dari *Database*, kemudian dikirim ke *Controller* dan ditampilkan melalui *View*. Dengan pola ini, pengembangan sistem menjadi lebih modular dan mudah dalam proses pemeliharaan serta pengembangan di masa mendatang.

## MySQL

MySQL adalah sistem manajemen basis data relasional (RDBMS) yang bersifat open-source dan banyak digunakan dalam pengembangan aplikasi berbasis web. MySQL menggunakan bahasa Structured Query Language (SQL) sebagai bahasa utama untuk mengelola dan memanipulasi data dalam database. MySQL dikenal karena kecepatan, keandalan, dan kemampuannya dalam menangani jumlah data yang besar.

Dalam pengembangan sistem manajemen data dosen, MySQL digunakan untuk menyimpan seluruh informasi yang berkaitan dengan dosen, seperti data pribadi, riwayat pendidikan, jabatan akademik, dan lainnya. Kelebihan MySQL yang mendukung integrasi dengan framework seperti CodeIgniter 4 membuatnya menjadi pilihan tepat untuk sistem ini, karena memudahkan proses koneksi, query, dan pengelolaan data secara efisien dan aman.

## Konsep CRUD

CRUD adalah singkatan dari Create, Read, Update, dan Delete, yang merupakan empat operasi dasar yang digunakan dalam pengelolaan data pada sistem informasi. Operasi CRUD menjadi fondasi dalam pembangunan sistem berbasis data karena mencerminkan interaksi utama pengguna terhadap data.

1. **Create**: Menambahkan data baru ke dalam sistem, seperti menambahkan data dosen baru.
2. **Read:** Menampilkan atau membaca data dari database, seperti melihat daftar dosen.
3. **Update:** Mengubah atau memperbarui data yang sudah ada, misalnya mengedit informasi riwayat pendidikan dosen.
4. **Delete:** Menghapus data dari sistem, seperti menghapus data dosen yang sudah tidak aktif.

Dalam sistem manajemen data dosen ini, seluruh modul utama menggunakan prinsip CRUD untuk mempermudah pengguna (admin atau dosen) dalam mengelola data. Dengan mengimplementasikan operasi CRUD secara efisien, sistem dapat memberikan pengalaman pengguna yang intuitif dan mendukung konsistensi data.

## Role-Based Access Control (RBAC)

Role-Based Access Control (RBAC) adalah sebuah metode pengelolaan hak akses pengguna ke dalam sistem informasi berdasarkan peran (role) yang dimilikinya. Dalam pendekatan ini, setiap pengguna diberikan hak akses sesuai dengan tanggung jawab atau peran yang ditetapkan, seperti admin, dosen, atau user biasa.

Dengan menerapkan RBAC, sistem dapat membatasi akses ke fitur-fitur tertentu sehingga hanya pengguna dengan otorisasi tertentu yang dapat melakukan tindakan seperti mengubah, menghapus, atau menambah data. Sebagai contoh, hanya admin yang dapat mengelola seluruh data dosen, sementara dosen hanya dapat melihat atau mengajukan perubahan pada data pribadinya.

Penerapan RBAC pada sistem manajemen data dosen bertujuan untuk meningkatkan keamanan dan kontrol terhadap data, sekaligus memastikan bahwa pengguna hanya dapat melakukan tindakan yang sesuai dengan wewenangnya. Hal ini juga membantu menghindari kesalahan atau penyalahgunaan akses dalam sistem.

## Antarmuka Pengguna (User Interface)

Antarmuka pengguna atau *User Interface (UI)* adalah komponen penting dalam pengembangan sistem informasi yang berperan sebagai media interaksi antara pengguna dan sistem. UI mencakup semua elemen visual seperti tombol, menu, formulir, dan halaman yang memungkinkan pengguna untuk memberikan perintah serta menerima informasi dari sistem. Menurut *Shneiderman (2016)*, antarmuka yang dirancang dengan baik dapat meningkatkan efisiensi, kenyamanan, dan keberhasilan penggunaan sistem secara keseluruhan.

Beberapa prinsip dasar dalam desain UI antara lain adalah konsistensi tampilan, kemudahan navigasi, kejelasan informasi, dan responsivitas terhadap berbagai perangkat. Dalam konteks pengembangan aplikasi web modern, pendekatan desain yang umum digunakan adalah *responsive design* yang memungkinkan sistem menyesuaikan tampilan secara otomatis di berbagai ukuran layar. Selain itu, penggunaan framework CSS seperti Bootstrap sangat membantu dalam mempercepat proses pembuatan UI yang estetis dan fungsional.

Pada sistem manajemen data dosen, penerapan UI yang baik menjadi penting agar admin maupun dosen dapat mengakses data dengan mudah, memahami fungsi setiap fitur, serta melakukan input dan pembaruan data tanpa kebingungan. UI yang buruk dapat menyebabkan kesalahan input, memperlambat proses kerja, bahkan membuat pengguna enggan menggunakan sistem. Oleh karena itu, pemahaman tentang prinsip dasar UI menjadi dasar penting dalam merancang tampilan sistem yang efektif.

## Metode Pengujian Sistem (Black Box Testing)

Pengujian sistem merupakan tahapan penting dalam proses pengembangan perangkat lunak untuk memastikan bahwa sistem berjalan sesuai dengan kebutuhan pengguna. Salah satu metode yang umum digunakan dalam pengujian perangkat lunak adalah Black Box Testing.

Black Box Testing merupakan metode pengujian yang berfokus pada pengujian fungsionalitas sistem tanpa memeriksa struktur internal kode program. Pengujian dilakukan dengan memberikan input pada sistem dan kemudian mengevaluasi apakah output yang dihasilkan sesuai dengan yang diharapkan. Metode ini sangat efektif untuk menguji sistem dari sisi pengguna akhir, karena hanya memperhatikan hasil keluaran berdasarkan masukan yang diberikan, tanpa melihat cara sistem menghasilkan keluaran tersebut.

Black Box Testing sangat sesuai digunakan dalam pengujian sistem manajemen data dosen ini karena sistem dibangun berbasis antarmuka pengguna dan melibatkan interaksi langsung dengan pengguna (user). Dengan pendekatan ini, fitur-fitur seperti login, pengelolaan data dosen, penambahan bidang keahlian, hingga tampilan grafik evaluasi kinerja dapat diuji melalui input normal dan ekstrem untuk memastikan stabilitas serta keakuratan output.

Metode ini juga memungkinkan identifikasi kesalahan pada logika aplikasi, validasi form, dan alur sistem yang mungkin tidak terdeteksi dari sisi pengkodean. Oleh karena itu, Black Box Testing digunakan sebagai pendekatan utama dalam pengujian sistem ini.

# BAB III METODOLOGI PENGEMBANGAN SISTEM

## Model Pengembangan Sistem

Model pengembangan sistem yang digunakan dalam proyek ini adalah model Waterfall. Model ini merupakan metode klasik dalam rekayasa perangkat lunak yang bersifat linear dan sistematis, di mana setiap tahap dilakukan secara berurutan dan tidak dapat kembali ke tahap sebelumnya sebelum tahap tersebut selesai dilaksanakan. Model Waterfall terdiri dari beberapa tahapan utama, yaitu:

### **Analisis Kebutuhan**

Pada tahap ini dilakukan identifikasi kebutuhan sistem berdasarkan permasalahan yang ada di lapangan. Kebutuhan sistem digali melalui studi literatur dan analisis terhadap proses pengelolaan data dosen secara manual.

### **Perancangan Sistem**

Tahap ini meliputi perancangan struktur sistem, seperti perancangan antarmuka pengguna (UI), struktur database, serta diagram yang merepresentasikan alur sistem (use case, ERD, dll).

### **Implementasi**

Tahap implementasi dilakukan dengan mengembangkan sistem menggunakan framework CodeIgniter 4. Proses ini mencakup pembuatan modul, integrasi database, dan pengkodean tampilan serta logika aplikasi.

### **Pengujian**

Setelah sistem selesai dikembangkan, dilakukan pengujian fungsional untuk memastikan bahwa setiap fitur berjalan sesuai dengan kebutuhan. Pengujian dilakukan secara manual melalui simulasi input-output.

### **Pemeliharaan**

Tahap ini mencakup perbaikan kesalahan atau bug yang ditemukan setelah pengujian, serta pengembangan fitur tambahan apabila dibutuhkan di masa mendatang.

Model Waterfall ini diimplementasikan dalam pengembangan sistem manajemen data dosen dengan mengikuti alur dari tahap awal hingga tahap implementasi awal. Pada tahap analisis, dilakukan identifikasi kebutuhan berdasarkan proses pengelolaan data dosen secara manual yang masih digunakan. Tahap perancangan meliputi pembuatan struktur folder CI4, wireframe halaman dashboard, serta perancangan database. Implementasi awal difokuskan pada pembangunan halaman dashboard menggunakan framework CodeIgniter 4, sementara tahap pengujian dan pemeliharaan direncanakan untuk dilakukan setelah fitur utama selesai dikembangkan.Alat & Teknologi yang Digunakan

## Alat dan Teknologi yang Digunakan

Dalam pengembangan sistem manajemen data dosen ini, digunakan beberapa alat dan teknologi utama yang mendukung proses perancangan, implementasi, dan pengujian sistem. Berikut adalah penjelasannya:

### **Bahasa Pemrograman**

PHP digunakan sebagai bahasa utama dalam pengembangan sisi backend karena bersifat open-source dan banyak didukung komunitas. Sementara itu, HTML, CSS, dan JavaScript digunakan untuk membangun tampilan antarmuka pengguna (frontend) agar sistem terlihat interaktif dan responsif.

### **Framework**

CodeIgniter 4 dipilih sebagai framework pengembangan karena mendukung arsitektur Model-View-Controller (MVC), memiliki dokumentasi yang lengkap, ringan, serta cocok untuk pengembangan aplikasi berbasis web secara terstruktur.

### **Database**

MySQL digunakan sebagai sistem manajemen basis data (DBMS) untuk menyimpan seluruh data dosen yang dikelola oleh sistem, mulai dari data pribadi, riwayat pendidikan, hingga jabatan akademik.

### **Server Lokal**

XAMPP digunakan sebagai web server lokal yang menyediakan layanan Apache dan MySQL secara terintegrasi, sehingga memudahkan proses pengembangan dan pengujian sistem secara offline.

### **Text Editor**

Visual Studio Code (VS Code) digunakan untuk menulis dan mengelola kode program karena memiliki tampilan yang fleksibel serta mendukung banyak ekstensi yang mempermudah pengembangan aplikasi web.

### **Peramban Web**

Google Chrome digunakan untuk menampilkan dan menguji sistem karena mendukung teknologi web terbaru dan memberikan performa yang cepat dalam proses rendering halaman.

## Spesifikasi Sistem

Pengembangan sistem manajemen data dosen ini memerlukan spesifikasi perangkat lunak dan perangkat keras tertentu agar sistem dapat berjalan dengan optimal. Selain itu, sistem ini juga dirancang dengan pembagian peran pengguna yang memiliki hak akses berbeda sesuai dengan fungsinya.

### **Spesifikasi Perangkat Keras**

Sistem dapat dijalankan pada komputer dengan spesifikasi minimal prosesor *Intel Core Celeron* atau setara, *RAM* sebesar 4 GB, dan ruang penyimpanan kosong minimal 2 GB. Perangkat dengan spesifikasi ini sudah cukup untuk menjalankan web server lokal dan browser modern secara bersamaan.

### **Spesifikasi Perangkat Lunak**

Perangkat lunak yang dibutuhkan meliputi sistem operasi Windows 10 atau yang lebih baru, web server lokal (XAMPP versi terbaru), PHP versi 7.4 ke atas, serta MySQL atau MariaDB sebagai basis data. Sistem ini juga dapat dijalankan melalui peramban web seperti Google Chrome atau Mozilla Firefox.

