

Peremajaan Website Kerja Praktik Digital Menggunakan Framework Laravel dan Vue.js

Christopher Richard Alexander^{#1}, Sulaeman Santoso^{*2}

[#]Program Studi S1 Teknik Informatika, Universitas Kristen Maranatha
Jl. Prof. Drg. Surya Sumantri No. 65 40164, Bandung, Indonesia

¹2172008@maranatha.ac.id

²sulaeman.santoso@maranatha.ac.id

Abstract — Websites are an important element in individual and organizational activities. Without regular maintenance, websites are at risk of performance degradation, security vulnerabilities, and data loss. The Maranatha Christian University KP2's website was designed to assist Informatics Engineering students in managing practical work, as well as supporting lecturers in the guidance process. However, this website needs to be updated over time due to limited features, and has not supported the Independent Campus Learning path. The maintenance process is carried out by adding features for the Merdeka Learning Campus Merdeka path, and improving platform performance. This website was developed using the Laravel framework for the backend and Vue.js for the frontend, which allows the interface to be more responsive, intuitive, and modern. With a more structured and efficient system, this maintenance is intended to simplify the practical work administration process, from submission to completion. In addition, it is expected to improve stability, access speed, and data integration, so that both students and lecturers can be more productive in using the features provided. These changes are expected to improve the overall user experience.

Keywords— Framework, Merdeka Learning Independent Campus, Update, Digital Internship, Website

I. PENDAHULUAN

Website merupakan elemen penting bagi individu dan organisasi dalam memperluas jangkauan [1]. Tanpa peremajaan dan pemeliharaan yang rutin, website dapat mengalami penurunan kinerja, kerentanan keamanan, hingga kehilangan data [2]. Tanda-tanda website yang memerlukan pembaruan mencakup kurangnya responsivitas, antarmuka yang tidak user-friendly, fitur yang tidak lengkap, serta performa Search Engine Optimization (SEO) yang buruk [3].

Website Kerja Praktik 2 (KP2) Universitas Kristen Maranatha dirancang untuk membantu mahasiswa Teknik Informatika dalam mengelola kerja praktik serta mempermudah dosen dalam proses bimbingan. Seiring waktu, website ini memerlukan peremajaan karena membutuhkan perombakan dan penambahan beberapa fitur, seperti pendaftaran jalur MBKM dan unggah dokumen mahasiswa. Kampus Merdeka merupakan kebijakan dari Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi (Kemendikbudristek) yang memberikan hak kepada mahasiswa untuk mengambil mata kuliah di luar program studi mereka selama satu semester. Fitur jalur MBKM perlu ditambahkan karena saat ini website KP2 belum mendukung fitur tersebut.

Website KP2 dikembangkan menggunakan framework Laravel untuk backend dan Vue.js untuk frontend. Dengan peremajaan ini, diharapkan platform dapat menghadirkan pengalaman yang lebih baik, lebih intuitif, dan lebih responsif bagi mahasiswa serta dosen. Perubahan ini bertujuan mempermudah proses administrasi dan bimbingan, termasuk pengajuan, bimbingan, hingga penyelesaian kerja praktik. Selain itu, sistem yang lebih terstruktur dan efisien diharapkan dapat meningkatkan kecepatan akses, memperbaiki stabilitas, serta mengoptimalkan integrasi data, sehingga baik mahasiswa maupun dosen dapat lebih produktif dalam memanfaatkan fitur-fitur yang disediakan oleh platform ini.

II. KAJIAN TEORI

A. Related Work

Beberapa pendekatan telah dilakukan untuk mengatasi permasalahan serupa. Sekolah Tinggi Analisis Bakti Asih Bandung (2020) mengembangkan website pengajuan sidang online guna mempercepat proses pengajuan sidang akibat ketidakhadiran kepala program studi. Website ini mempermudah pengelolaan sidang, verifikasi waktu, dan pemilihan penguji, namun tidak memiliki fitur revisi nilai akhir [4].

Pada tahun 2023, Fakultas Teknologi Industri dan Informatika UHAMKA mengembangkan sistem yang mempermudah pendaftaran sidang dan pencatatan revisi tanpa perlu pihak sekretariat. Sistem ini juga mengurangi kesalahan input dan mempercepat penilaian melalui otomatisasi, tetapi tidak menyediakan template dokumen yang dapat diunduh mahasiswa [5].

Teknik Informatika Universitas Muhammadiyah Yogyakarta menerapkan sistem pendaftaran sidang tugas akhir yang efisien [6]. Namun, sistem ini masih memiliki kekurangan, seperti tampilan yang kurang menarik, tidak adanya penilaian akhir, serta fitur input dan template dokumen.

STMIK Bina Sarana Global (2013) mengembangkan sistem registrasi dan pendaftaran skripsi online yang memungkinkan pemberian nilai oleh dosen pembimbing serta pemantauan bimbingan oleh program studi. Kekurangan sistem ini meliputi absennya fitur penyimpanan dokumen, tampilan kurang menarik, serta tidak adanya revisi penilaian akhir [7].

