

Implementasi *Middleware* Laravel untuk Akses *Multi-User*: Studi Kasus Sistem Berita Acara Perkuliahan

<http://dx.doi.org/10.28932/jutisi.v11i2.10713>

Riwayat Artikel

Received: 26 Desember 2024 | Final Revision: 26 Juli 2025 | Accepted: 26 Juli 2025

Creative Commons License 4.0 (CC BY – NC)



Robby Tan[✉] #1

[#] Program Studi SI Teknik Informatika, Fakultas Teknologi dan Rekayasa Cerdas, Universitas Kristen Maranatha
Jl. Prof. drg. Surya Sumantri No. 65, Bandung, 40165, Indonesia

¹robby.tan@it.maranatha.edu

[✉]Corresponding author: robbytan@it.maranatha.edu

Abstrak — Salah satu peranan dosen dalam Tridarma pendidikan tinggi adalah pengajaran. Dosen memiliki kewajiban untuk melakukan dokumentasi kegiatan pengajaran sebagai pelaporan ke universitas dan beban kerja dosen (BKD). Dokumentasi dilakukan dengan menggunakan berita acara perkuliahan baik yang berbentuk *hard copy* atau formulir *online*. Beberapa masalah yang dihadapi dalam proses pencatatan khususnya secara digital yaitu duplikasi pencatatan digital karena tidak adanya histori data, proses pengisian data yang panjang, serta sulitnya melakukan rekapitulasi data kegiatan. Untuk mengatasi masalah tersebut, diperlukan sebuah sistem yang dapat menangani proses *input* data dan melakukan pembagian sehingga tata usaha, dosen, serta pejabat struktural dapat memantau kegiatan yang dilakukan. Sistem tersebut dibuat dengan menggunakan *Laravel framework* yang mengutilisasi *object-relational mapping* (ORM) dan *middleware*. Implementasi tersebut digunakan untuk membagi hak akses pengguna yaitu tata usaha, dosen, serta pejabat struktural. Masing-masing *role* memiliki peran tersendiri yang tidak dapat dipenuhi oleh pengguna lain. Sistem ini juga telah diuji melalui *user acceptance test* kepada tata usaha, dosen, serta pejabat struktural. Hasil pengujian untuk responsivitas sistem berjalan dengan baik dibuktikan dengan 95.83% responden menyatakan bahwa sistem responsif untuk versi *desktop* dan *mobile*. Mayoritas responden survei juga memberikan nilai yang baik untuk informasi yang disampaikan serta fungsionalitas fitur.

Kata kunci— berita acara perkuliahan; *middleware*; multi-user; Laravel.

Laravel Middleware Implementation for Multi-User Access: Case Study Lecturer Minutes System

Abstract — One of the lecturers' roles in Tridharma is teaching. There is a shift in the role of lecturers where lecturers are not only compiling and delivering materials but also guiding students in class so that they can think actively and independently. Universities must record teaching activities carried out by lecturers. Teaching activities are documented using lecture minutes in either hard copy or online form. There are problems faced in the digital recording process, namely data duplication due to the absence of data history, long data entry, and the difficulty of recapitulating required data. To solve the problem, a system is needed to handle the data input process and share access so administrative staff, lecturers, and structural officials can monitor the activities. Each role has its specific responsibilities that cannot be fulfilled by others. The system has also been tested through a user acceptance test involving administrative staff, lecturers, and structural officials. The testing results indicate that the system's responsiveness performs well, with

95,83% of respondents confirming that the system is responsive on desktop and mobile versions. The majority of respondents also provided positive feedback regarding the clarity of the presented information and the functionality of the features.

Keywords— Laravel; lecturer minutes; middleware; multi-user.

I. PENDAHULUAN

Pendidikan tinggi di Indonesia memegang peranan penting dalam pengembangan sumber daya manusia. Sebagian besar ilmu yang didapatkan di tingkat pendidikan tinggi didapatkan melalui proses pengajaran. Pengajaran adalah salah satu Tridarma dosen yang dilakukan setiap semester. Berdasarkan Permenristekdikti No. 53 pasal 16, pemenuhan kegiatan pengajaran dapat dilakukan dalam bentuk kuliah, responsi, tutorial, praktikum, praktik studio, penelitian, perancangan, pengembangan, tugas akhir, pelatihan bela negara, pertukaran pelajar, magang, wirausaha, pengabdian kepada masyarakat, dan/ atau bentuk pembelajaran lain [1]. Dosen memiliki peran penting dalam memenuhi pemenuhan kegiatan pengajaran, di mana dosen tidak hanya menyusun materi perkuliahan, namun membimbing mahasiswa agar dapat berpikir secara aktif dan mandiri.

Universitas selaku penyelenggara pendidikan tinggi, wajib melakukan dokumentasi kegiatan pengajaran yang dilaporkan sebagai beban kerja dosen (BKD). Pencatatan kegiatan pengajaran dilakukan dengan menggunakan lembar fisik atau formulir *online* [2]. Data pengajaran tersebut akan digabungkan dengan data dosen sebagai pembimbing, penguji, atau dosen wali sehingga didapatkan beban kerja di bidang pengajaran setiap semesternya [3], [4]. Salah satu kelebihan jika menggunakan data digital adalah dimungkinkannya pengembangan seperti pencatatan presensi siswa di kelas dengan *smart card*, sidik jari [5], atau menggunakan wajah [6]. Terdapat beberapa penelitian yang pernah dilakukan dalam bidang akademik di perguruan tinggi untuk mendukung kelancaran organisasi dan administrasi. Pusung dkk. membuat sebuah sistem informasi beban kerja (BKD) dan laporan kinerja dosen (LKD). Sistem ini juga dibangun dengan Laravel *framework* menggunakan metode *prototype*. Hasil penelitian ini berfungsi untuk mengoptimalkan prosedur pengisian dan pemeriksaan BKD dan LKD [7]. Iskandar dkk. membuat rancangan sistem konversi mata kuliah dengan Laravel. Sistem ini dibangun dengan tujuan melakukan pemetaan mata kuliah dari kurikulum lama ke kurikulum baru [8]. Andreansyah dkk. juga melakukan penelitian dengan menggunakan Laravel *framework*. Penelitian tersebut membuat sebuah sistem untuk pengelolaan surat akademik mahasiswa. Sistem ini telah berhasil mengotomatisasi pembuatan surat dan pengelolaan data akademik yang lebih terstruktur.