### **Peran Pengguna Sistem**

Sistem dirancang untuk mendukung beberapa peran pengguna, yaitu admin dan dosen. Admin memiliki akses penuh terhadap seluruh modul sistem, termasuk menambah, mengedit, dan menghapus data dosen. Sementara itu, dosen hanya memiliki akses terbatas, yaitu melihat data pribadinya dan mengajukan permintaan perubahan data kepada admin.

# BAB IV PERANCANGAN DAN IMPLEMENTASI SISTEM

## Diagram Perancangan Sistem

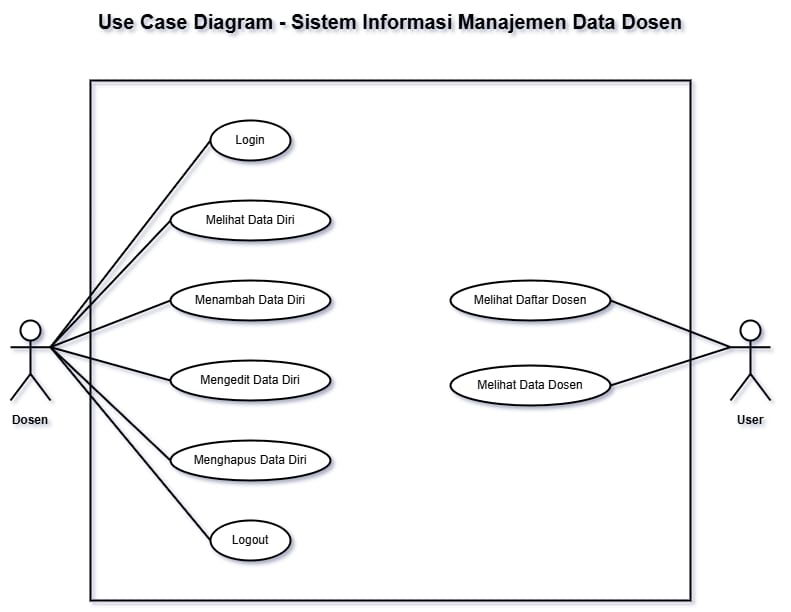
Perancangan sistem merupakan tahapan penting dalam proses pengembangan perangkat lunak, yang bertujuan untuk menggambarkan bagaimana sistem akan bekerja sebelum diimplementasikan secara nyata. Tahapan ini membantu pengembang memahami hubungan antara pengguna dan sistem, struktur data yang akan digunakan, serta alur proses yang terjadi di dalam sistem.

Pada sistem manajemen data dosen ini, proses perancangan dilakukan dengan menggunakan beberapa jenis diagram, antara lain use case diagram, entity relationship diagram (ERD), dan diagram alur sistem. Diagram-diagram ini memberikan gambaran menyeluruh mengenai interaksi pengguna dengan sistem, struktur basis data yang digunakan, serta aliran data atau logika sistem secara umum.

### **Use Case Diagram**

Use case diagram digunakan untuk menggambarkan interaksi antara pengguna (aktor) dengan sistem yang dikembangkan. Diagram ini memperlihatkan fungsi-fungsi utama yang dapat dilakukan oleh masing-masing aktor serta bagaimana mereka berinteraksi dengan sistem. Pada sistem manajemen data dosen ini, terdapat tiga jenis pengguna utama, yaitu Admin, Dosen, dan Pengguna Umum. Masing-masing aktor memiliki hak akses dan fungsionalitas yang berbeda sesuai dengan perannya dalam sistem.

**Gambar 4.1 Use Case Diagram Sistem Manajemen Data Dosen** berikut menampilkan hubungan antara kedua aktor tersebut dengan fitur-fitur yang tersedia di sistem. Dosen dapat login, melihat serta mengubah data pribadinya, dan mengelola informasi keahliannya. Sementara itu, pengguna umum tidak perlu melakukan login dan hanya dapat melihat daftar dosen, jadwal mengajar, serta mengakses halaman utama sistem.

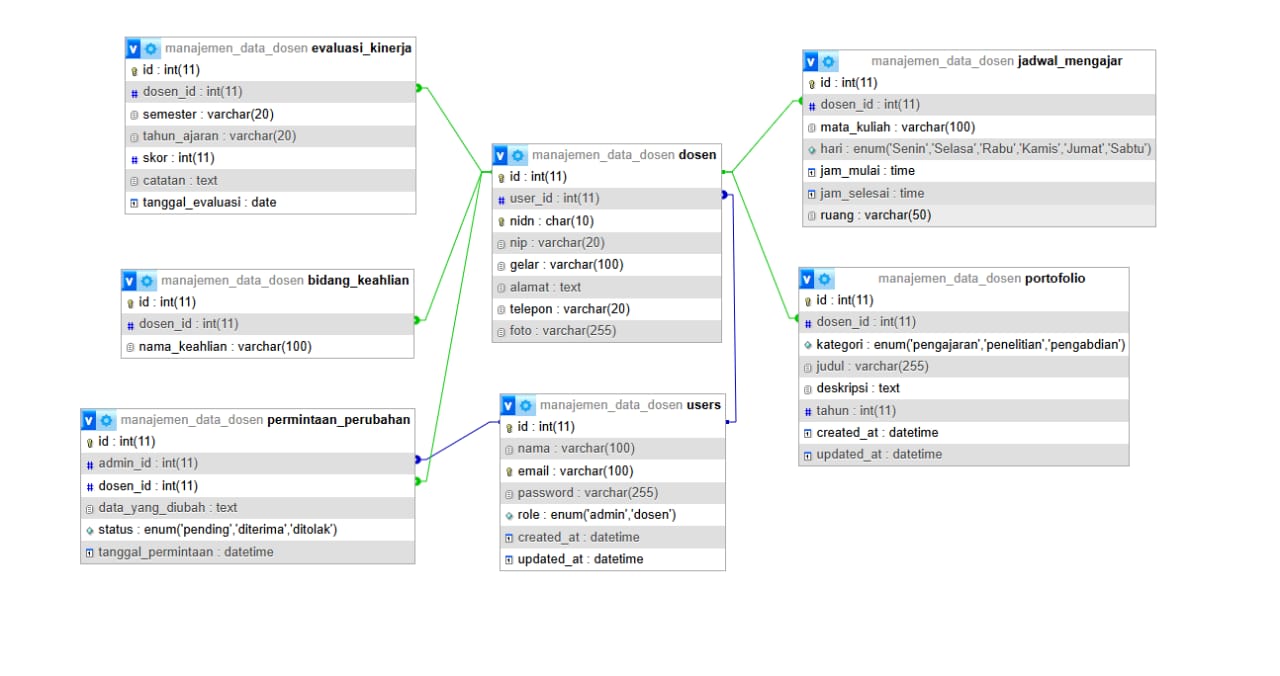


**Gambar 4.1 Use Case Diagram Sistem Manajemen Data Dosen**

1. **Dosen** dapat melakukan login menggunakan email dan password untuk melihat serta memperbarui data pribadinya. Hal ini memungkinkan dosen untuk menjaga akurasi informasi secara mandiri.
2. **Pengguna Umum** (seperti mahasiswa atau pengunjung website) tidak perlu melakukan login. Mereka dapat mengakses halaman dashboard awal (index.php), melihat daftar dosen yang tersedia, serta melihat jadwal dosen. Halaman awal juga menyediakan tombol login bagi admin dan dosen yang ingin mengakses sistem lebih lanjut.

### **Entity Relationship Diagram (ERD)**

Entity Relationship Diagram (ERD) digunakan untuk menggambarkan struktur basis data pada sistem manajemen data dosen. Diagram ini memvisualisasikan entitas utama dalam sistem, atribut-atribut penting, serta relasi antar entitas melalui foreign key. ERD memudahkan pengembang dalam memahami bagaimana data disimpan, saling terhubung, dan dikelola secara menyeluruh oleh sistem.



**Gambar 4.2 Entity Relationship Diagram**

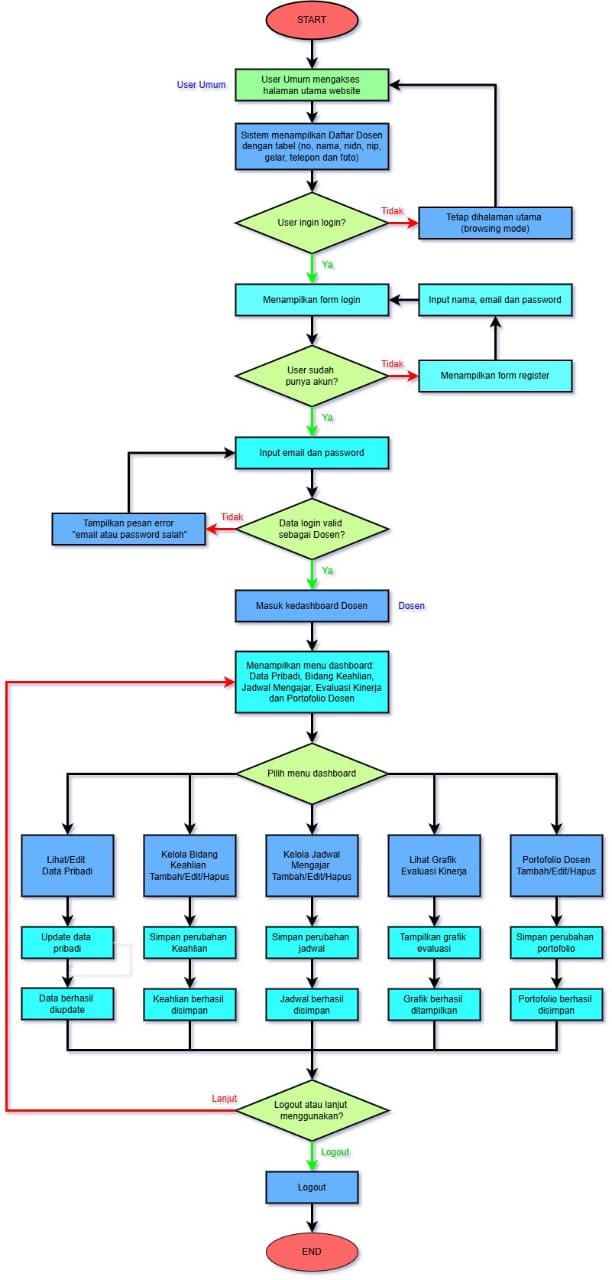
**Gambar 4.2 Entity Relationship Diagram** (ERD) Sistem Manajemen Data Dosen memperlihatkan enam entitas utama yang saling terhubung, yaitu: users, dosen, bidang\_keahlian, jadwal\_mengajar, evaluasi\_kinerja, dan permintaan\_perubahan. Penjelasan tiap entitas dan relasinya sebagai berikut:

1. **users**: Menyimpan data akun pengguna (admin dan dosen), terdiri dari id, nama, email, password, dan role. Relasi one-to-one dengan dosen, karena satu user hanya memiliki satu profil dosen (dan sebaliknya).
2. **dosen**: Menyimpan data pribadi dosen seperti nidn, nip, gelar, alamat, telepon, dan foto. Entitas ini menjadi pusat relasi ke beberapa tabel lain: jadwal\_mengajar, bidang\_keahlian, evaluasi\_kinerja, dan permintaan\_perubahan.
3. **bidang\_keahlian**: Relasi many-to-one ke dosen, karena satu dosen bisa punya banyak bidang keahlian. Terdiri dari id, dosen\_id, dan nama\_keahlian.
4. **jadwal\_mengajar**: Juga relasi many-to-one ke dosen. Menyimpan jadwal pengajaran seperti mata\_kuliah, hari, jam\_mulai, jam\_selesai, dan ruang.
5. **evaluasi\_kinerja**: Menyimpan hasil penilaian dosen, dengan atribut semester, tahun\_ajaran, skor, catatan, dan tanggal\_evaluasi. Satu dosen bisa punya banyak evaluasi (one-to-many).
6. **permintaan\_perubahan**: Menyimpan histori perubahan data yang diajukan oleh dosen dan ditinjau oleh admin. Berelasi dengan dosen dan users (admin) melalui foreign key dosen\_id dan admin\_id. Atribut lainnya meliputi data\_yang\_diubah, status (enum), dan tanggal\_permintaan.