B. Pengertian Website

Website adalah sekumpulan halaman *web* yang saling terhubung dan diakses melalui internet menggunakan protokol *HTTP* atau *HTTPS*. *Website* dapat berisi teks, gambar, video, serta elemen interaktif yang disusun dalam sebuah struktur untuk menyajikan informasi atau layanan kepada pengguna [8]. Salah satu komponen utama yang membentuk dasar struktur sebuah *website* adalah *HTML*.

C. Pengertian HTML

HyperText Markup Language (*HTML*) memiliki dua fitur utama: hiperteks dan universalitas. Hiperteks memungkinkan pembuatan tautan yang mengarahkan pengguna ke halaman web lain atau sumber daya lain di internet, sehingga informasi dapat diakses dari berbagai arah [9]. Untuk meningkatkan interaktivitas dan responsivitas terhadap tindakan pengguna, *JavaScript* digunakan dalam pengembangan *website*.

D. Pengertian Javascript

JavaScript adalah bahasa skrip utama untuk browser web dan digunakan dalam pengembangan aplikasi yang kompleks [10]. *JavaScript* dapat ditulis langsung dalam tag `<script>` pada file *HTML* atau disimpan dalam file terpisah dengan format `.js` yang dihubungkan ke *HTML*. Untuk membangun *website* yang lebih kompleks dan efisien, penggunaan *framework* menjadi solusi yang mempermudah pengembangan dan pengelolaan kode.

E. Pengertian Framework

Framework adalah kerangka kerja yang digunakan untuk mengembangkan *website* dengan lebih mudah, cepat, dan terstruktur [11]. *Framework* membantu web developer dalam menulis kode yang lebih efisien dan rapi. Saat ini terdapat berbagai pilihan *framework* untuk pengembangan *website*, namun dalam kasus ini, *Laravel* akan digunakan untuk membangun sistem backend.

1) Pengertian Laravel

Laravel adalah *framework* *PHP* modern yang dirancang untuk mempermudah, mempercepat, dan membuat pengembangan *PHP* lebih intuitif. *Laravel* menerapkan pola arsitektur Model-View-Controller (*MVC*) dalam pengembangan aplikasi web [12]. Untuk desain tampilan, *Vue.js* dipilih sebagai *framework* frontend karena kemampuannya dalam membangun antarmuka pengguna yang interaktif dan responsif.

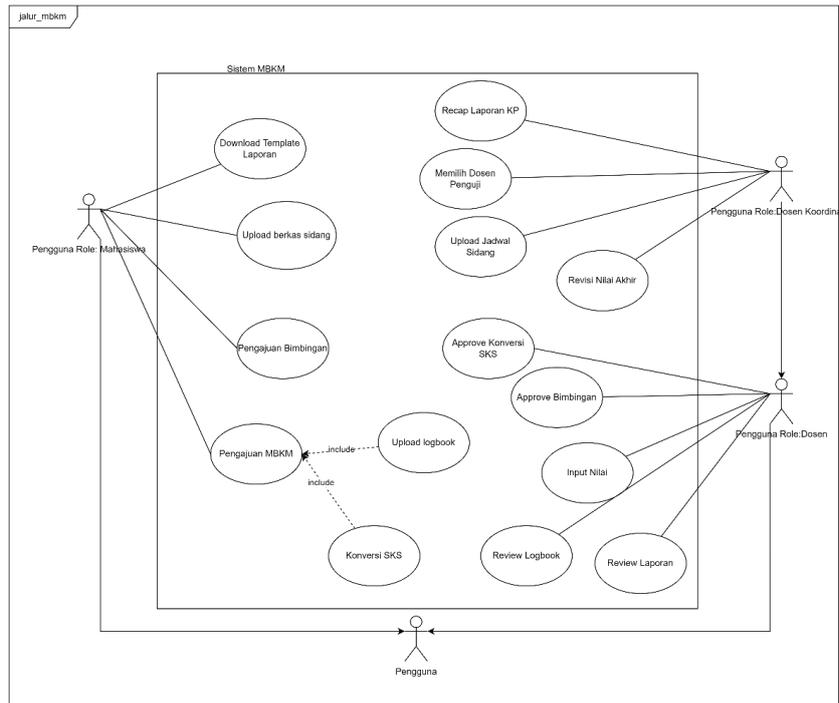
2) Pengertian Vue.js

Vue adalah kerangka kerja untuk membangun antarmuka pengguna secara bertahap. Fokus utamanya adalah pada tampilan, sehingga mudah diintegrasikan dengan pustaka lain atau proyek yang sudah ada [13].