Untuk mendukung proses dokumentasi kegiatan tersebut, diperlukan sebuah sistem yang memungkinkan dosen untuk mencatat dan melaporkan kegiatan pengajaran. Sebelum terjadinya pandemi Covid-19, proses pencatatan dilakukan secara manual menggunakan formulir berita acara perkuliahan di mana dosen mencatatkan tanggal, materi, dan jumlah siswa. Proses pencatatan asisten yang membantu proses kegiatan perkuliahan dicatat pada formulir terpisah. Kedua formulir ini dikumpulkan setiap minggunya untuk dilakukan rekapitulasi dan pencatatan oleh wakil dekan bagian sumber daya atau tata usaha yang ditugaskan. Memasuki masa pandemi Covid-19, proses pencatatan dilakukan dengan menggunakan formulir *online*. Dengan menggunakan formulir *online*, proses pencatatan semakin detail. Proses yang semula dosen hanya mengisi 1 lembar berita acara perkuliahan yang disediakan khusus untuk setiap jadwal, kemudian berubah di mana dosen perlu memilih dahulu program studi kemudian memilih jadwal mata kuliah sesuai program studi, tanggal kegiatan, jumlah mahasiswa, serta dokumentasi kegiatan. Terdapat beberapa kesulitan yang dihadapi dosen serta wakil dekan sumber daya dalam pencatatan kegiatan ini antara lain:

1. Duplikasi pencatatan kegiatan karena tidak adanya histori data yang dimasukkan.
2. Proses pemilihan jadwal yang cukup panjang karena dosen perlu memilih program studi serta memilih mata kuliah yang diajar. Proses pemilihan jadwal mata kuliah yang diajar rentan menemui kesalahan.
3. Proses rekapitulasi data masih dilakukan terpusat oleh 1 orang sehingga perlu memilah dahulu data per masing-masing program studi.
4. Proses rekapitulasi asisten yang dilakukan secara terpisah dari berita acara dosen sehingga perlu perhitungan ulang.

Berdasarkan masalah-masalah yang dihadapi tersebut, diperlukan sebuah sistem yang dapat menangani duplikasi *input* data serta dapat membagi akses agar dosen, wakil dekan, atau ketua program studi dapat melakukan pemantauan data kegiatan perkuliahan yang telah dimasukkan. Pengaturan hak akses dalam sistem dapat disimpan dalam sebuah tabel *role* yang disimpan dalam basis data. Tabel *role* ini akan dihubungkan dengan tabel pengguna sehingga setiap pengguna yang terdaftar akan memiliki *role* tersendiri.

Laravel memiliki sebuah *layer* untuk melakukan pemisahan hak akses berupa *layer middleware*. Laravel *middleware* merupakan sebuah *class* yang berfungsi memproses HTTP *Request*. Berdasarkan dokumentasi Laravel, *middleware* dapat digunakan untuk autentikasi, *logging* dan pengaturan akses web [9]. Gambar 1 adalah kode program *middleware* untuk melakukan proses validasi apakah token yang disertakan dalam HTTP *Request* memiliki *string secret-key*. Jika token yang disertakan dalam HTTP *Request* tidak memiliki kunci tersebut, maka seluruh *request* akan dialihkan ke halaman *home*.

```

1. class RoleValidation
2. {
3.     public function handle($request, Closure $next)
4.     {
5.         if ($request->input('token') !== 'secret-key') {
6.             return redirect('home');
7.         }
8.
9.         return $next($request);
10.    }
11. }

```

Gambar 1 Kode Program *Middleware* dalam Laravel

Terdapat beberapa penelitian yang menekankan pentingnya pemisahan akses pengguna. Penelitian pertama adalah penelitian yang dibuat oleh Aji dkk. mengenai implementasi Laravel dalam sistem informasi desa. Penelitian ini membangun sebuah sistem informasi desa yang di dalamnya terdapat dua aktor yaitu tata pamong desa dan penduduk. Sistem ini dapat mengatur akses pengguna membuat dua buah alamat URL yang di mana satu URL diakses oleh penduduk, sedangkan URL lain diakses oleh tata pamong desa [10]. Penelitian lain yang dilakukan oleh Awaluddin dkk. membagi akses untuk sistem pengelolaan aset laboratorium komputer. Sistem pengelolaan aset yang dibangun memiliki 2 akses pengguna yaitu admin (kepala laboratorium) dan asisten laboratorium. Kepala laboratorium dapat mengakses seluruh menu, sedangkan asisten laboratorium tidak dapat mengakses menu *role* dan *user* [11]. Penelitian lain dilakukan oleh Purnama Sari dkk. Web yang dirancang dalam penelitian ini membagi pengguna dalam dua *role* yaitu admin dan *member* [12].

Selain menjawab masalah-masalah yang ditemukan, salah satu fokus dari penelitian ini adalah merancang sebuah sistem yang dapat menangani *multi-user* dengan akses dan data yang beragam. Hal ini cukup berbeda dengan penelitian-penelitian di yang telah disebutkan di mana sistem atau aplikasi yang dibuat hanya memiliki 2 hak akses.

II. METODOLOGI PENELITIAN

Metodologi yang digunakan untuk perancangan dan pengembangan perangkat lunak (SDLC) menggunakan *waterfall* dengan susunan sebagai berikut:

1. Analisis Kebutuhan

Tahapan pertama dalam penelitian adalah wawancara terhadap pengguna yaitu tata usaha, dosen, serta pejabat struktural (wakil dekan sumber daya). Wawancara dilakukan untuk menemukan kendala yang terjadi selama penggunaan formulir *online*. Hasil wawancara untuk masing-masing pengguna dituangkan dalam analisis yang menjadi dasar untuk pengembangan sistem.