Struktur ini dirancang agar fleksibel dalam pengelolaan data akademik dosen, sekaligus mendukung keamanan dan transparansi melalui mekanisme perizinan perubahan data. Dengan pendekatan relational database, sistem dapat memastikan integritas data dan efisiensi pengelolaan informasi di lingkungan akademik.

### **Diagram Alur Sistem**

Diagram alur sistem (flowchart) digunakan untuk menggambarkan aliran proses utama dalam sistem manajemen data dosen, mulai dari akses pengguna hingga pengelolaan berbagai fitur dalam dashboard. Diagram ini memudahkan pemahaman alur kerja aplikasi, khususnya bagaimana interaksi antara pengguna (baik user umum maupun dosen) dengan sistem dalam menjalankan fungsi-fungsi utama. Tampilan lengkap dari alur ini dapat dilihat pada **Gambar 4.3 Diagram Alur Sistem Manajemen Data Dosen.**



**Gambar 4.3 Diagram Alur Sistem Manajemen Data Dosen.**

Alur dimulai ketika user umum mengakses halaman utama website. Sistem akan menampilkan daftar dosen yang bisa diakses tanpa login. Jika user ingin login, maka sistem akan mengarahkan ke form login dan melakukan proses verifikasi akun. Bila user belum memiliki akun, maka akan ditampilkan form registrasi terlebih dahulu. Proses ini mengatur akses masuk ke dalam sistem dengan validasi yang ketat untuk menjaga keamanan dan memastikan hanya dosen yang memiliki hak akses ke dashboard.

Setelah login berhasil, user diarahkan ke dashboard dosen yang menampilkan menu utama: Data Pribadi, Bidang Keahlian, Jadwal Mengajar, Evaluasi Kinerja, dan Portofolio Dosen. Masing-masing menu mengarahkan ke halaman pengelolaan data yang berbeda. Dosen dapat memperbarui data pribadi, menambah atau mengedit bidang keahlian dan jadwal mengajar, melihat grafik evaluasi kinerja, serta mengelola portofolio Tri Dharma secara langsung melalui antarmuka yang telah disediakan.

Setiap proses update atau penyimpanan data akan divalidasi oleh sistem, dan sistem akan menampilkan notifikasi keberhasilan setelah data tersimpan. Di akhir sesi, dosen dapat memilih untuk logout atau melanjutkan penggunaan sistem. Flowchart ini menggambarkan bagaimana sistem berjalan secara berurutan, terstruktur, dan efisien, serta mendukung pengelolaan data akademik dosen secara digital dan modern.

## Struktur Folder Aplikasi CodeIgniter 4

Struktur folder pada sistem manajemen data dosen ini dibangun menggunakan framework CodeIgniter 4 (CI4) dengan pendekatan arsitektur Model-View-Controller (MVC). Struktur ini bertujuan untuk memisahkan antara logika aplikasi, antarmuka pengguna, dan pengelolaan data, sehingga sistem menjadi lebih terorganisir dan mudah dikelola.

Secara umum, folder inti dari proyek sistem ini terdiri atas:

1. app/: berisi semua komponen utama dari aplikasi. Di dalamnya terdapat:
   1. Controllers/: tempat seluruh pengendali logika aplikasi. Diatur berdasarkan role pengguna, seperti Admin/, Dosen/, dan juga Auth.php, PublicController.php.
   2. Models/: berisi file model seperti UserModel.php, DosenModel.php, EvaluasiModel.php, dan lainnya, yang berfungsi untuk mengelola query dan koneksi ke database.
   3. Views/: menyimpan semua antarmuka pengguna (HTML + PHP). Terdapat folder terpisah untuk admin/, dosen/, dan auth/.
   4. Config/: konfigurasi utama sistem seperti Routes.php, Database.php, dan konfigurasi lainnya.
   5. Filters/: terdapat RoleFilter.php untuk mengatur akses berdasarkan role pengguna.
   6. Database/Migrations/: berisi file migrasi untuk membangun tabel-tabel di database secara otomatis.
2. public/: merupakan folder root yang diakses melalui browser. File index.php berada di sini sebagai entry point aplikasi. Terdapat juga folder uploads/ untuk menyimpan file seperti foto dosen.
3. vendor/: berisi library pihak ketiga yang diinstal melalui Composer.
4. File penting lainnya: seperti .env untuk konfigurasi environment, composer.json untuk manajemen dependensi, dan spark sebagai command-line tool CI4.

Struktur folder ini mencerminkan praktik terbaik dalam pengembangan aplikasi web dengan CI4. Pengelompokan berdasarkan role, serta pemisahan antar logika dan tampilan, memberikan kemudahan dalam pemeliharaan kode dan pengembangan lebih lanjut.

## Desain Antarmuka Pengguna

Desain antarmuka pengguna (user interface/UI) dalam sistem manajemen data dosen dirancang dengan mempertimbangkan kemudahan penggunaan, konsistensi tampilan, dan aksesibilitas. Sistem ini dibangun menggunakan kombinasi HTML, CSS, dan framework Bootstrap 5, serta mengikuti prinsip responsive design agar dapat diakses melalui berbagai perangkat.

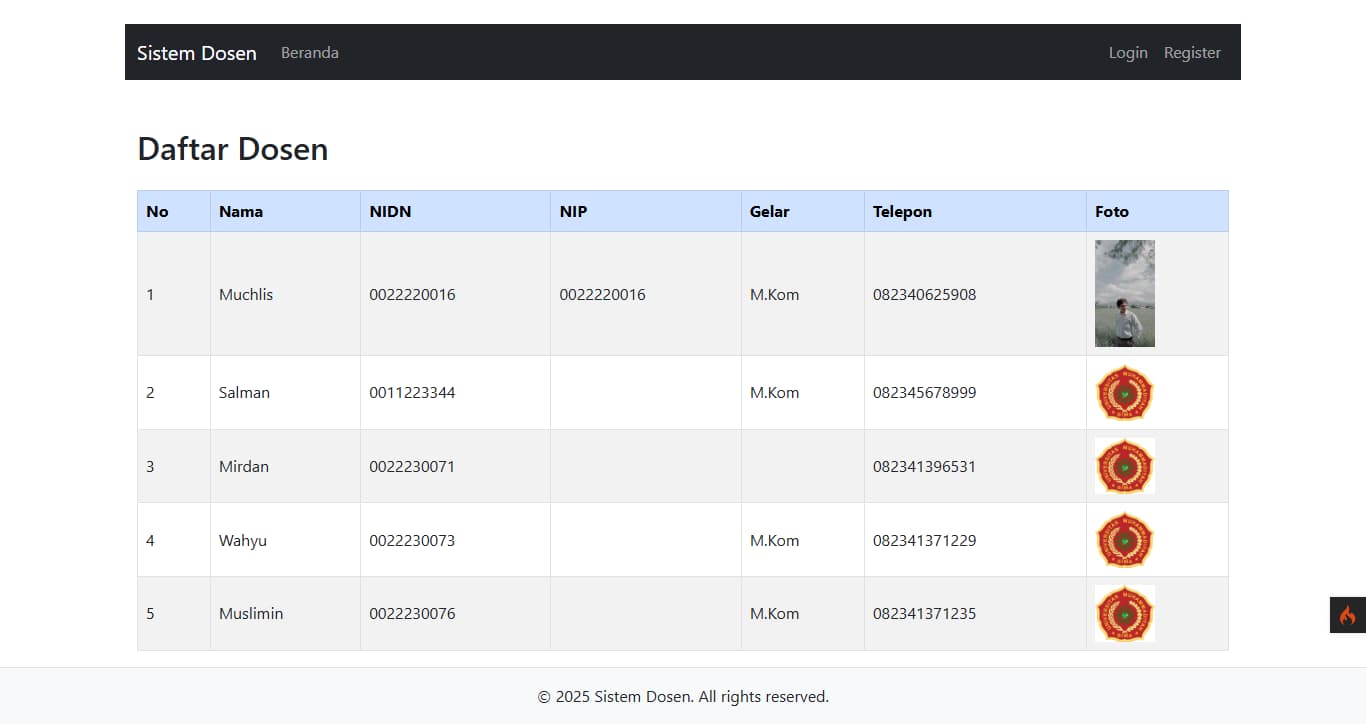
Antarmuka pengguna dibagi menjadi beberapa tampilan utama berdasarkan peran pengguna, yaitu halaman umum (dashboard publik), login, dashboard admin, dan dashboard dosen. Berikut adalah penjelasan dari masing-masing tampilan:

### **Halaman Dashboard (Pengguna Umum)**

Halaman dashboard pengguna umum merupakan tampilan awal dari sistem yang dapat diakses tanpa melakukan login. Fungsinya adalah menyediakan informasi terbuka kepada publik, khususnya daftar dosen aktif yang tercatat dalam sistem. Halaman ini juga menjadi pintu masuk utama menuju fitur login dan registrasi bagi pengguna internal seperti admin maupun dosen.

Sebagaimana dapat dilihat pada **Gambar 4.4 Tampilan Halaman Dashboard Pengguna Umum**, sistem menampilkan tabel data dosen yang terdiri atas kolom nomor, nama, NIDN, NIP, gelar akademik, nomor telepon, serta foto. Data ditampilkan secara terstruktur dan mudah dibaca, sehingga memudahkan pengunjung untuk memperoleh informasi dosen secara cepat dan jelas.

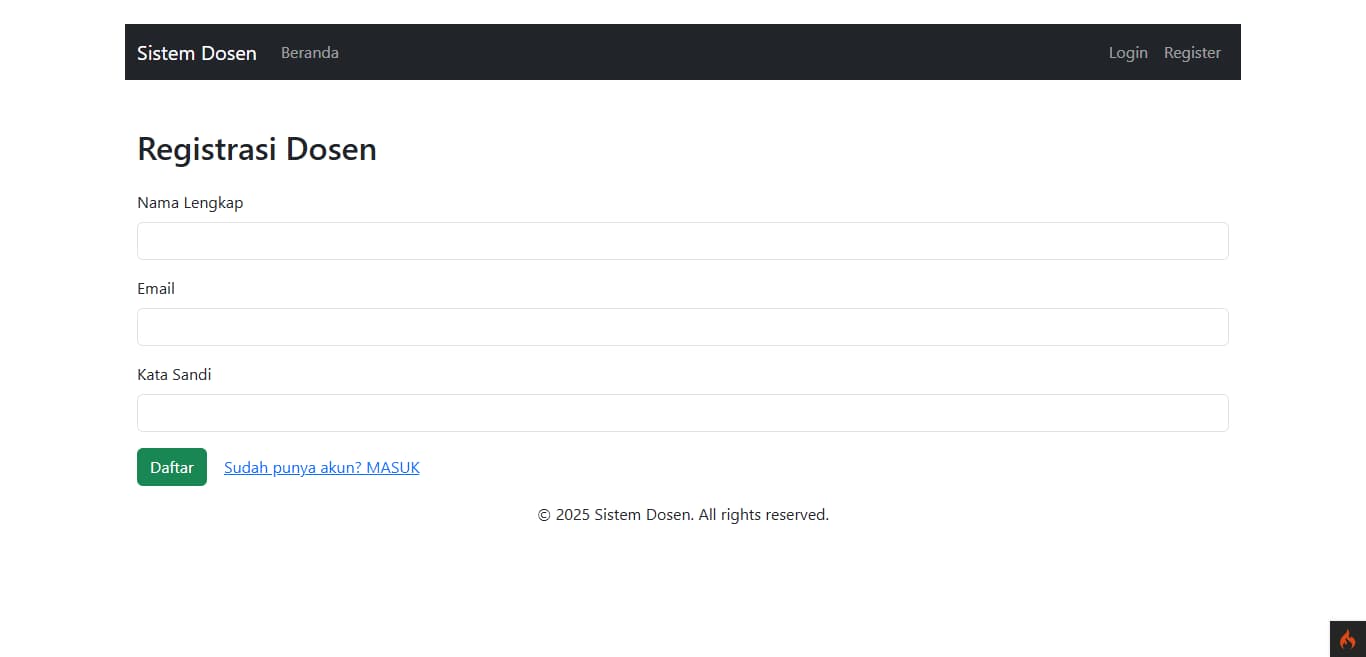
Selain itu, pada bagian atas halaman terdapat menu navigasi sederhana yang memuat tautan menuju halaman login dan registrasi. Sementara pada bagian bawah halaman ditampilkan footer dengan informasi hak cipta sistem. Desain antarmuka ini bersifat minimalis dan responsif, dengan tujuan agar pengguna dari berbagai perangkat tetap dapat mengakses informasi dosen secara optimal tanpa gangguan elemen visual yang berlebihan.