III. ANALISIS DAN RANCANGAN SISTEM

A. Use Case Diagram

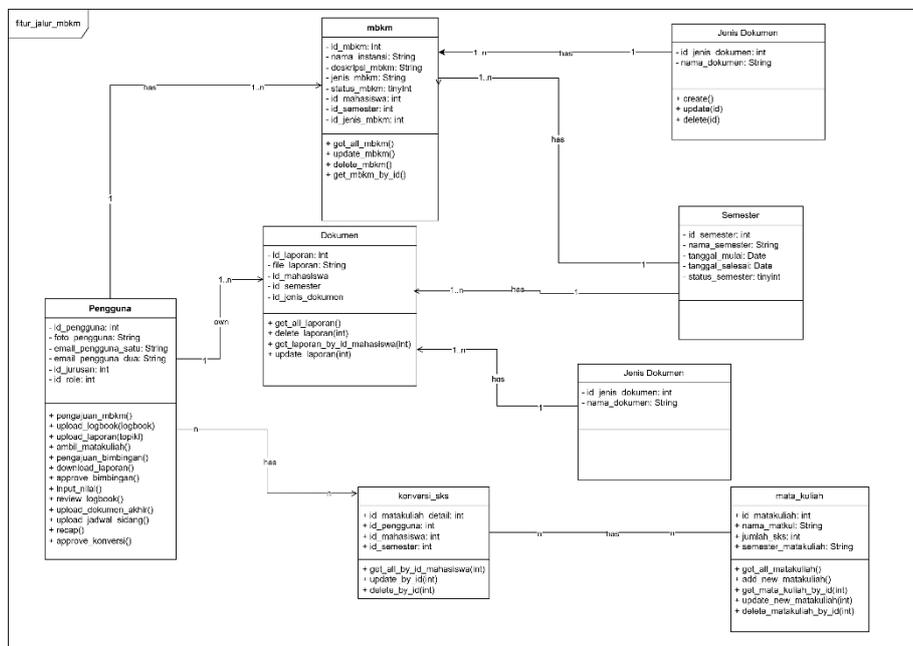
Use Case Diagram pada Gambar 1 menggambarkan interaksi Mahasiswa, Dosen, dan Dosen Koordinator dalam sistem MBKM. Mahasiswa dapat mengajukan MBKM, mengunggah logbook, laporan, serta dokumen akhir, sementara Dosen bertugas memeriksa bimbingan, meninjau logbook, menyetujui konversi SKS, dan memberikan nilai. Dosen Koordinator berperan dalam rekam laporan, memilih dosen penguji, mengunggah jadwal sidang, dan merevisi nilai akhir.



Gambar 1. Use-Case Diagram Jalur MBKM

B. Class Diagram

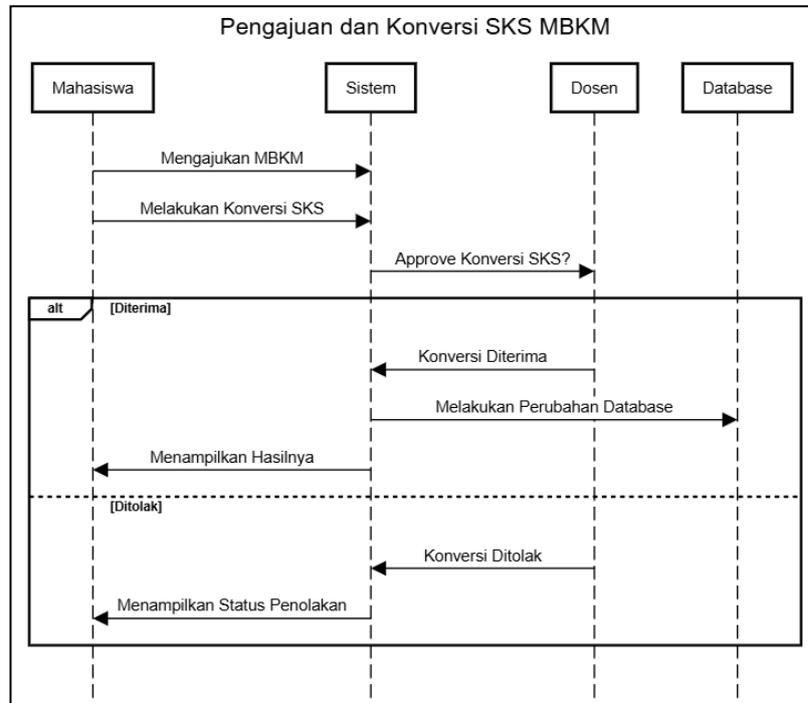
Sistem MBKM terdiri dari beberapa kelas utama yang mengelola berbagai aspek administrasi. Pengguna bertanggung jawab atas informasi dasar pengguna serta memiliki metode untuk pengajuan MBKM, unggah logbook, dan pengelolaan laporan. MBKM menyimpan data mengenai program MBKM yang diajukan mahasiswa, sementara Dokumen mengelola penyimpanan dan pengunggahan dokumen terkait progres akademik. Konversi SKS berfungsi untuk mengatur data konversi mata kuliah berdasarkan program MBKM yang diikuti. Selain itu, Mata Kuliah menyimpan informasi terkait semester dan mata kuliah yang diambil mahasiswa.