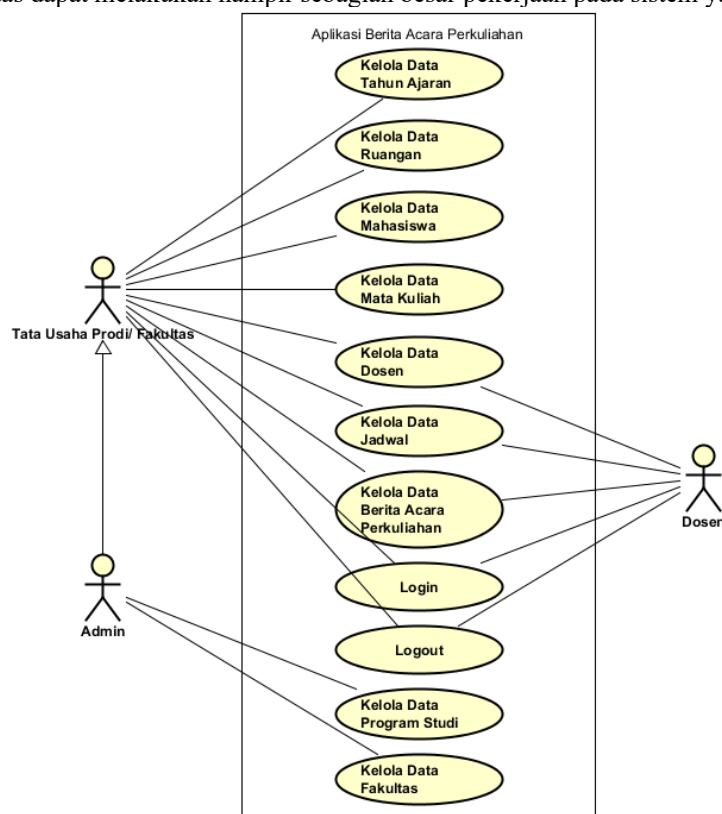
TABEL 1
ANALISIS HASIL WAWANCARA

Pengguna	Hasil wawancara	Analisis
Dosen	<ul style="list-style-type: none"> Kesulitan pada saat proses <i>input</i> data. Untuk sampai dapat melakukan <i>input</i> data, dosen perlu melakukan <i>input</i> email dan pemilihan data program studi serta mata kuliah terlebih dahulu. Beberapa <i>field</i> isian data tidak dapat diisi secara otomatis. Dosen perlu memeriksa email untuk mengetahui apakah sudah memasukkan data untuk pertemuan perkuliahan. Tidak dapat melihat data pertemuan kuliah secara langsung. Bisa terjadi duplikasi <i>input</i> kegiatan perkuliahan. 	<ul style="list-style-type: none"> Data pengajaran yang ditampilkan spesifik untuk setiap dosen. Dosen tidak perlu memilih program studi atau memasukkan email untuk setiap data kegiatan pengajaran. Isian-isian <i>default</i> seperti mata kuliah, tanggal <i>input</i> data akan mengikuti data mata kuliah yang dipilih. Dosen akan memiliki halaman untuk memantau data kegiatan pengajaran yang telah dimasukkan. Setiap <i>input</i> kegiatan perkuliahan akan divalidasi sehingga tidak terjadi duplikasi kegiatan.
Pejabat struktural dan tata usaha	<ul style="list-style-type: none"> Fakultas atau program studi sulit melakukan pemantauan siapa saja dosen yang belum memasukkan data kegiatan perkuliahan. Tahapan proses rekapitulasi BAP yang cukup panjang dimulai dari pengambilan data dari formulir <i>online</i> yang kemudian dipisahkan dari tanggal pengambilan terakhir 	<ul style="list-style-type: none"> Perlu dibuat sebuah halaman yang menampilkan data per fakultas atau program studi yang menunjukkan data BAP yang telah dimasukkan dan dikonfirmasi. Perlu penyederhanaan proses sehingga pejabat struktural atau tata usaha tingkat

Pengguna	Hasil wawancara	Analisis
	<p>yang kemudian dibuat dalam <i>file</i> terpisah. Proses yang panjang ini akan meningkatkan adanya <i>human error</i> misalkan kesalahan <i>cut off</i> tanggal laporan, adanya data yang terlewat, dll.</p> <ul style="list-style-type: none"> • Perlunya proses pemeriksaan data antara data yang telah dimasukkan dengan data yang baru. • Rekapitulasi asisten perlu pemrosesan lebih lanjut karena jumlah asisten per mata kuliah bisa lebih dari 1, sedangkan rekapitulasi berupa hasil akumulasi untuk setiap nama asisten 	<p>fakultas/ program studi dapat melihat dan mengambil data dalam tahapan yang lebih singkat.</p> <ul style="list-style-type: none"> • Pelaporan untuk dosen dan asisten dapat difilter berdasarkan tanggal. • Data rekapitulasi sudah otomatis terhitung melalui menu pelaporan.

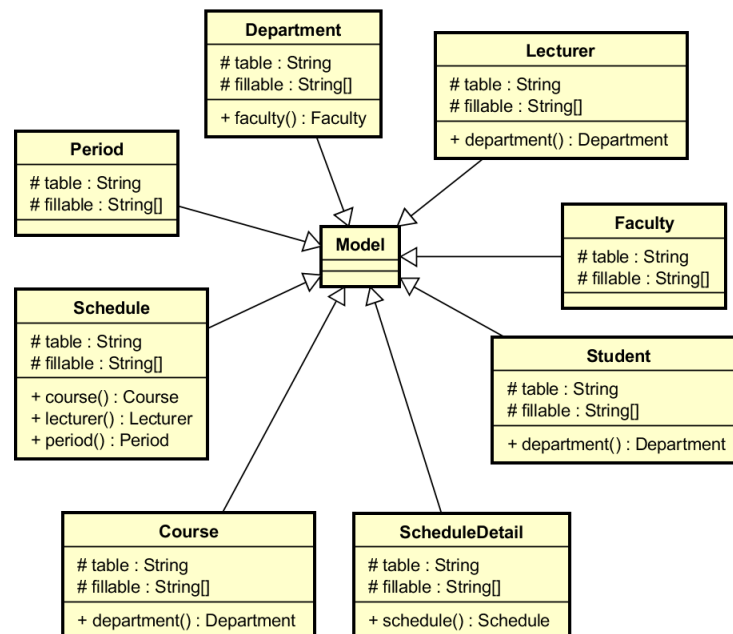
2. Desain Sistem

Rancangan solusi dituangkan dengan menggunakan *Unified Modeling Diagram* (UML). Sistem yang dirancang akan memiliki 3 aktor yaitu dosen, admin, serta pejabat struktural atau tata usaha. Pejabat struktural atau tata usaha di tingkat program studi hanya dapat melihat data pada program studinya masing-masing. Hal yang sama juga berlaku untuk pejabat struktural atau tata usaha di tingkat fakultas. Berdasarkan *use case diagram* pada Gambar 2, aktor tata usaha program studi/ fakultas dapat melakukan hampir sebagian besar pekerjaan pada sistem yang dibuat.



Gambar 2 Use Case Diagram Sistem Berita Acara Perkuliahan

Tahapan pengembangan perangkat lunak berikutnya adalah penyusunan *activity diagram*, *class diagram*, serta *entity relationship diagram*. *Activity diagram* diperlukan untuk menggambarkan proses yang terjadi untuk masing-masing *use case*. *Class diagram* diperlukan untuk mendeskripsikan atribut-atribut serta *method* yang akan digunakan dalam implementasi. Perancangan *class diagram* akan menyesuaikan dengan *framework* yang digunakan yaitu Laravel. Laravel menggunakan konsep *Model-View-Controller* (MVC) beserta dengan Eloquent *Object-Relational Mapping* (ORM) di mana kedua konsep ini membantu struktur perancangan dan implementasi perangkat lunak dengan lebih cepat [13][14][15]. Gambar 3 adalah susunan *class* yang dibuat dalam *class diagram*. *Class Model* adalah *super class* yang merupakan induk dari seluruh *class* yang dibuat dengan menggunakan ORM.