**Gambar 4.4 Tampilan Halaman Dashboard Pengguna Umum**

### **Halaman Registrasi**

Halaman registrasi digunakan untuk mendaftarkan akun baru bagi dosen yang belum memiliki akses ke dalam sistem. Form registrasi terdiri dari tiga input utama, yaitu Nama Lengkap, Email, dan Kata Sandi. Setelah semua data diisi dengan benar, dosen dapat menekan tombol Daftar untuk menyelesaikan proses pendaftaran. Jika dosen sudah memiliki akun, tersedia tautan “Sudah punya akun? MASUK” yang mengarahkan kembali ke halaman login. Desain halaman ini dibuat minimalis dan fokus pada fungsionalitas agar proses registrasi berlangsung cepat dan mudah dipahami oleh pengguna. Tampilan antarmuka ini dapat dilihat pada **Gambar 4.5 Halaman Registrasi Dosen**.



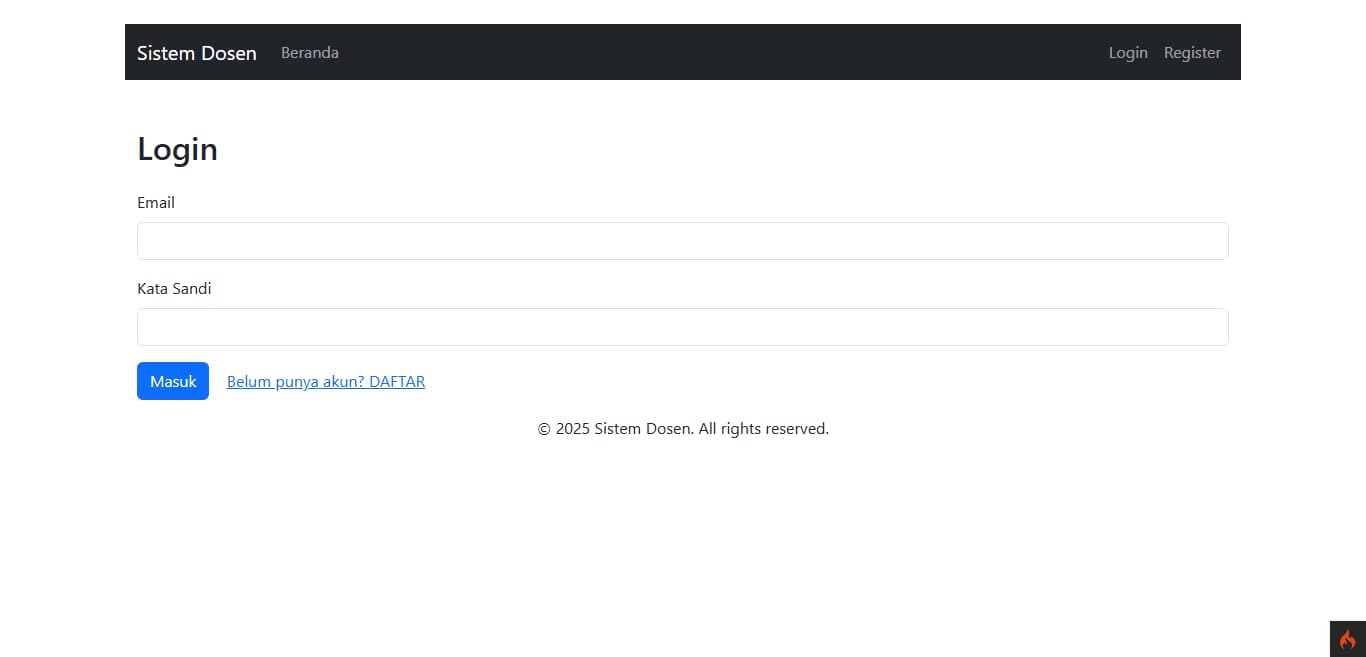
**Gambar 4.5 Halaman Registrasi Dosen**.

### **Halaman Login**

Halaman login merupakan pintu masuk utama bagi pengguna terdaftar, yaitu dosen, untuk mengakses fitur internal sistem. Halaman ini terdiri atas dua input utama, yaitu email dan kata sandi, yang kemudian akan diverifikasi oleh sistem untuk menentukan hak akses pengguna.

Sebagaimana ditunjukkan pada **Gambar 4.6 Tampilan Halaman Login**, desain antarmuka login dibuat sesederhana mungkin dengan fokus pada fungsi utama: autentikasi. Tampilan form login bersih dan minimalis, dengan tombol “Masuk” berwarna biru yang menonjol di bagian bawah. Terdapat juga tautan tambahan “Belum punya akun? DAFTAR” yang mengarahkan pengguna ke halaman registrasi akun baru (jika fitur tersebut diaktifkan).

Sistem akan memverifikasi data yang dimasukkan, dan jika sesuai, pengguna akan diarahkan ke dashboard masing-masing sesuai dengan peran yang tersimpan dalam database. Jika terjadi kesalahan input atau akun tidak ditemukan, maka sistem akan memberikan umpan balik berupa pesan kesalahan. Hal ini bertujuan untuk menjaga keamanan sistem dan memastikan bahwa hanya pengguna yang berwenang yang dapat mengakses fitur-fitur internal.

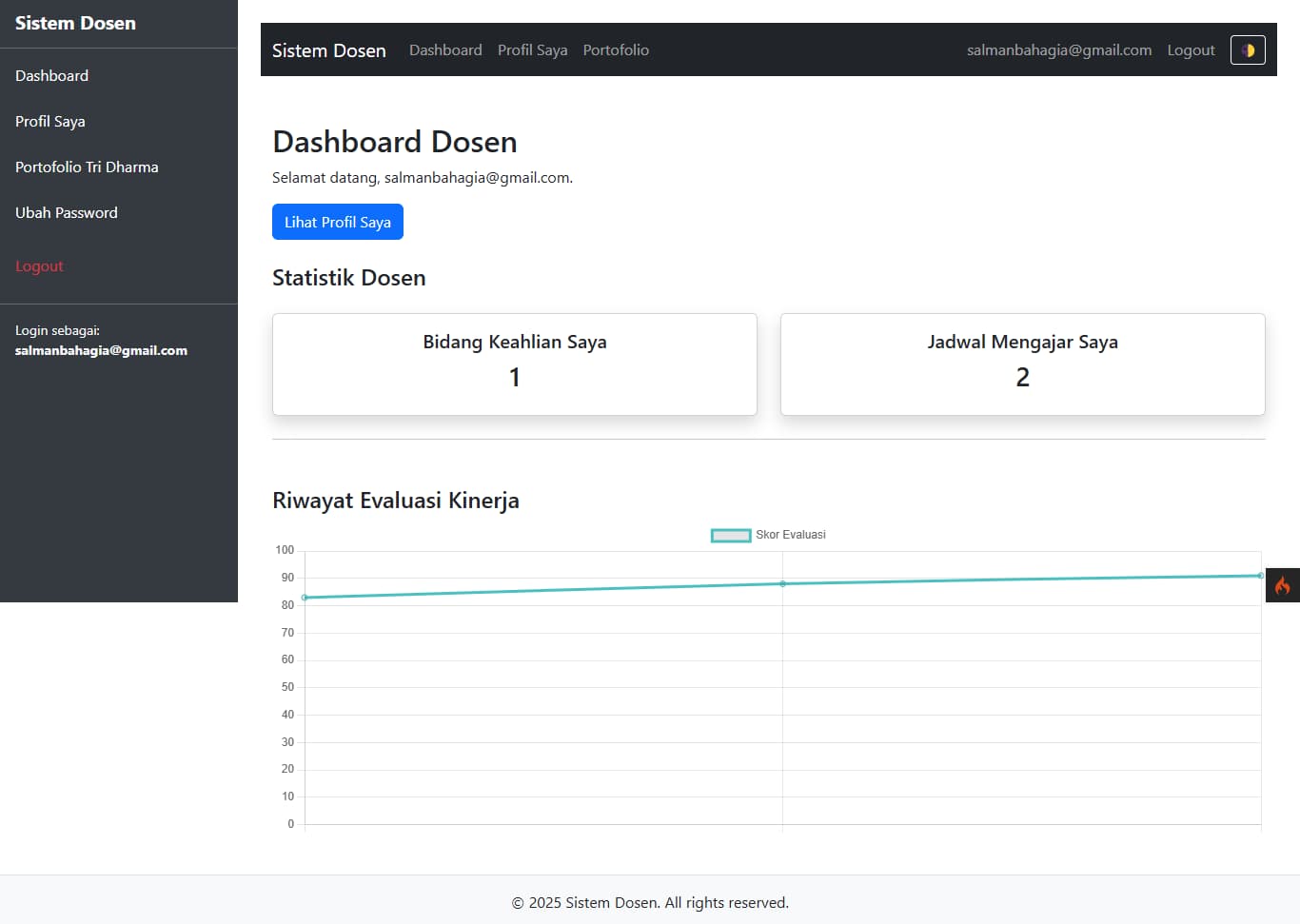


**Gambar 4.6 Tampilan Halaman Login**

### **Halaman Dashboard Dosen**

Dashboard dosen merupakan halaman utama yang ditampilkan setelah pengguna dengan peran dosen berhasil login ke dalam sistem. Seperti terlihat pada **Gambar 4.7 Tampilan Dashboard Dosen**, tampilan ini menyajikan informasi profil lengkap seperti nama, email, NIDN, NIP, gelar, alamat, dan nomor telepon, disertai dengan foto profil. Di bawahnya terdapat dua panel statistik sederhana yang menunjukkan total bidang keahlian serta total jadwal mengajar yang dimiliki oleh dosen tersebut.

Selain itu, dosen dapat menambahkan atau mengelola bidang keahlian dan jadwal mengajar melalui form input yang tersedia. Fitur aksi seperti Edit dan Hapus memungkinkan dosen mengatur datanya sendiri secara mandiri. Di bagian paling bawah terdapat placeholder untuk grafik evaluasi kinerja dosen yang akan aktif bila data penilaian sudah tersedia. Desain ini mengusung prinsip minimalis dan responsif, agar mudah digunakan di berbagai perangkat.