Gambar 2. Class Diagram Fitur Jalur MBKM

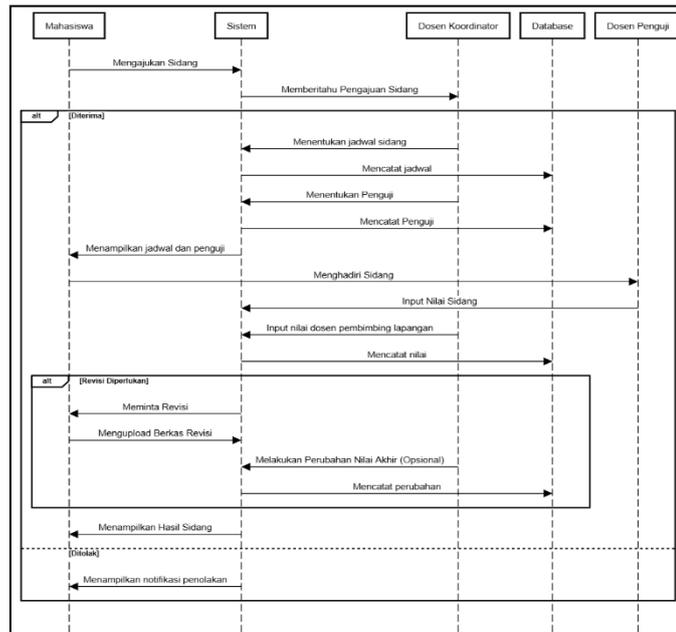
C. Sequence Diagram

Sequence Diagram pada Gambar 3 hingga 5 menggambarkan alur proses dalam sistem MBKM, mulai dari pengajuan hingga penyelesaian. Pada Gambar 3 proses diawali dengan mahasiswa yang mengajukan MBKM, di mana sistem mengonfirmasi pengajuan kepada dosen untuk disetujui atau ditolak, dengan status yang dicatat dalam database.



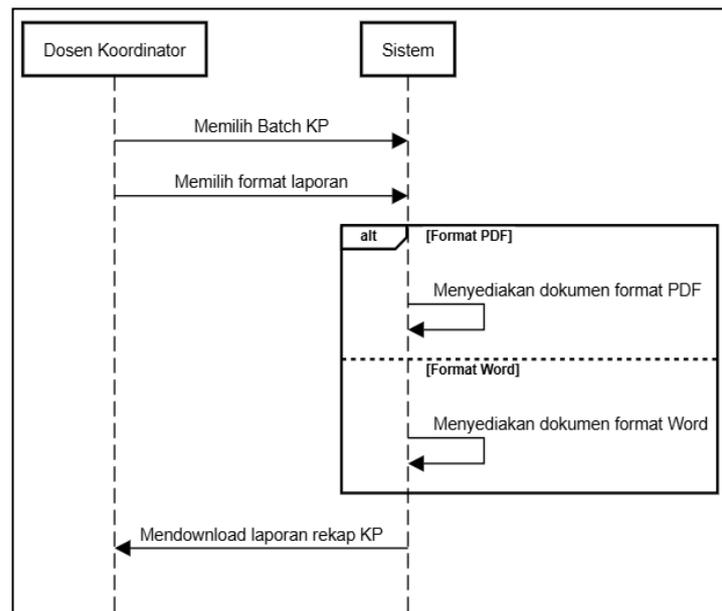
Gambar 3. Sequence Diagram Pengajuan dan Konversi MBKM

Pada Gambar 4 merupakan tahap pengajuan sidang, mahasiswa mengajukan permohonan sidang, kemudian Dosen Koordinator menetapkan jadwal dan penguji, dengan sistem mencatat dan menampilkan hasil sidang serta revisi jika diperlukan.



Gambar 4. Sequence Diagram Pengajuan Sidang MBKM

Selanjutnya, pada Gambar 5 dalam rekap laporan KP, Dosen Koordinator memilih batch dan format laporan, lalu sistem menghasilkan dan menyediakan laporan untuk diunduh sesuai kebutuhan.

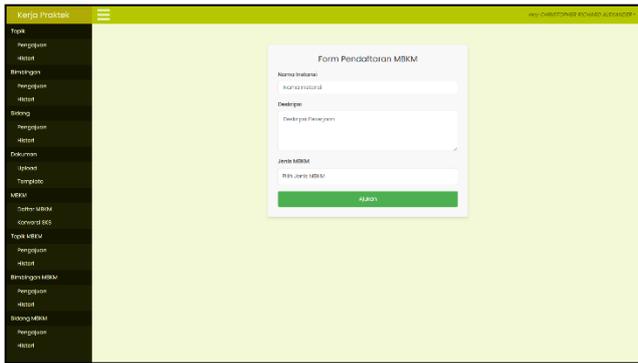


Gambar 5. Sequence Diagram Rekap Laporan KP

IV. IMPLEMENTASI

A. Implementasi Tampilan

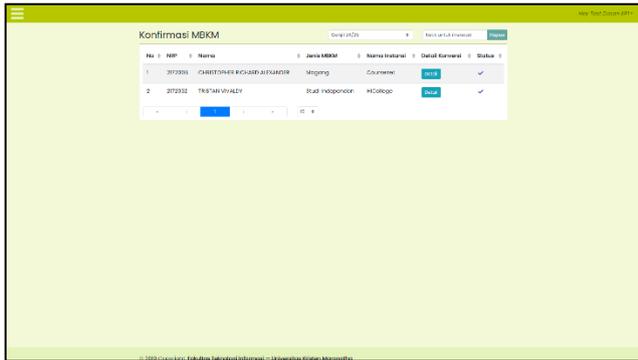
Desain antarmuka dimulai dengan merancang tampilan visual yang mudah dipahami oleh pengguna. Proses implementasi ini mempertahankan tata letak, tipografi, dan kenyamanan bagi penggunanya.