Gambar 3 Rancangan Class Diagram untuk Model

3. Implementasi

Tahap ini dilakukan dengan menggunakan Laravel sebagai *framework* untuk perancangan aplikasi. Laravel dipilih karena dapat membuat proses perancangan perangkat lunak berjalan dengan lebih baik. Tahapan ini menyesuaikan dengan desain sistem yang menggunakan MVC. Implementasi Laravel *middleware* dilakukan dengan menggunakan perintah artisan yang merupakan salah satu perintah dalam Laravel *Command Line Interface* (CLI). Gambar 4 merupakan contoh implementasi *middleware* pada *class* yang menjadi *controller*. Kode program tersebut menjelaskan bahwa yang dapat melakukan akses untuk seluruh *method* yang terdapat pada *class* ini adalah pengguna dengan *role* adm, tuf, dan tup.

```

1. class MahasiswaController extends Controller
2. {
3.
4.     public function __construct()
5.     {
6.         $this->middleware('role:adm,tuf,tup');
7.     }
8.
9.     // Method lain yang terdapat pada controller
10. }

```

Gambar 4 Implementasi Middleware pada Controller

4. Pengujian dan Deployment

Tahap pengujian dilakukan dengan menggunakan pengujian *black box*. Pengujian ini dilakukan untuk menguji fungsionalitas-fungsionalitas dari aplikasi berdasarkan rancangan yang telah disusun. Setelah tahap pengujian dilalui, maka dilakukan proses *deployment*. Sistem yang telah dibuat diunggah ke sebuah domain sehingga dapat digunakan oleh dosen, tata usaha tingkat prodi/ fakultas, serta wakil dekan. Setelah penggunaan sistem selama kurang lebih 1 semester, penelitian dilanjutkan dengan melakukan *user acceptant test* untuk menguji apakah sistem yang dibuat telah memenuhi harapan dan kebutuhan pengguna.

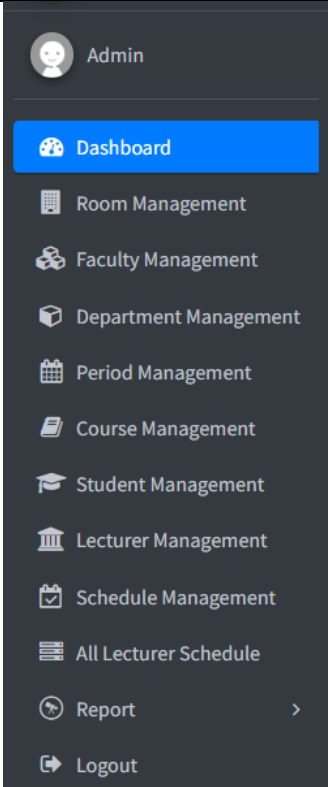
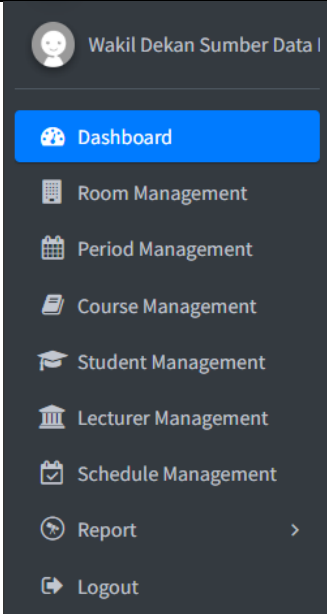
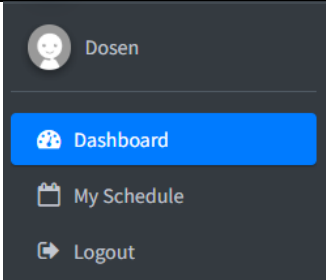
5. Maintenance

Pada tahap terakhir ini, sistem diperbaiki secara berkala sesuai dengan saran-saran perbaikan yang didapatkan pada *user acceptant test* dan selama sistem digunakan. Selain perbaikan, akan terdapat waktu tertentu untuk proses pemeliharaan dari sistem sehingga dapat digunakan dengan nyaman.