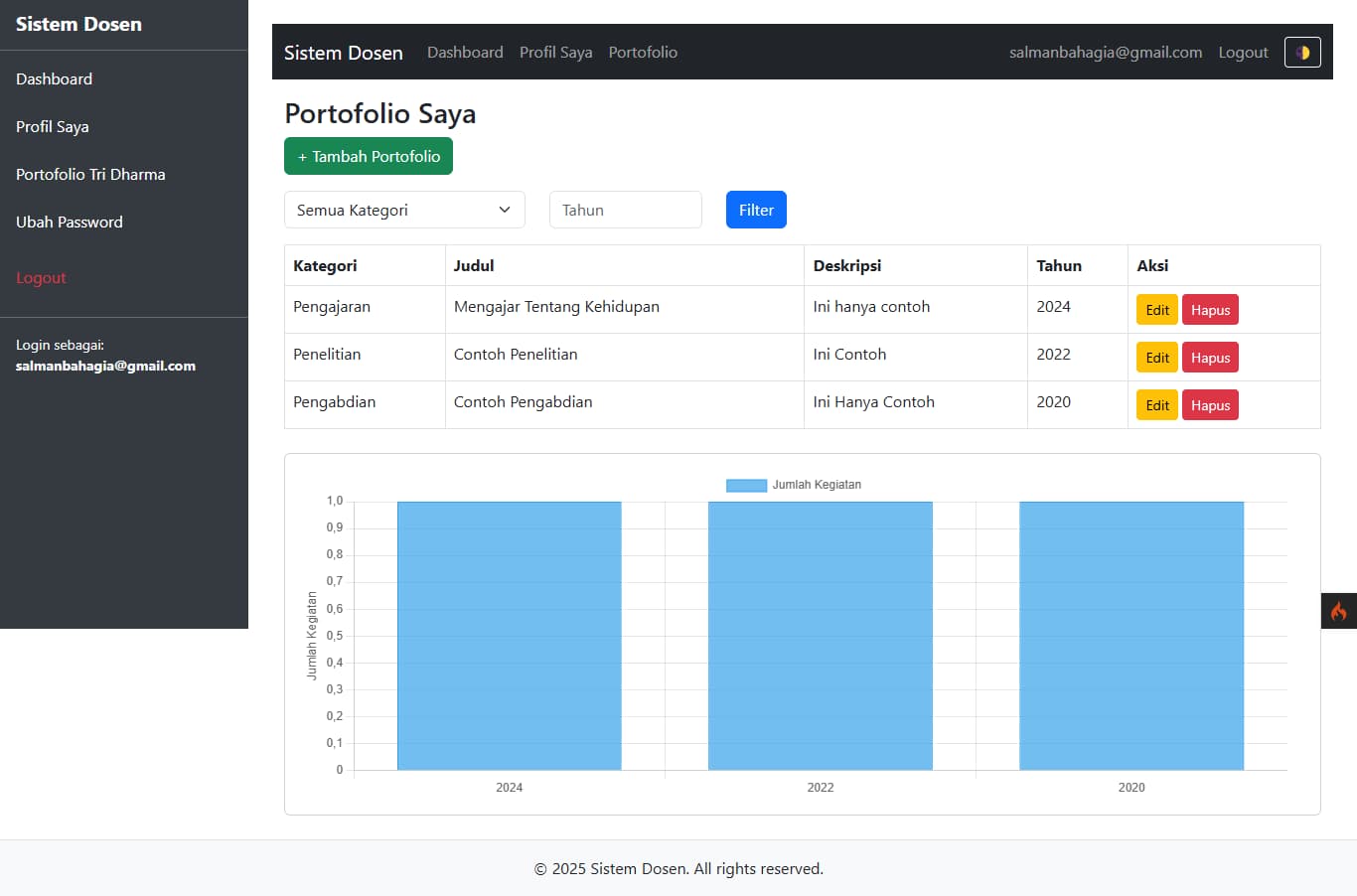
**Gambar 4.7 Tampilan Dashboard Dosen**

### **Pengelolaan Portofolio Dosen**

Fitur portofolio merupakan bagian penting dalam sistem yang dirancang untuk mencatat aktivitas dosen dalam tiga aspek Tri Dharma Perguruan Tinggi, yaitu Pengajaran, Penelitian, dan Pengabdian kepada Masyarakat. Data portofolio mencakup kategori kegiatan, judul, deskripsi, dan tahun pelaksanaan. Dosen dapat menambahkan entri baru menggunakan tombol “+ Tambah Portofolio”, serta mengedit dan menghapus data yang sudah ada secara mandiri. Seluruh data ditampilkan dalam tabel responsif yang mudah dibaca dan dikelola. Fitur ini dibagi ke dalam tiga halaman utama, yaitu:

1. **Halaman Daftar Portofolio**

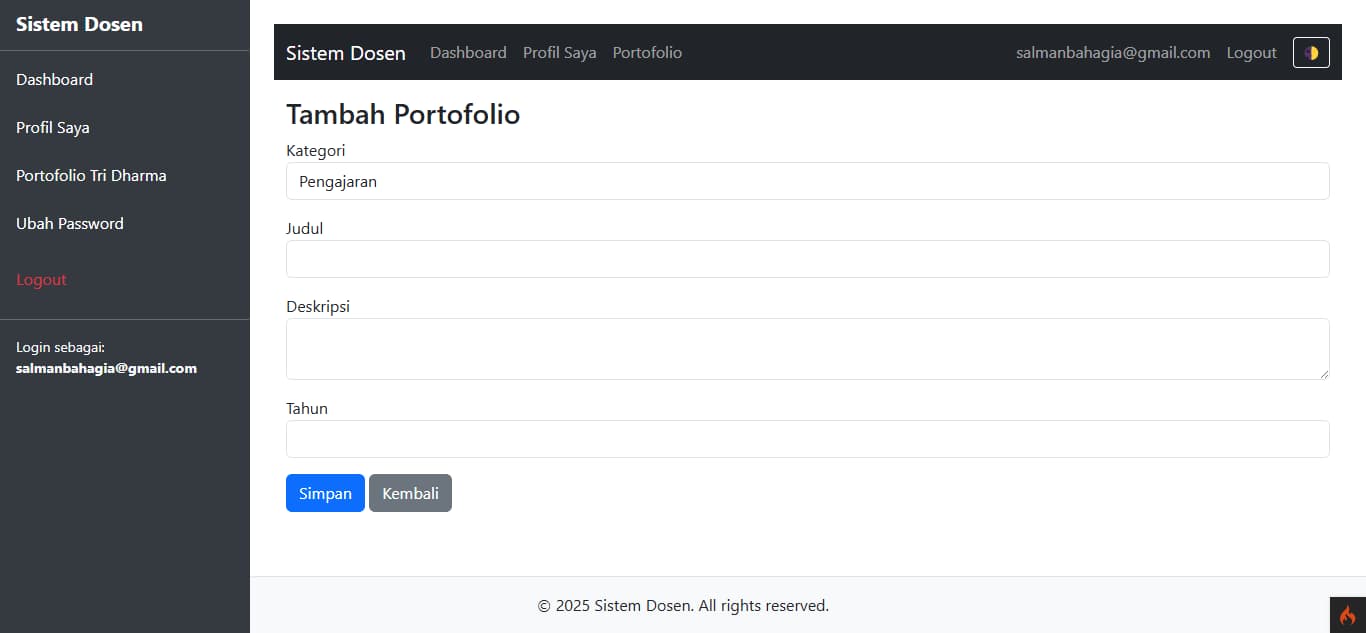
Halaman ini berfungsi untuk menampilkan seluruh data portofolio dosen dalam bentuk tabel yang mencakup kategori kegiatan (Pengajaran, Penelitian, atau Pengabdian), judul, deskripsi, tahun pelaksanaan, serta tombol aksi untuk mengedit dan menghapus data. Selain itu, tersedia fitur filter berdasarkan kategori dan tahun untuk memudahkan pencarian data tertentu. Pada bagian bawah, ditampilkan grafik jumlah kegiatan per tahun yang memberikan visualisasi produktivitas dosen. Tampilan antarmuka halaman ini dapat dilihat pada **Gambar 4.8 Tampilan Halaman Daftar Portofolio Dosen**.



**Gambar 4.8 Tampilan Halaman Daftar Portofolio Dosen**.

1. **Halaman Tambah Portofolio:**

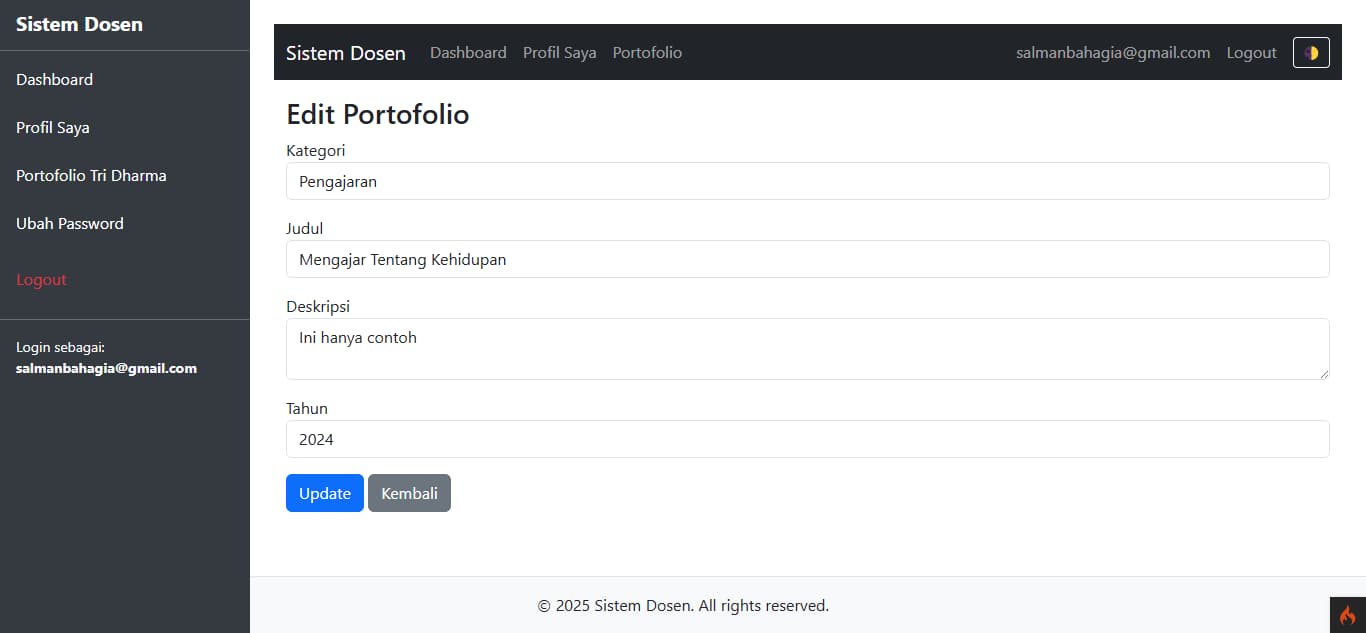
Halaman ini digunakan oleh dosen untuk menambahkan data portofolio baru ke dalam sistem. Formulir input terdiri dari kolom kategori kegiatan, judul, deskripsi, dan tahun pelaksanaan. Setelah semua data diisi, dosen dapat menyimpan entri baru melalui tombol Simpan, atau membatalkan proses menggunakan tombol Kembali. Halaman ini didesain dengan layout sederhana dan fokus pada kemudahan input, agar proses pencatatan aktivitas akademik dapat dilakukan secara cepat dan efisien. Tampilan halaman ini ditunjukkan pada **Gambar 4.9 Halaman Tambah Portofolio Dosen**.