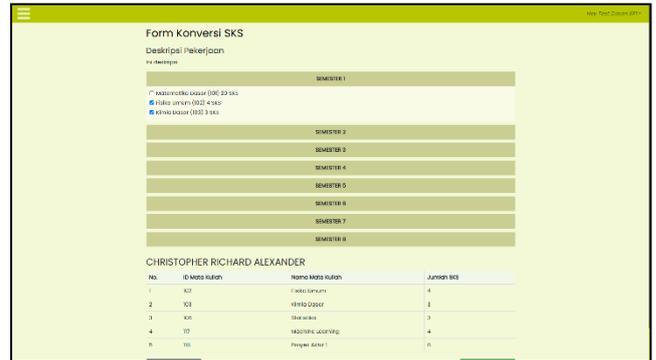
Gambar 6. Tampilan pendaftaran MBKM



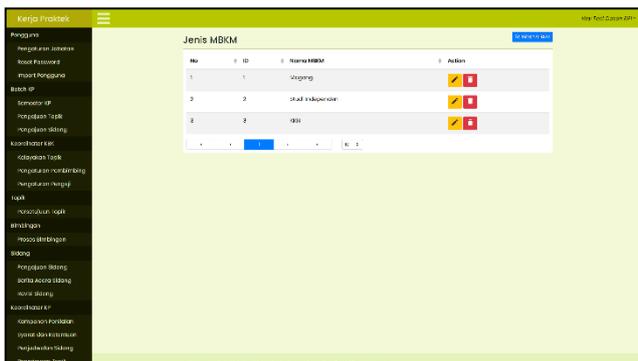
Gambar 7. Tampilan konversi SKS



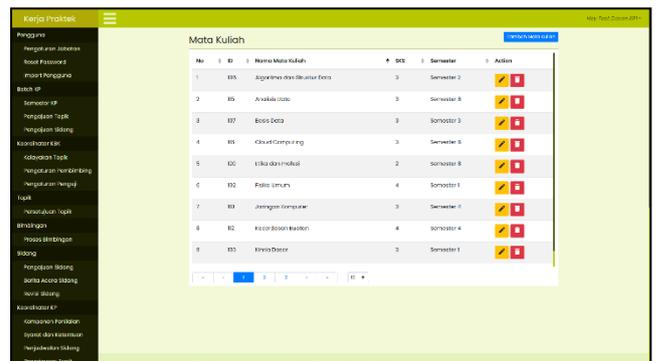
Gambar 8. Tampilan kumpulan mahasiswa yang mendaftar MKBM