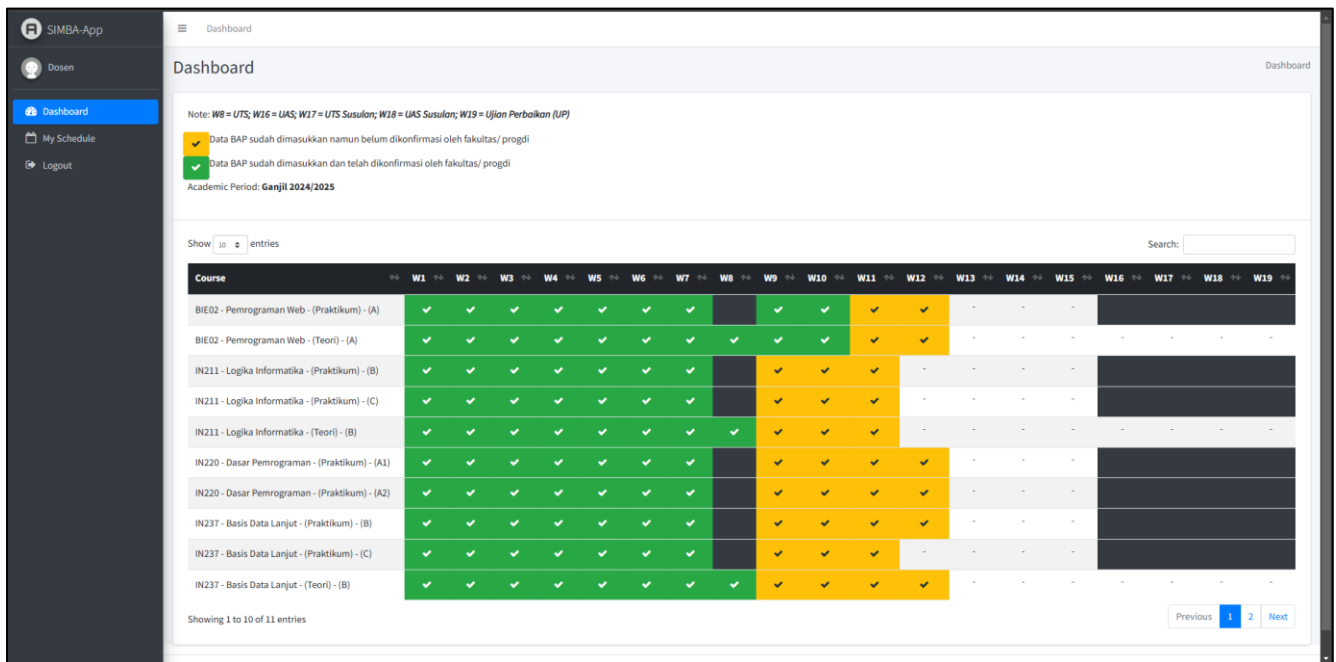
III. HASIL DAN PEMBAHASAN

Tabel 2 menunjukkan perbedaan menu yang terdapat pada masing-masing pengguna. Sesuai dengan rancangan *use case* pada Gambar 2, pengguna dengan *role* admin memiliki hak akses terbanyak yaitu pengelolaan ruangan, fakultas, departemen, periode, mata kuliah, mahasiswa, dosen, jadwal, serta *report* BAP. Pengguna dengan *role* pejabat struktural atau tata usaha fakultas memiliki hak akses untuk pengelolaan ruangan, periode, mata kuliah, mahasiswa, dosen, jadwal, serta *report* BAP. Pengguna dengan *role* dosen hanya dapat melihat *dashboard* untuk mata kuliah yang diajar serta dapat melihat BAP untuk setiap mata kuliah yang diajar. BAP yang telah dimasukkan oleh dosen masih dapat diperbaiki atau dihapus asalkan data tersebut belum dikonfirmasi oleh wakil dekan atau tata usaha. Pembagian ini sesuai dengan *use case diagram* pada Gambar 2.

TABEL 2
PERBANDINGAN MENU HASIL IMPLEMENTASI MIDDLEWARE

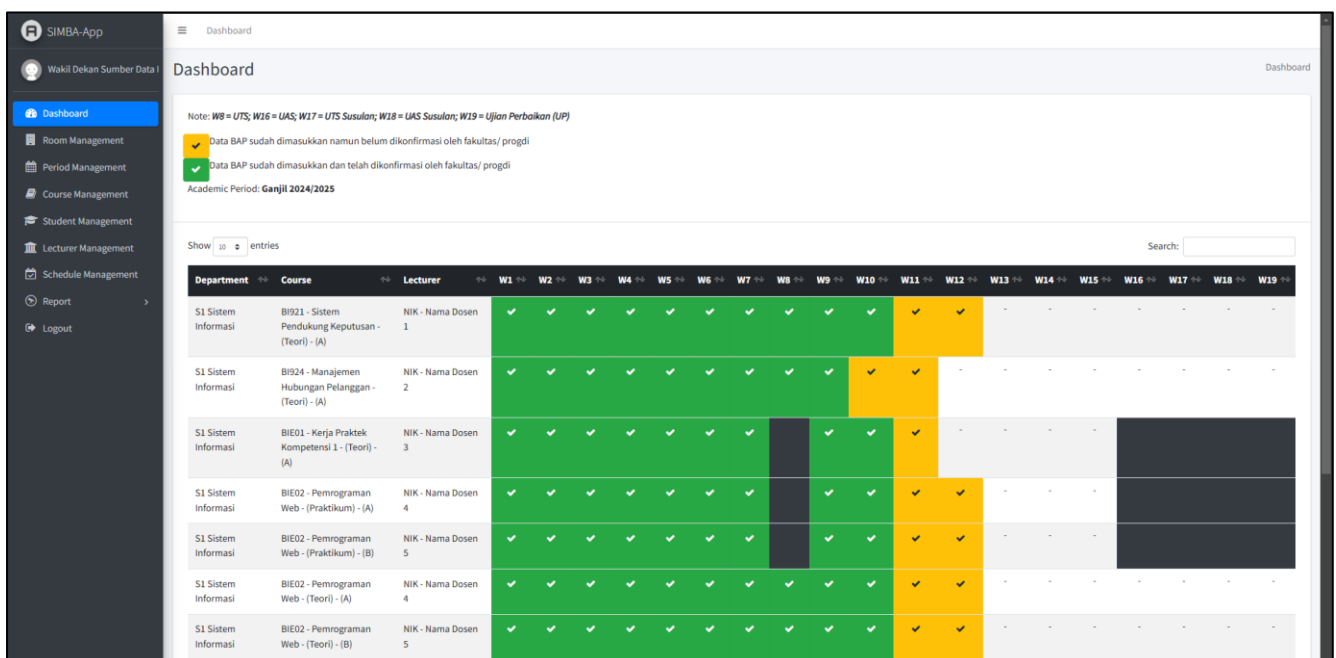
Menu Admin	Menu Pejabat Struktural Atau Tata Usaha Fakultas	Menu Dosen
		

Gambar 5 merupakan solusi yang dibuat untuk mengatasi permasalahan di mana dosen tidak dapat memantau data BAP yang telah dimasukkan. Solusi tersebut berupa sebuah *dashboard* yang menampilkan data informasi mata kuliah yang diajar oleh dosen yang bersangkutan. Data yang ditampilkan pada halaman ini adalah dilengkapi dengan data BAP yang telah dimasukkan untuk setiap pertemuan. Terdapat diferensiasi data antara data yang dimasukkan oleh dosen. Data BAP berwarna kuning merupakan data BAP yang telah dimasukkan namun belum dikonfirmasi oleh pejabat struktural atau tata usaha fakultas/ program studi. Data berwarna hijau adalah data BAP yang telah dikonfirmasi oleh pejabat struktural atau tata usaha fakultas/ program studi. Dosen dapat langsung memunculkan formulir *input* BAP untuk mata kuliah yang diajar dengan cara melakukan klik minggu perkuliahan yang masih kosong (minggu yang masih bertanda -).



Gambar 5 Dashboard Pengguna dengan Role Dosen

Gambar 6 adalah *dashboard* bagi pengguna dengan *role* pejabat struktural atau tata usaha tingkat fakultas. Data yang ditampilkan adalah data kegiatan perkuliahan di seluruh program studi pada fakultas yang sesuai. Pejabat struktural atau tata usaha tingkat fakultas dapat memantau kemajuan proses *input* BAP oleh dosen sehingga dapat memberikan pengingat jika terdapat data yang belum dimasukkan. Pejabat struktural atau tata usaha fakultas/ program studi tidak memiliki wewenang untuk menambahkan data BAP milik dosen. Fungsi pengelolaan data oleh pejabat struktural atau tata usaha tingkat fakultas lebih banyak dibandingkan pengguna dengan *role* dosen. Terdapat fungsi untuk manajemen ruangan, periode, mata kuliah, mahasiswa, dosen, jadwal, serta laporan (*report*) BAP.