**Gambar 4.9 Halaman Tambah Portofolio Dosen**.

1. **Halaman Edit Portofolio**

Halaman ini memungkinkan dosen untuk melakukan pembaruan data portofolio yang sudah pernah ditambahkan sebelumnya. Formulir pengeditan mencakup kolom input untuk kategori, judul kegiatan, deskripsi, dan tahun pelaksanaan. Dosen dapat mengubah informasi yang ada, lalu menyimpan perubahan dengan menekan tombol Update, atau membatalkan proses melalui tombol Kembali. Antarmuka halaman ini dirancang minimalis dan user-friendly, sehingga memudahkan pengguna dalam memperbarui data secara cepat dan efisien. Tampilan halaman edit ini dapat dilihat pada **Gambar 4.10 Halaman Edit Portofolio Dosen**.



**Gambar 4.10 Halaman Edit Portofolio Dosen**.

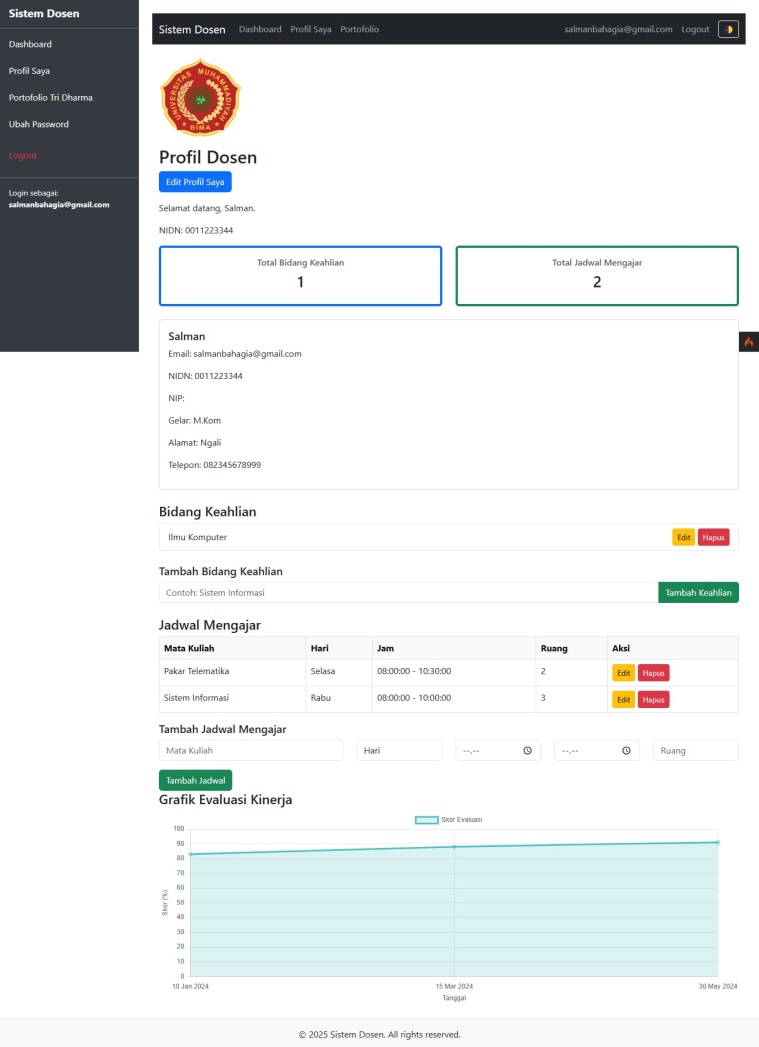
Di bagian bawah halaman daftar, sistem juga menampilkan grafik jumlah kegiatan portofolio berdasarkan tahun. Visualisasi ini menggunakan Chart.js untuk membantu dosen memahami tren aktivitas akademiknya dari waktu ke waktu. Proses pengolahan data dilakukan menggunakan pendekatan MVC dalam CodeIgniter 4, dan setiap form input dilengkapi validasi. Dengan fitur ini, sistem berperan sebagai alat dokumentasi digital yang mendukung akuntabilitas dan pelaporan aktivitas dosen secara terstruktur.

### **Pengelolaan Profil Dosen**

Sistem manajemen data dosen ini menyediakan fitur khusus yang memungkinkan setiap dosen untuk mengelola data pribadinya secara mandiri. Pengelolaan ini meliputi informasi profil, jadwal mengajar, serta bidang keahlian yang dimiliki. Seluruh fitur tersebut dirancang dengan antarmuka yang sederhana namun fungsional, sehingga dosen dapat memperbarui data secara langsung tanpa perlu campur tangan admin. Berikut ini merupakan tampilan dan penjelasan dari masing-masing fitur pengelolaan data yang tersedia dalam sistem.

1. **Halaman Profil Dosen**

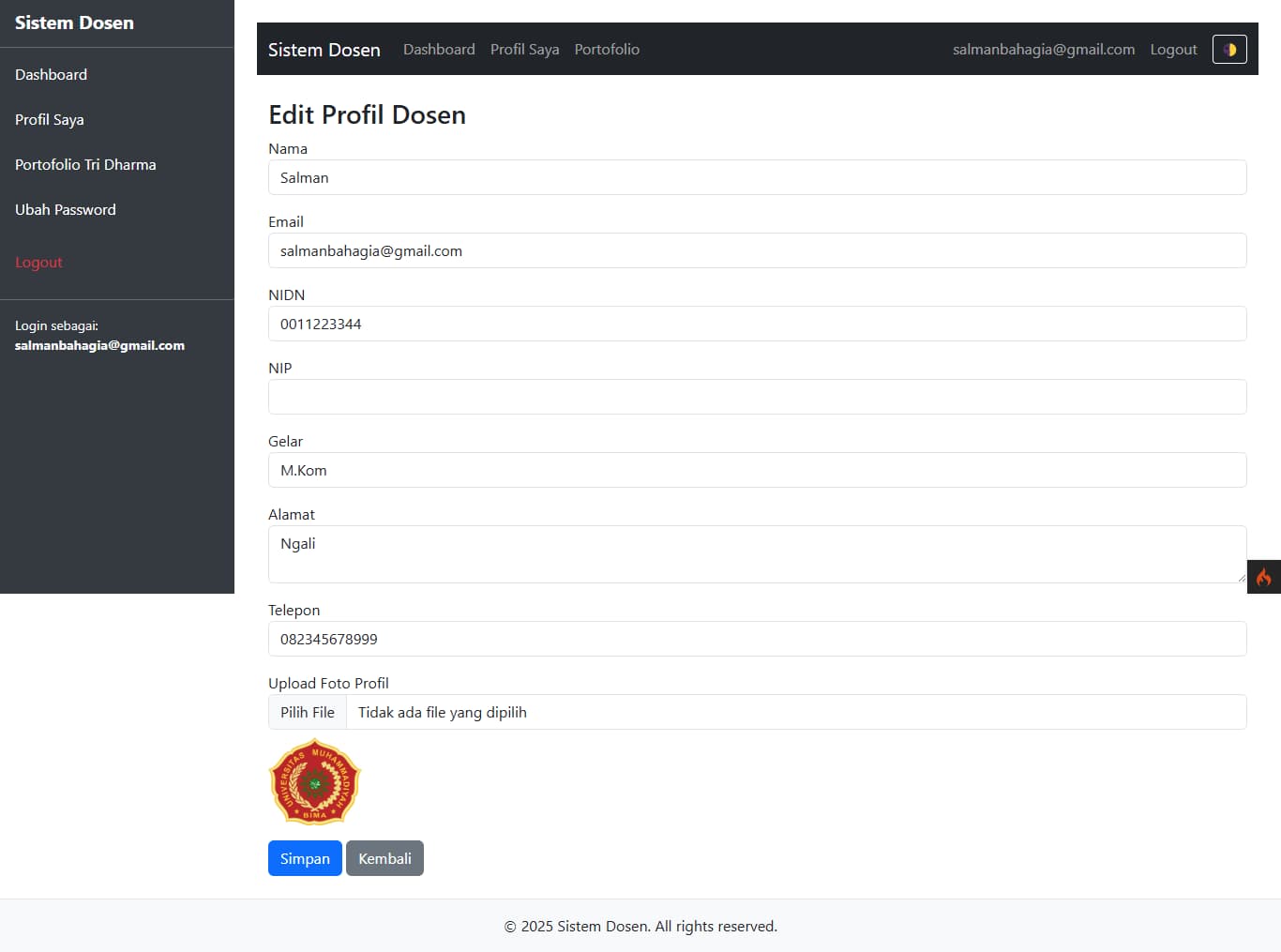
Halaman ini menampilkan informasi pribadi dosen secara lengkap, termasuk nama, email, NIDN, NIP, gelar, alamat, dan nomor telepon. Di bagian atas halaman, terdapat tombol **Edit Profil Saya** yang mengarahkan dosen ke form pengeditan data pribadi. Selain itu, sistem juga menampilkan jumlah total bidang keahlian dan jadwal mengajar dalam bentuk kartu statistik sederhana. Tampilan halaman ini dapat dilihat pada **Gambar 4.11 Halaman Profil Dosen**.



**Gambar 4.11 Halaman Profil Dosen**.

1. **Halaman Edit Profil Dosen**

Halaman ini memungkinkan dosen untuk memperbarui data pribadinya secara mandiri. Form input yang tersedia meliputi nama, email, NIDN, NIP, gelar akademik, alamat, nomor telepon, serta fitur unggah foto profil. Setelah melakukan perubahan, dosen dapat menekan tombol Simpan untuk menyimpan data atau Kembali untuk membatalkan perubahan. Antarmuka yang sederhana dan jelas membantu memastikan proses pengeditan berjalan lancar tanpa kebingungan. Tampilan halaman ini ditunjukkan pada **Gambar 4.12 Halaman Edit Profil Dosen**.



**Gambar 4.12 Halaman Edit Profil Dosen**.

## Implementasi Fitur Sistem

Pada bagian ini dijelaskan mengenai proses implementasi fitur-fitur utama dalam sistem manajemen data dosen yang telah dikembangkan. Implementasi dilakukan menggunakan framework CodeIgniter 4 dengan pola arsitektur *Model-View-Controller* (MVC). Setiap fitur dikembangkan berdasarkan hasil perancangan sebelumnya dan telah diuji untuk memastikan berfungsi sesuai kebutuhan pengguna.

Fitur-fitur inti yang diimplementasikan meliputi proses login multi-user (admin dan dosen), pengelolaan data dosen, manajemen bidang keahlian dan jadwal mengajar oleh dosen, serta visualisasi penilaian kinerja dosen dalam bentuk grafik. Penjelasan masing-masing fitur beserta tampilan dan mekanismenya dapat dilihat pada subbagian berikut.