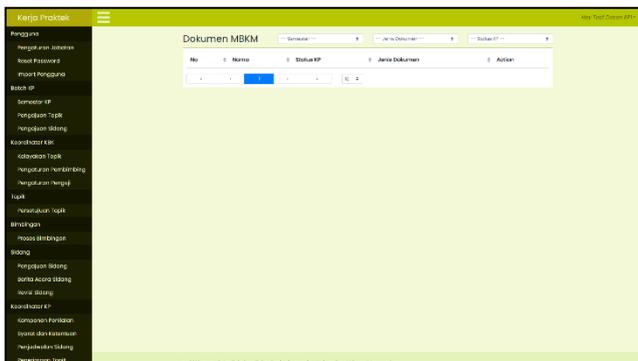
Gambar 9. Tampilan detail konfirmasi



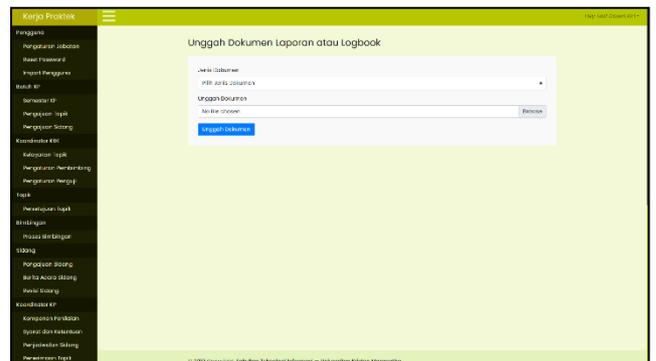
Gambar 10. Tampilan jenis MBKM



Gambar 11. Tampilan mata kuliah



Gambar 12. Kumpulan dokumen MBKM



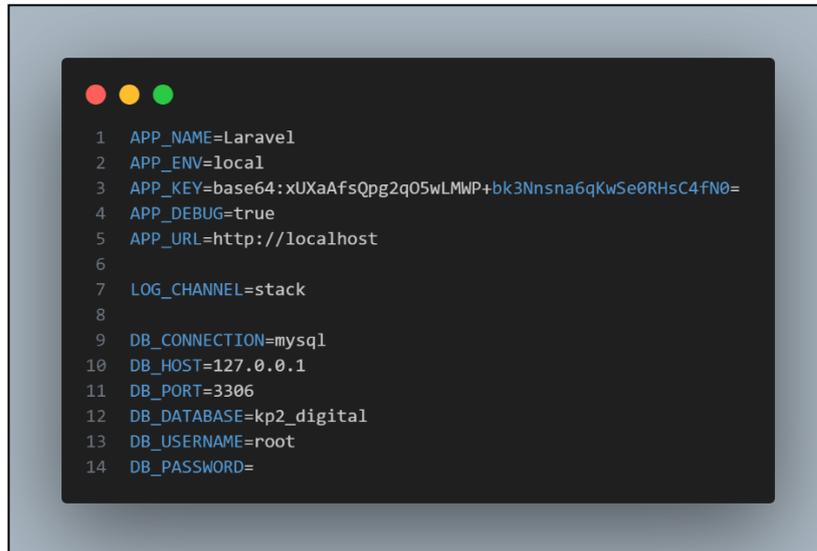
Gambar 13. Unggah template dokumen

B. Implementasi Back-end

Pada bagian ini, akan dibahas mengenai implementasi sisi *back-end* dari sistem yang dikembangkan, yang bertanggung jawab untuk mengelola logika bisnis, pengolahan data, serta komunikasi antara *database* dan antarmuka pengguna.

1) Database dan Pengelolaan Data

Pada Gambar 14 merupakan pengaturan untuk melakukan integrasi *database* terhadap *server* lokal. Nama *database* yang digunakan adalah *kp2_digital*, port 3306, dan menggunakan koneksi MySQL.

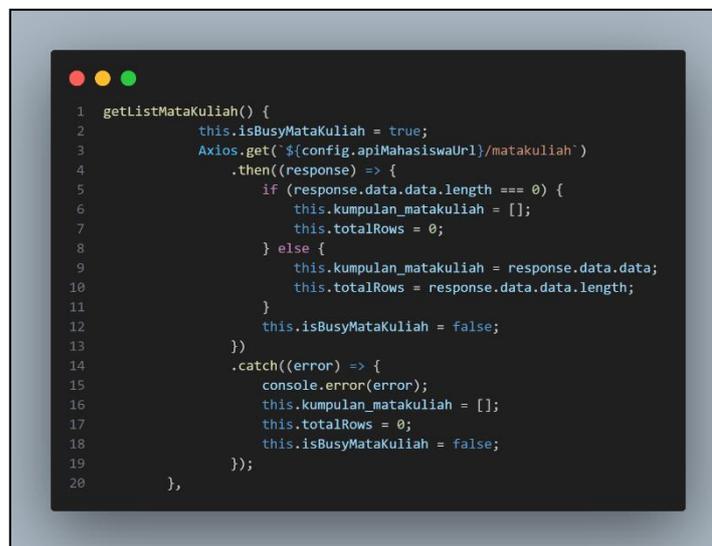


```
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:xUXaAfsQpg2q05wLMWP+bk3Nnsna6qKwSe0RHsC4fN0=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8
9 DB_CONNECTION=mysql
10 DB_HOST=127.0.0.1
11 DB_PORT=3306
12 DB_DATABASE=kp2_digital
13 DB_USERNAME=root
14 DB_PASSWORD=
```

Gambar 14. Pengaturan untuk melakukan integrasi *database*

2) API dan Integrasi

API ini dirancang untuk menangani berbagai permintaan *HTTP* seperti *GET*, *POST*, *PUT*, dan *DELETE*, yang memudahkan pengelolaan data dari sisi *client*. Pada Gambar 15 akan menjelaskan mengenai sebuah fungsi yang memanggil *REST API GET*.



```
1 getListMatakuliah() {
2     this.isBusyMatakuliah = true;
3     Axios.get(`${config.apiMahasiswaUrl}/matakuliah`)
4         .then((response) => {
5         if (response.data.data.length === 0) {
6             this.kumpulan_matakuliah = [];
7             this.totalRows = 0;
8         } else {
9             this.kumpulan_matakuliah = response.data.data;
10            this.totalRows = response.data.data.length;
11        }
12        this.isBusyMatakuliah = false;
13    })
14    .catch((error) => {
15        console.error(error);
16        this.kumpulan_matakuliah = [];
17        this.totalRows = 0;
18        this.isBusyMatakuliah = false;
19    });
20 }
```

Gambar 15. Fungsi menampilkan kumpulan mata kuliah

Pada Gambar 16 merupakan rute API yang telah dipanggil sebelumnya, jika terdapat *request* berbentuk *GET* dengan rute mahasiswa/matakuliah maka program akan otomatis memanggil fungsi *index* pada file *MatakuliahController*.



```
1 Route::get('mahasiswa/matakuliah', 'MatakuliahController@index');
```