Gambar 6 Dashboard Pengguna dengan Role Pejabat Struktural (Wakil Dekan)

Gambar 7 merupakan solusi yang dibuat untuk mengatasi masalah proses *input* data yang cukup panjang. Halaman web yang ditampilkan sudah menggunakan beberapa nilai *default* sebagai nilai awal untuk data BAP. Beberapa data yang sudah otomatis diisi adalah kode dan nama mata kuliah, kelas, jenis kelas, tanggal pertemuan, waktu pertemuan, serta daftar asisten. Tanggal pelaksanaan kelas secara otomatis sudah diisi dengan tanggal dibukanya halaman ini sedangkan jam kelas disesuaikan dengan jam yang terdapat pada jadwal. Asisten untuk mata kuliah sudah dibuat sama seperti pertemuan sebelumnya sehingga dosen hanya perlu memasukkan total jam asistensi untuk setiap asisten.

The screenshot displays the 'BAP Form' interface. The form is pre-filled with the following data:

- Course: BIE02, Pemrograman Web
- Class: A, Type: Praktikum
- Week #: 12
- Teaching Date: 12/12/2024
- Start Time: 03:00 PM
- Students Count: 1
- Lecture Material (Topic): e.g. OOP Concept (Inheritance)
- Class Information: Cara Kuliah: Materi Morning, Latihan Assignment, Onsite/ Online; Mahasiswa Tidak Hadir: NRP - Nama
- File (pdf, jpeg, png): Choose File, No file chosen
- Assistant 1: NRP - Mahasiswa 1, Total Hour: 0
- Assistant 2: NRP - Mahasiswa 2, Total Hour: 0

Red circles highlight the 'Teaching Date', 'Start Time', and the 'Assistant' rows, indicating pre-filled or default values.

Gambar 7 Proses Input BAP dengan Beberapa Field yang Diotomatisasi

Gambar 8 merupakan solusi yang dibuat untuk mengatasi masalah rekapitulasi data serta pemilahan data BAP yang sudah dikonfirmasi dan yang belum dikonfirmasi. Halaman ini hanya dapat diakses oleh pejabat struktural atau tata usaha tingkat fakultas/ program studi. Fungsi utama dari halaman ini adalah melakukan konfirmasi data untuk BAP yang telah dimasukkan oleh dosen. Pejabat struktural atau tata usaha tingkat fakultas/ program studi dapat menentukan tanggal awal dan tanggal akhir BAP yang akan diproses. Setelah menentukan tanggal, seluruh data BAP akan diubah statusnya menjadi terkonfirmasi sehingga tidak dapat diubah atau dihapus oleh dosen. Pada tahapan ini, pejabat struktural atau tata usaha tingkat fakultas/ program studi dapat mencetak *report* (laporan) data BAP yang telah dikonfirmasi sesuai dengan tanggal yang dipilih.

Department	Course	Lecturer	Week	Schedule	Class Date & Time	Student Count	Topic	Input Date	Status	Confirmed Date
S1 Sistem Informasi	BI924 - Manajemen Hubungan Pelanggan - (Teori) - (A)	NIK - Nama Dosen 1	9	Thursday, 12:30:00 - 15:00:00	Thu, 21 Nov 2024, 12:30:00	14	Topik 9 pada Manajemen Hubungan Pelanggan	03 Dec 2024 21:12:07	Confirmed	Fri, 06 Dec 2024
S1 Sistem Informasi	BI924 - Manajemen Hubungan Pelanggan - (Teori) - (A)	NIK - Nama Dosen 1	10	Thursday, 12:30:00 - 15:00:00	Thu, 28 Nov 2024, 12:30:00	18	Topik 10 Manajemen Hubungan Pelanggan	05 Dec 2024 09:12:25	Unchecked	-
S1 Sistem Informasi	BI924 - Manajemen Hubungan Pelanggan - (Teori) - (A)	NIK - Nama Dosen 1	11	Thursday, 12:30:00 - 15:00:00	Thu, 05 Dec 2024, 12:30:00	18	Topik 11 Manajemen Hubungan Pelanggan	11 Dec 2024 10:12:44	Unchecked	-
S1 Sistem Informasi	BID01 - Pengenalan Data Science - (Teori) - (A)	NIK - Nama Dosen 2	9	Monday, 07:00:00 - 09:30:00	Mon, 06 May 2024, 07:00:00	17	Week 08: Categorical Encoding: Label Encoding & One Hot Encoding	06 May 2024 00:05:48	Confirmed	Thu, 30 May 2024
S1 Sistem Informasi	BID01 - Pengenalan Data Science - (Teori) - (B)	NIK - Nama Dosen 2	9	Thursday, 07:00:00 - 09:30:00	Thu, 09 May 2024, 07:00:00	15	Week 08: Categorical Encoding: Label Encoding & One Hot Encoding	10 May 2024 00:05:57	Confirmed	Thu, 30 May 2024
S1 Sistem Informasi	BID01 - Pengenalan Data Science - (Teori) - (A)	NIK - Nama Dosen 2	10	Monday, 07:00:00 - 09:30:00	Mon, 13 May 2024, 07:00:00	17	Week 09: Mengenal Bag of Words, Stop Word Filtering & TF-IDF	13 May 2024 00:05:20	Confirmed	Thu, 30 May 2024
S1 Sistem Informasi	BID01 - Pengenalan Data Science - (Teori) - (B)	NIK - Nama Dosen 2	10	Thursday, 07:00:00 - 09:30:00	Thu, 16 May 2024, 07:00:00	15	Week 09: Mengenal Bag of Words, Stop Word Filtering & TF-IDF	16 May 2024 01:05:33	Confirmed	Thu, 30 May 2024
S1 Sistem Informasi	BID01 - Pengenalan Data Science - (Teori) - (A)	NIK - Nama Dosen 2	11	Monday, 07:00:00 - 09:30:00	Mon, 20 May 2024, 07:00:00	17	Week 10: Logistic Regression pada Binary Classification Task	20 May 2024 01:05:44	Confirmed	Tue, 11 Jun 2024

Gambar 8 Rekapitulasi Data BAP

Gambar 9 merupakan solusi yang dibuat untuk rekapitulasi asisten. Permasalahan yang sebelumnya dihadapi yaitu sulitnya melakukan proses rekapitulasi dibuat dalam bentuk sebuah halaman yang menampilkan data asisten untuk setiap data BAP. Data asisten akan dikonfirmasi secara bersamaan dengan data BAP dosen sehingga pejabat struktural atau tata usaha tingkat fakultas/ program studi tidak perlu mengulangi pekerjaan yang sama.