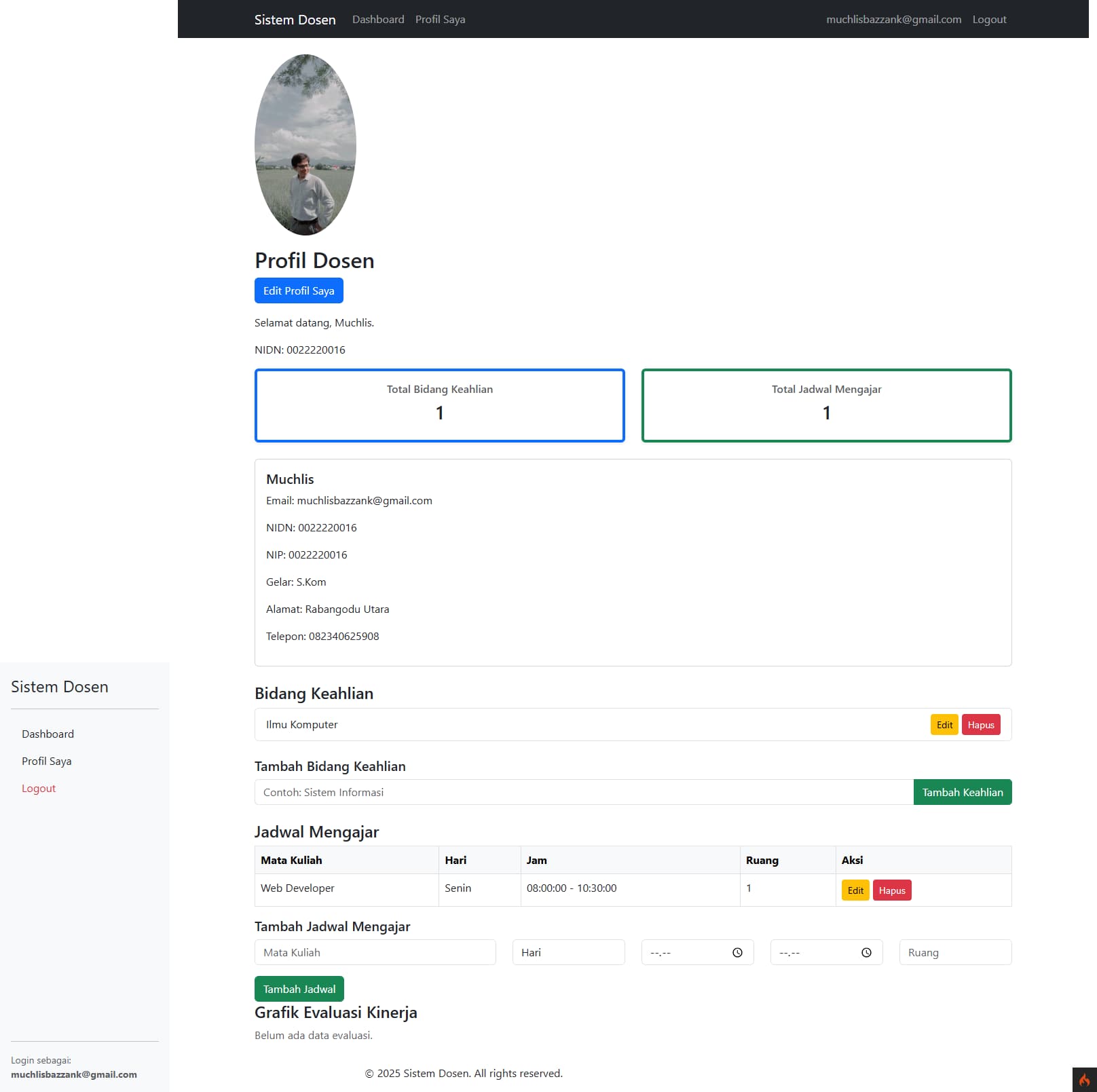
### **Pengelolaan Data Dosen**

Fitur pengelolaan data dosen dalam sistem ini diimplementasikan untuk memberikan keleluasaan bagi **dosen** dalam mengelola informasi pribadinya secara mandiri. Fitur ini dapat diakses setelah dosen berhasil login, dan disediakan melalui dashboard dosen yang menampilkan data pribadi secara lengkap serta opsi untuk melakukan pengeditan.

Sebagaimana terlihat pada **Gambar 4.13 Tampilan Form Profil Dosen**, sistem menampilkan data seperti nama, NIDN, NIP, gelar, alamat, nomor telepon, dan email, lengkap dengan foto profil dosen yang ditampilkan secara visual. Dosen dapat menekan tombol Edit Profil Saya untuk membuka form input yang memungkinkan pembaruan data secara langsung. Setiap perubahan data akan dikirim ke server melalui controller yang terhubung dengan model DosenModel.php, lalu disimpan ke basis data setelah divalidasi.

Desain form dibuat sederhana dan intuitif untuk meminimalkan kesalahan input. Proses edit data ini memastikan bahwa setiap dosen dapat bertanggung jawab atas akurasi datanya masing-masing tanpa perlu menunggu validasi manual dari pihak admin.

Fitur ini bertujuan untuk meningkatkan efisiensi pengelolaan data dan memberikan otonomi lebih kepada dosen dalam memelihara informasi pribadinya di dalam sistem.



**Gambar 4.13 Tampilan Form Profil Dosen**

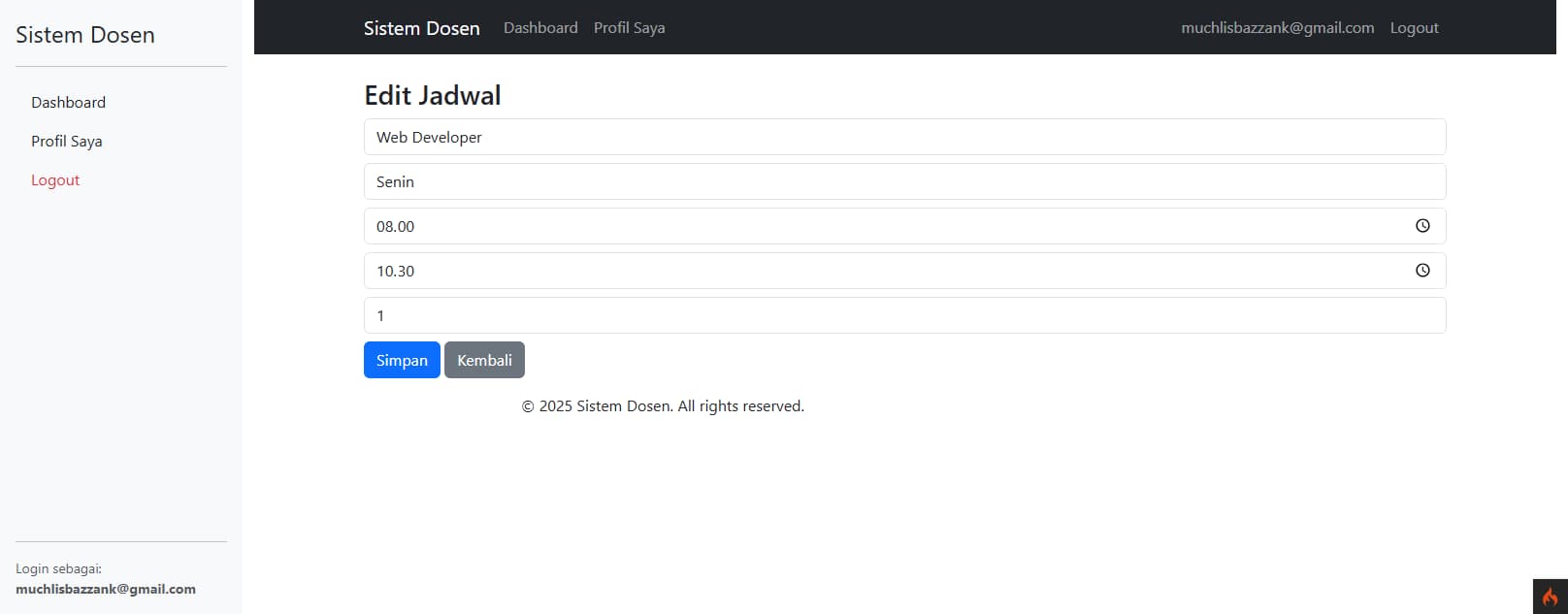
### **Fitur bidang keahlian dan jadwal mengajar**

Sistem ini menyediakan fitur yang memungkinkan dosen untuk mengelola data akademiknya secara mandiri, yaitu melalui pengelolaan bidang keahlian dan jadwalmengajar. Kedua fitur tersebut tersedia dalam dashboard dosen, dan dapat diakses setelah proses login berhasil.

Fitur bidang keahlian memungkinkan dosen untuk menambahkan keahlian akademik yang dikuasai, seperti Sistem Informasi, Jaringan Komputer, atau Rekayasa Perangkat Lunak. Data ini ditambahkan melalui form input sederhana, dan ditampilkan dalam bentuk tabel dinamis. Meskipun form input bidang keahlian hanya terdiri dari satu kolom (nama keahlian), sistem mengizinkan penambahan data secara berulang tanpa batas, serta dilengkapi dengan tombol aksi seperti Hapus untuk pengelolaan data.

Sementara itu, fitur jadwal mengajar memiliki struktur input yang lebih kompleks dan mencakup beberapa kolom, seperti nama mata kuliah, hari, jam mulai, jam selesai, dan ruang kelas. Data yang diinput akan ditampilkan dalam tabel yang memudahkan dosen untuk melihat dan memodifikasi jadwalnya. Implementasi dari fitur ini dilakukan melalui controller JadwalController.php dan model JadwalModel.php, dengan validasi pada setiap form input sebelum data disimpan.

Sebagaimana ditunjukkan pada **Gambar 4.14 Tampilan Jadwal Mengajar Dosen**, form input dirancang responsif dan sederhana untuk memastikan kemudahan penggunaan oleh dosen. Fitur ini menjadi salah satu elemen penting dalam mendukung akurasi informasi yang ditampilkan di halaman publik dan mendukung pengelolaan data akademik yang fleksibel.



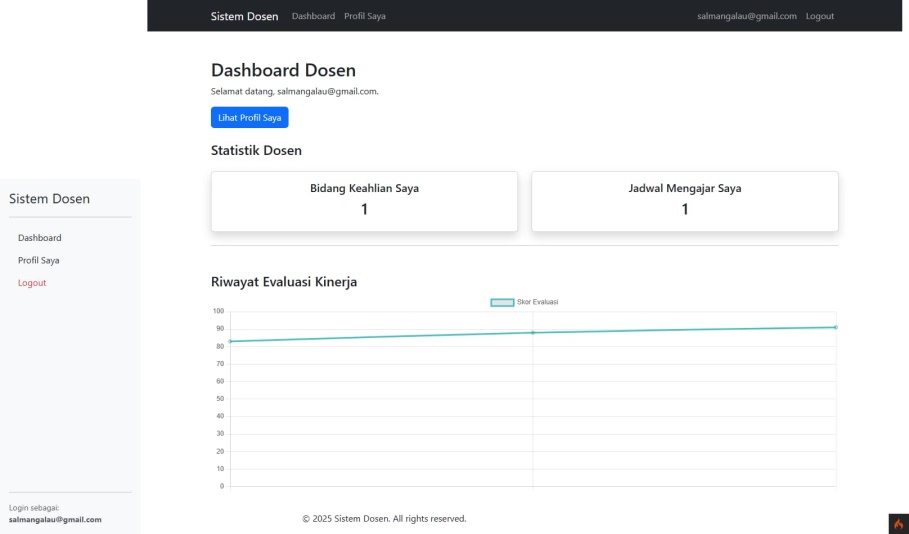
**Gambar 4.14 Tampilan Jadwal Mengajar Dosen**

### **Tampilan Grafik Evaluasi Kinerja Dosen**

Fitur grafik evaluasi kinerja dosen ditampilkan pada bagian bawah dashboard dan berfungsi untuk memberikan visualisasi terhadap hasil penilaian kinerja yang telah dilakukan. Tujuan utama dari fitur ini adalah menyajikan informasi performa dosen secara kuantitatif dalam bentuk grafik yang mudah dibaca dan dipahami.

Sebagaimana ditunjukkan pada **Gambar 4.15 Tampilan Grafik Evaluasi Kinerja Dosen**, sistem menampilkan grafik garis yang merepresentasikan skor evaluasi dosen berdasarkan waktu atau periode tertentu. Setiap titik pada grafik menunjukkan nilai evaluasi yang tercatat dalam database, dengan sumbu horizontal merepresentasikan waktu atau sesi penilaian, dan sumbu vertikal menunjukkan skor evaluasi. Grafik ini ditampilkan secara otomatis ketika data penilaian tersedia, dan akan terus diperbarui seiring bertambahnya data baru.