Gambar 16. Rute API GET Mata kuliah

Pada Gambar 17 merupakan fungsi indeks yang akan dipanggil, fungsi indeks akan mendeklarasi variabel *mataKuliah* dengan seluruh data pada tabel *MataKuliah*. Lalu akan mengembalikannya dalam bentuk *collection*.



```
1 public function index()  
2 {  
3     $mataKuliah = MataKuliah::all();  
4     return MataKuliahResources::collection($mataKuliah);  
5 }
```

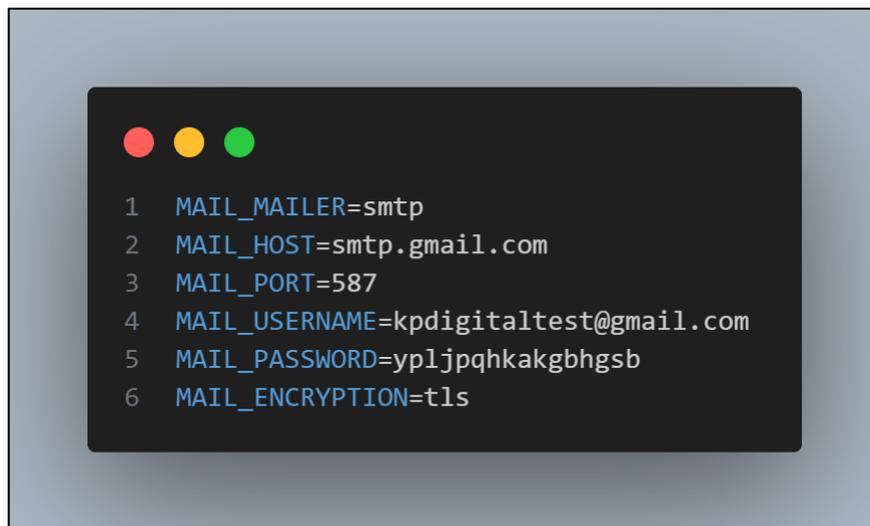
Gambar 17. Fungsi *index* pada *MatakuliahController*

C. Implementasi Notifikasi Email

Fitur notifikasi email sangat penting untuk memberitahu dosen penguji, pembimbing, dan mahasiswa mengenai jadwal sidang yang sudah ditetapkan oleh koordinator KP.

1) Konfigurasi Pengaturan Email

File yang memiliki *extension* *.env* merupakan sebuah *file* yang menyimpan pengaturan aplikasi, termasuk konfigurasi *email*. Pada Gambar 18 merupakan konfigurasi email pada *website* kerja praktik digital.



```
1 MAIL_MAILER=smtp  
2 MAIL_HOST=smtp.gmail.com  
3 MAIL_PORT=587  
4 MAIL_USERNAME=kpdigitaltest@gmail.com  
5 MAIL_PASSWORD=ypljppqhkakgbhgsb  
6 MAIL_ENCRYPTION=tls
```

Gambar 18. Pengaturan pada file *.env*

2) Membuat Logika Email Notifikasi

Pada Gambar 19 merupakan sebuah controller yang akan memproses logika pengiriman dan validasi email yang akan dikirimkan kepada dosen penguji, pembimbing, dan mahasiswa.

```
1 public function sendEmail(Request $request, $id_sidang)
2 {
3     // Validate incoming request data
4     $request->validate([
5         'recipients' => 'required|array',
6         'recipients.*' => 'email', // Ensure all recipients are valid email addresses
7         'subject' => 'required|string',
8         'message' => 'required|string',
9     ]);
10
11     // Find the 'sidang' by its ID
12     $sidang = Sidang::find($id_sidang);
13
14     if (!$sidang) {
15         return response()->json(['message' => 'Sidang tidak ditemukan.'], 404);
16     }
17
18     // Get the list of recipients from the request
19     $recipients = $request->input('recipients');
20     $subject = $request->input('subject');
21     $message = $request->input('message');
22
23     // Create the email data payload
24     $emailData = [
25         'sidang' => $sidang,
26         'subject' => $subject,
27         'message' => $message,
28     ];
29     return Mail::to($recipients)->send(new SidangNotification($emailData));
30     // Send the email using Laravel's Mail facade
31     try {
32         Mail::to($recipients)->send(new SidangNotification($emailData)); // Assuming you have a 'ScheduleMail' mailable
33         return response()->json(['message' => 'Email berhasil dikirim.'], 200);
34     } catch (\Exception $e) {
35         return response()->json(['message' => 'Gagal mengirim email.', 'error' => $e->getMessage()], 500);
36     }
37 }
```