Course	Class	Assistant	Date	Working Hours	Status	Confirmed Date
BIE02 - Pemrograman Web - Praktikum	A	NRP - Nama Mahasiswa 1	21-Nov-2024	2	Confirmed	Fri, 06 Dec 2024
BIE02 - Pemrograman Web - Praktikum	A	NRP - Nama Mahasiswa 1	28-Nov-2024	2	Confirmed	Fri, 06 Dec 2024
BIE02 - Pemrograman Web - Praktikum	A	NRP - Nama Mahasiswa 2	21-Nov-2024	2	Confirmed	Fri, 06 Dec 2024
BIE02 - Pemrograman Web - Praktikum	A	NRP - Nama Mahasiswa 2	28-Nov-2024	2	Confirmed	Fri, 06 Dec 2024
BIE02 - Pemrograman Web - Praktikum	B	NRP - Nama Mahasiswa 3	31-Oct-2024	2	Confirmed	Fri, 06 Dec 2024
BIE02 - Pemrograman Web - Praktikum	B	NRP - Nama Mahasiswa 3	21-Nov-2024	2	Confirmed	Fri, 06 Dec 2024
BIE02 - Pemrograman Web - Praktikum	B	NRP - Nama Mahasiswa 3	28-Nov-2024	2	Confirmed	Fri, 06 Dec 2024
BIE02 - Pemrograman Web - Praktikum	B	NRP - Nama Mahasiswa 4	31-Oct-2024	2	Confirmed	Fri, 06 Dec 2024
BIE02 - Pemrograman Web - Praktikum	B	NRP - Nama Mahasiswa 4	21-Nov-2024	2	Confirmed	Fri, 06 Dec 2024
BIE02 - Pemrograman Web - Praktikum	B	NRP - Nama Mahasiswa 4	28-Nov-2024	2	Confirmed	Fri, 06 Dec 2024

Gambar 9 Rekapitulasi Data Asisten

Pengujian menggunakan kuesioner didesain dan disebarikan kepada pengguna untuk mengetahui tingkat penerimaan dan keberhasilan dari aplikasi. Kuesioner dibagikan kepada dosen serta pejabat struktural tingkat fakultas/ program studi serta

tata usaha. Kuesioner dibagi menjadi tiga bagian yaitu bagian umum, dosen, serta pejabat struktural atau tata usaha. Daftar pertanyaan yang diajukan dalam kuesioner ditunjukkan pada Tabel 3.

TABEL 3
PERTANYAAN DAN TARGET PENGGUNA

Pertanyaan	Isi Pertanyaan	Target Pengguna
Q1	Tampilan sistem selalu responsif pada perangkat yang digunakan	Dosen, pejabat struktural, tata usaha
Q2	Informasi yang ditampilkan pada sistem sesuai dan mudah dipahami	Dosen, pejabat struktural, tata usaha
Q3	Kemudahan untuk menemukan fitur atau fungsionalitas yang dibutuhkan	Dosen, pejabat struktural, tata usaha
QD1	Kelancaran proses pencatatan berita acara dengan menggunakan sistem	Dosen
QD2	Waktu yang diperlukan untuk menyelesaikan proses <i>input</i> berita acara	Dosen
QD3	Kemudahan memantau berita acara perkuliahan	Dosen
QD4	Setiap data yang dimasukkan telah divalidasi sehingga membantu dalam melihat apakah terdapat duplikasi data atau data yang kurang	Dosen
QM1	Manajemen data pada sistem berlangsung dengan baik dan lancar	Pejabat struktural, tata usaha
QM2	Sistem membantu pejabat struktural dan/ atau tata usaha untuk melakukan pemantauan terhadap <i>input</i> berita acara perkuliahan di tingkat fakultas dan/ atau program studi	Pejabat struktural, tata usaha
QM3	Fitur konfirmasi membantu dalam memilah data berita acara perkuliahan yang baru dan yang telah dimasukkan dalam sistem gaji	Pejabat struktural, tata usaha
QM4	Waktu yang diperlukan untuk menyelesaikan konfirmasi dan rekapitulasi data singkat	Pejabat struktural, tata usaha

Tabel 4 adalah hasil dari kuesioner yang dibagikan. Terdapat 24 responden yang mengisi kuesioner dengan pembagian 20 responden sebagai dosen dan sisanya adalah pejabat struktural atau tata usaha.

TABEL 4
HASIL SURVEI

Pertanyaan	Hasil Survei
Q1	Sebanyak 95,83% (23 responden) setuju dengan pernyataan bahwa sistem responsif pada perangkat yang digunakan.
Q2	Sebanyak 91,67% (22 responden) memberikan nilai 5 untuk kesesuaian dan pemahaman informasi yang ditampilkan pada sistem, sedangkan 8,33% (2 responden) memberikan nilai 4.
Q3	Sebanyak 83,33% (20 responden) memberikan nilai 5 untuk kemudahan dalam menemukan fitur yang akan digunakan, sedangkan 16,67% (4 responden) memberikan nilai 4.
QD1	Sebanyak 90% (18 responden) memberikan nilai 5 (sangat mudah) terhadap kelancaran proses <i>input</i> data, sedangkan 10% (2 responden) memberikan nilai 5 (mudah).
QD2	Mayoritas responden (60% atau 12 responden) menyatakan bahwa waktu pengisian di sistem berlangsung antara 2-5 menit. Sebanyak 30% (6 responden) membutuhkan waktu pengisian kurang dari 2 menit. Sementara 10% (2 responden) membutuhkan waktu pengisian antara 5-10 menit.
QD3	Sebanyak 70% (17 responden) memberikan nilai 5 (sangat mudah) terhadap kemudahan dalam memantau sistem, sedangkan 30% (7 responden) memberikan nilai 4 (mudah).
QD4	Total 100% (20 responden) menyatakan bahwa setiap masukan data divalidasi sehingga tidak terjadi duplikasi data.
QM1	Sebanyak 75% (3 responden) memberikan nilai 5 untuk kelancaran proses manajemen data. Sedangkan 25% (1 responden) memberikan penilaian 4.
QM2	Sebanyak 75% (3 responden) memberikan nilai 5 (sangat mudah) untuk proses pemantauan berita acara oleh pejabat struktural atau tata usaha. Sedangkan 25% (1 responden) memberikan nilai 4 (mudah).
QM3	Sebanyak 75% (3 responden) memberikan nilai 5 (sangat mudah) untuk fitur konfirmasi berita acara yang dapat memisahkan data lama dan data baru, sedangkan 25% (1 responden) memberikan nilai 4 (mudah).
QM4	Sebanyak 75% (3 responden) memberikan nilai 5 (waktu konfirmasi sangat singkat), sedangkan 25% (1 responden) memberikan nilai 4 (waktu konfirmasi singkat).