Fitur ini memanfaatkan pustaka visualisasi frontend seperti Chart.js untuk menggambar grafik berdasarkan data yang diperoleh dari tabel evaluasi\_kinerja. Seluruh data diakses melalui Evaluasi Model.php dan dikontrol melalui controller yang sesuai pada sisi dosen. Dengan adanya grafik ini, dosen dapat memantau perkembangan performa akademiknya dari waktu ke waktu secara langsung melalui sistem.



**Gambar 4.15 Tampilan Grafik Evaluasi Kinerja Dosen**

## Pengujian Sistem

### **Tujuan Pengujian**

Pengujian sistem dilakukan untuk memastikan bahwa seluruh fitur dan fungsi yang telah diimplementasikan dalam sistem manajemen data dosen berjalan sesuai dengan kebutuhan dan harapan pengguna. Dengan melakukan pengujian, pengembang dapat memverifikasi bahwa sistem tidak hanya dapat dijalankan secara teknis, tetapi juga memberikan keluaran (output) yang benar, akurat, dan dapat diandalkan.

Selain itu, pengujian bertujuan untuk mengidentifikasi kemungkinan kesalahan (bug) atau ketidaksesuaian pada sistem, baik dari sisi logika proses, tampilan antarmuka, maupun validasi input. Hasil dari proses pengujian ini dapat dijadikan dasar dalam proses evaluasi dan penyempurnaan sistem agar dapat digunakan secara optimal oleh pengguna akhir, seperti dosen, admin, dan pengguna umum.

### **Metode Pengujian**

Metode pengujian yang digunakan dalam sistem manajemen data dosen ini adalah Black Box Testing, yaitu metode pengujian yang berfokus pada pengujian fungsionalitas sistem tanpa melihat struktur internal kode program. Pengujian dilakukan dengan memberikan input tertentu pada sistem, lalu mengevaluasi apakah output yang dihasilkan sesuai dengan hasil yang diharapkan.

Black Box Testing dipilih karena sistem ini memiliki berbagai fitur antarmuka yang langsung berinteraksi dengan pengguna, seperti login, pengelolaan data dosen, bidang keahlian, jadwal mengajar, hingga tampilan grafik penilaian. Dengan metode ini, setiap fungsi diuji berdasarkan masukan yang diberikan oleh pengguna dan keluaran yang ditampilkan oleh sistem, tanpa perlu memperhatikan bagaimana sistem memproses di balik layar.

Pengujian dilakukan secara manual dengan menjalankan sistem pada localhost menggunakan perintah *php spark serve* melalui terminal Visual Studio Code, kemudian diakses melalui browser. Setiap fitur diuji satu per satu menggunakan input valid dan tidak valid untuk mengamati bagaimana sistem merespons masukan yang diberikan.

### **Hasil Pengujian**

Pengujian sistem dilakukan pada beberapa fitur utama menggunakan metode Black Box Testing. Setiap fitur diuji berdasarkan input yang diberikan oleh pengguna, keluaran yang diharapkan, dan hasil aktual yang ditampilkan oleh sistem. Pengujian difokuskan pada fitur-fitur yang berkaitan langsung dengan interaksi pengguna, khususnya dosen.

**Tabel 4.1 Uji Fungsi Sistem** berikut menyajikan hasil pengujian fitur-fitur utama sistem manajemen data dosen. Pengujian dilakukan secara manual melalui browser dengan menjalankan sistem pada localhost menggunakan php spark serve. Setiap pengujian mencakup satu skenario input dan dievaluasi berdasarkan kesesuaian hasil output dengan harapan sistem.

**Tabel 4.1 Uji Fungsi Sistem**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Fitur yang Diuji** | **Input** | **Output yang Diharapkan** | **Hasil** | **Keterangan** |
| 1 | Login Dosen | Email dan password valid | Masuk ke dashboard dosen | Berhasil | Sesuai |
| 2 | Edit Data Profil Dosen | Perubahan data nama, alamat, dll | Data diperbarui dan ditampilkan di dashboard | Berhasil | Sesuai |
| 3 | Tambah Bidang Keahlian | Nama bidang: "Sistem Informasi" | Tampil di tabel keahlian | Berhasil | Sesuai |
| 4 | Tambah Jadwal Mengajar | MK, Hari, Jam, Ruangan | Tampil di tabel jadwal | Berhasil | Sesuai |
| 5 | Tampilkan Grafik Evaluasi | Data evaluasi tersedia | Grafik muncul di dashboard | Berhasil | Sesuai |

### **Evaluasi Hasil Pengujian**

Berdasarkan hasil pengujian yang telah dilakukan dan disajikan pada **Tabel 4.1** **Uji Fungsi Sistem**, seluruh fitur utama dalam sistem manajemen data dosen dapat berjalan dengan baik dan sesuai dengan yang diharapkan. Setiap masukan (input) yang diberikan oleh pengguna berhasil diproses oleh sistem dan menghasilkan keluaran (output) yang sesuai dengan skenario yang telah dirancang.

Fitur-fitur seperti login dosen, pengelolaan data profil, penambahan bidang keahlian, jadwal mengajar, hingga visualisasi grafik evaluasi kinerja telah diuji menggunakan metode Black Box Testing. Hasilnya menunjukkan bahwa sistem mampu menangani data masukan dengan benar, menampilkan data secara akurat, serta menampilkan elemen visual (tabel dan grafik) secara fungsional dan responsif.

Dengan tidak ditemukannya kesalahan fungsional selama pengujian, maka dapat disimpulkan bahwa sistem ini layak digunakan dan telah memenuhi kebutuhan dasar pengguna yang menjadi sasaran dari sistem, khususnya dosen. Evaluasi lanjutan dapat dilakukan pada versi pengembangan berikutnya apabila ditambahkan fitur baru atau peran pengguna lain seperti admin telah diaktifkan sepenuhnya.

## Evaluasi istem

Evaluasi sistem dilakukan untuk menilai sejauh mana sistem manajemen data dosen yang telah dikembangkan memenuhi tujuan awal pengembangannya. Berdasarkan hasil pengujian yang telah dilakukan, sistem telah berhasil menjalankan seluruh fitur utama seperti login multi-user, pengelolaan data dosen, penambahan bidang keahlian dan jadwal mengajar, serta visualisasi grafik evaluasi kinerja secara fungsional dan akurat.

Selain dari sisi teknis, evaluasi sistem juga mempertimbangkan aspek kegunaan (usability), kinerja (performance), dan kemudahan pemeliharaan (maintainability). Dari segi kegunaan, antarmuka pengguna dinilai cukup intuitif dan mudah dipahami oleh dosen sebagai pengguna utama. Fitur-fitur dirancang sederhana dan langsung ke fungsi yang dibutuhkan, tanpa kompleksitas berlebihan. Dari sisi performa, sistem mampu memproses input dengan cepat dan menampilkan output tanpa keterlambatan signifikan selama dijalankan di lingkungan localhost. Sementara itu, struktur sistem berbasis CodeIgniter 4 dan pendekatan MVC memungkinkan sistem untuk lebih mudah dikembangkan dan diperbaiki di masa mendatang.

Namun, evaluasi ini juga mengidentifikasi bahwa sistem masih memiliki keterbatasan, khususnya pada peran admin yang belum diimplementasikan secara penuh. Selain itu, keamanan akses dan otorisasi perlu ditingkatkan untuk penggunaan skala institusi. Oleh karena itu, sistem ini dinilai layak digunakan sebagai prototipe awal, dan dapat dikembangkan lebih lanjut menjadi sistem produksi dengan cakupan peran yang lebih luas dan fitur tambahan seperti manajemen file atau ekspor data.

# BAB V PENUTUP

## Kesimpulan

Sistem manajemen data dosen berbasis web ini dikembangkan menggunakan framework CodeIgniter 4 dengan pendekatan arsitektur Model-View-Controller (MVC). Sistem dirancang untuk membantu dosen dan pihak institusi dalam mengelola informasi akademik secara lebih efisien, terpusat, dan transparan. Fitur-fitur utama yang disediakan meliputi pengelolaan profil dosen, bidang keahlian, jadwal mengajar, portofolio Tri Dharma, serta visualisasi evaluasi kinerja dalam bentuk grafik interaktif.

Permasalahan utama yang dihadapi sebelumnya adalah proses pencatatan data dosen yang masih dilakukan secara manual, tidak terintegrasi, dan rawan kesalahan. Hal ini seringkali menghambat kelancaran proses administrasi akademik dan pelaporan institusi. Dengan hadirnya sistem ini, seluruh data dosen dapat dikelola secara digital, tersimpan dalam basis data yang terstruktur, dan dapat diperbarui secara langsung oleh dosen melalui akun masing-masing. Fitur validasi, autentikasi, dan kontrol akses juga diterapkan untuk menjaga keamanan data.

Hasil implementasi sistem menunjukkan bahwa semua fitur utama berjalan dengan baik sesuai kebutuhan pengguna. Proses login, edit data, penambahan keahlian dan jadwal, serta tampilan grafik evaluasi telah diuji menggunakan metode Black Box Testing dan dinyatakan berhasil tanpa bug yang berarti. Antarmuka sistem juga dinilai responsif dan mudah dipahami oleh pengguna, baik saat dijalankan di desktop maupun perangkat mobile.

Dengan demikian, sistem ini dinilai layak digunakan sebagai prototipe awal dalam digitalisasi pengelolaan data dosen. Sistem tidak hanya menyederhanakan proses administratif, tetapi juga meningkatkan akurasi dan transparansi data akademik. Ke depan, sistem ini masih dapat dikembangkan lebih lanjut dengan penambahan fitur-fitur tambahan serta perluasan peran pengguna untuk mendukung skala operasional yang lebih besar.

## Saran Pengembangan

Meskipun sistem manajemen data dosen ini telah berjalan sesuai dengan kebutuhan dasar pengguna, namun masih terdapat ruang untuk pengembangan lebih lanjut, khususnya dari sisi fitur dan keamanan. Salah satu saran utama adalah menambahkan fitur ekspor data ke format PDF atau Excel untuk keperluan pelaporan, arsip, dan administrasi formal. Hal ini akan sangat membantu dosen maupun admin dalam menyimpan dan mencetak informasi penting secara praktis.

Selain itu, pengembangan pada sisi peran pengguna admin juga perlu dilakukan secara lebih komprehensif. Saat ini sistem masih berfokus pada peran dosen, sehingga dibutuhkan modul khusus untuk admin yang memiliki hak akses penuh terhadap semua data, termasuk menyetujui permintaan perubahan data, memantau aktivitas dosen, dan mengelola akun pengguna lainnya. Penambahan fitur otorisasi berbasis role yang lebih rinci juga dapat meningkatkan kontrol dan keamanan sistem.

Pengembangan selanjutnya dapat diarahkan pada integrasi dengan sistem akademik kampus secara menyeluruh, seperti SIAKAD atau sistem penjadwalan kuliah. Selain itu, sistem juga dapat dikembangkan untuk mendukung manajemen file seperti unggah sertifikat, laporan kegiatan, atau dokumen pendukung lainnya. Dengan pengembangan berkelanjutan, sistem ini diharapkan mampu menjadi solusi digital terintegrasi dalam pengelolaan aktivitas akademik dosen secara profesional dan efisien.

# DAFTAR PUSTAKA

**Stair, R., & Reynolds, G. (2016).** Principles of Information Systems (12th ed.). Boston: Cengage Learning.

**Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., & Elmqvist, N.** (2016). *Designing the User Interface: Strategies for Effective Human-Computer Interaction* (6th ed.). Pearson.