IV. SIMPULAN

Dalam perkembangan proses digitalisasi administrasi, banyak perubahan yang perlu dilakukan baik dokumen yang dikirimkan maupun perbaikan proses. Hal tersebut perlu dipertimbangkan mengingat struktur organisasi yang dinamis dan akses sistem yang beragam. Melalui penelitian ini, sistem BAP yang dikembangkan dengan Laravel dapat meningkatkan

proses penunjang akademik di perguruan tinggi. *Middleware* pada Laravel terbukti dapat membantu pengaturan hak akses dengan melakukan pembagian berdasarkan *role* untuk masing-masing tingkat seperti fakultas dan program studi. Sistem BAP yang dibuat memudahkan pengguna baik dosen maupun tata usaha tingkat fakultas/ program studi. Dosen dapat memantau proses perkuliahan yang dijalani dengan data BAP yang telah dimasukkan serta menghilangkan beberapa tahapan proses yang berulang seperti *input* detail kegiatan perkuliahan dan asisten. Pejabat struktural atau tata usaha yang menggunakan sistem BAP diberikan akses untuk melakukan pemantauan BAP serta melakukan konfirmasi BAP dan data asisten yang hasilnya dapat diubah ke format lain seperti Excel atau PDF. Pengujian yang dilakukan dengan *user acceptance test* menyatakan bahwa sebanyak 95.83% responden menilai bahwa sistem yang dibangun responsif dan dapat digunakan dengan baik untuk versi *desktop* atau versi *mobile*. Survei umum kepada seluruh pengguna juga menyatakan bahwa 87.50% informasi dan fungsionalitas fitur dapat ditemukan dengan sangat mudah sedangkan 12.5% memberikan penilaian mudah. Survei kepada dosen juga memberikan nilai yang baik dibuktikan dengan mayoritas responden memberikan penilaian dengan nilai 4 atau 5. Survei kepada pejabat struktural dan tata usaha juga membuktikan bahwa sistem telah berjalan dengan baik dan dapat membagi data serta mempersingkat waktu yang diperlukan untuk melakukan konfirmasi berita acara perkuliahan.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih untuk Fakultas Teknologi dan Rekayasa Cerdas (dahulu Fakultas Teknologi Informasi) dan Lembaga Penelitian dan Pengabdian kepada Masyarakat (LPPM) Universitas Kristen Maranatha karena berkat bantuan, dukungannya, penelitian ini dapat terlaksana dengan baik.

DAFTAR PUSTAKA

- [1] Kemdikbudristek Republik Indonesia, “Permenristekdikti No. 53 tahun 2023 mengenai Penjaminan Mutu Pendidikan Tinggi.” Accessed: Dec. 04, 2023. [Online]. Available: <https://silemkerma.kemdikbud.go.id/vokasi>
- [2] Y. Huda and A. Muni, “Implementasi E-Arsip Beban Kerja Dosen Universitas Islam Indragiri,” *Jurnal Perangkat Lunak*, vol. 5, no. 2, pp. 118–131, 2023.
- [3] W. Hamukti, L. Andrawina, and L. W. Suwarsono, “Analisis Beban Kerja Dosen Bidang Pendidikan dan Penunjang Menggunakan Metode Knowledge Conversion 5C-4C,” *JISI: Jurnal Integrasi Sistem Industri*, vol. 4, no. 2, 2017.
- [4] R. Hayati, D. Puspita Sari, and G. Kartika Hanum, “Catatan Kehadiran Dosen Melalui Management System FO AO,” *CICES (Cyberpreneurship Innovative and Creative Exact and Social Science)*, vol. 10, no. 1, pp. 104–114, 2024.
- [5] I. Dharma Wijaya, A. Rahmadia Aqilah, F. Ramadhani Prihandiva, and N. Haidar Ramadhona Arzin, “Perancangan Sistem Absensi Mahasiswa Berbasis Smart Card dan Fingerprint Menggunakan Framework ITIL,” *Jurnal JUPITER*, vol. 16, no. 1, pp. 239–251, 2024.
- [6] S. Budi *et al.*, “IBaTS - Image based attendance system: A low cost solution to record student attendance in a classroom,” in *Proceedings - 2018 IEEE International Symposium on Multimedia, ISM 2018*, Institute of Electrical and Electronics Engineers Inc., 2018, pp. 259–266.
- [7] D. R. Pusung, C. P. Munaiseche, and O. Kembuan, “Sistem Informasi Beban Kerja dan Laporan Kinerja Dosen Berbasis Web,” *JOINTER*, vol. 01, no. 01, pp. 17–24, 2020.
- [8] D. Iskandar, I. Erliyani, and F. Eliyana, “Rancang Bangun Sistem Informasi Konversi Mata Kuliah Menggunakan Framework Laravel,” *Journal Cerita*, vol. 10, no. 1, pp. 39–45, 2024.
- [9] Laravel Holding Inc., “Laravel Middleware Documentation.” Accessed: Oct. 15, 2024. [Online]. Available: <https://laravel.com/docs/10.x/middleware>
- [10] S. Aji, D. Prاتمanto, and A. Ardiansyah, “Implementasi Framework Laravel Dalam Perancangan Sistem Informasi Desa,” 2021. [Online]. Available: <http://ejournal.bsi.ac.id/ejurnal/index.php/ijse>
- [11] M. I. Awaluddin, R. W. Arifin, and D. Setiyadi, “Implementasi Framework Laravel Pada Sistem Informasi Pengelolaan Aset Laboratorium Komputer,” *BINA INSANI ICT JOURNAL*, vol. 7, no. 2, pp. 187–197, 2020.
- [12] D. Purnama Sari, R. Wijanarko, and J. X. Menoreh Tengah, “Implementasi Framework Laravel pada Sistem Informasi Penyewaan Kamera (Studi Kasus Di Rumah Kamera Semarang),” *Jurnal Informatika dan Rekayasa Perangkat Lunak*, vol. 2, no. 1, pp. 32–36, 2019.
- [13] S. Ranti and A. D. Putra, “Penarapan Framework Laravel pada Sistem E-Booking (Studi Kasus: 4People Studio),” *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 9, no. 3, pp. 1643–1656, 2024.
- [14] D. Aipina and H. Witriyono, “Pemanfaatan Framework Laravel dan Framework Bootstrap pada Pembangunan Aplikasi Penjualan Hijab Berbasis Web,” *Jurnal Media Infotama*, vol. 18, no. 1, 2022.
- [15] R. Yuniarti, I. H. Santi, and W. D. Puspitasari, “Perancangan Aplikasi Point of Sale untuk Manajemen Pemesanan Bahan Pangan Berbasis Framework Laravel,” *Jurnal Mahasiswa Teknik Informatika*, vol. 6, no. 1, pp. 67–74, 2022.