Gambar 19. EmailController

Pada Gambar 20 merupakan badan *email* dan template email yang akan dikirim oleh sistem ke *email* yang tertera pada *recipients*.

```
1 openEmailModal() {
2     this.prepareEmailRecipients()
3     .then((recipients) => {
4         this.emailData.recipients = recipients.join(', '); // Gabungkan email dengan koma
5         const hari = this.getDayName(this.tanggal_sidang);
6
7         // Menyusun subjek dan pesan email
8         this.emailData.subject = `Pemberitahuan Jadwal Sidang Anda`;
9         this.emailData.message = `
10     Yth. Bapak/Ibu dan Mahasiswa,
11
12     Berikut adalah detail jadwal sidang Anda:
13
14     Hari: ${hari}
15     Tanggal: ${this.tanggal_sidang}
16     Waktu: ${this.waktu_mulai_sidang} - ${this.waktu_akhir_sidang}
17     Ruangan: ${this.ruangan_sidang}
18
19     Penguji:
20     - ${this.sidang.penguji_sidang || 'Penguji 1 tidak ditentukan'}
21     - ${this.sidang.penguji_sidang_dua || 'Penguji 2 tidak ditentukan'}
22
23     Pembimbing:
24     - ${this.sidang.topik.pembimbing || 'Pembimbing Utama tidak ditentukan'}
25     - ${this.sidang.topik.pembimbing_lapangan || 'Pembimbing Lapangan tidak ditentukan'}
26
27     Mahasiswa:
28     - ${this.sidang.nama || 'Nama Mahasiswa tidak diketahui'}
29
30     Mohon hadir tepat waktu.
31
32     Terima kasih.
33     `;
34
35     this.showEmailModal = true;
36     this.$refs.emailModal.show();
37     });
38     .catch((error) => {
39         console.error(error);
40     });
41 },
```

Gambar 20. Email yang akan dikirim

V. PENGUJIAN

A. Pengujian Black Box

Dengan memahami konsep penerapan dalam pengujian black box, dapat mengidentifikasi kesalahan fungsional dan memastikan kompatibilitas antar komponen. Selain itu dengan pengujian black box dapat memberikan sebuah kasus nyata untuk memperjelas penerapannya dalam proyek perangkat lunak. Pada bagian pengujian black box akan dibagi menjadi beberapa bagian berdasarkan fitur yang telah diimplementasikan.

1) Fitur Pengajuan MBKM

TABEL I
PENGUJIAN FITUR MBKM

Aktivitas Pengujian	Case Skenario	Output	Kesimpulan
Submit dengan form terisi penuh	Input: Nama Instansi: Instansi Test Deskripsi: Test Deskripsi Jenis MBKM: Test MBKM	Pendaftaran berhasil, menampilkan status	Berhasil
Submit form kosong	Input: Field form tidak diisi	Pesan error: Semua bagian harus diisi.	Berhasil
Memasuki halaman ketika sudah mengisi	Input: -	Menampilkan status	Berhasil

2) Fitur Konversi SKS

TABEL II
PENGUJIAN FITUR KONVERSI SKS

Aktivitas Pengujian	Case Skenario	Output	Kesimpulan
Konversi SKS sesuai dengan 20 SKS	Input: Total konversi mencapai 20 SKS	Pengajuan konversi SKS berhasil dilakukan, menampilkan status pengajuan	Berhasil
Konversi SKS kurang dari 20 SKS	Input: Total konversi kurang dari 20 SKS	Tombol <i>submit</i> tidak dapat ditekan	Berhasil
Konversi SKS lebih dari 20 SKS	Input: Total konversi lebih dari 20 SKS	Tombol <i>submit</i> tidak dapat ditekan	Berhasil

3) Fitur Upload Dokumen Mahasiswa

TABEL III
PENGUJIAN FITUR UPLOAD DOKUMEN MAHASISWA

Aktivitas Pengujian	Case Skenario	Output	Kesimpulan
Upload dokumen .pdf	Input: Upload dokumen .pdf	Dokumen berhasil diupload	Berhasil
Upload dokumen .pdf di bagian <i>input</i> yang tidak menerima format .pdf	Input: Upload dokumen selain format .pdf	Pesan error: Dokumen tidak sesuai dengan format.	Berhasil

4) Fitur Download Dokumen Template

TABEL IV
PENGUJIAN FITUR DOWNLOAD TEMPLATE

Aktivitas Pengujian	Case Skenario	Output	Kesimpulan
<i>Download template logbook</i>	Menekan tombol <i>download</i> pada bagian <i>logbook</i>	Mengunduh dokumen <i>logbook</i>	Berhasil
<i>Download template laporan</i>	Menekan tombol <i>download</i> pada bagian laporan	Mengunduh dokumen laporan	Berhasil
<i>Download template HistoryKPTA</i>	Menekan tombol <i>download</i> pada bagian HistoryKPTA	Mengunduh dokumen HistoryKPTA	Berhasil
<i>Download template FormBimbinganTA</i>	Menekan tombol <i>download</i> pada bagian FormBimbinganTA	Mengunduh dokumen FormBimbinganTA	Berhasil
<i>Download template Imas</i>	Menekan tombol <i>download</i> pada bagian Imas	Mengunduh dokumen Imas	Berhasil

5) Fitur Upload Dokumen Template

TABEL V
PENGUJIAN FITUR UPLOAD TEMPLATE

Aktivitas Pengujian	Case Skenario	Output	Kesimpulan
<i>Upload template logbook</i>	Menekan tombol <i>upload</i> pada bagian <i>logbook</i>	Mengunggah dokumen <i>logbook</i>	Berhasil
<i>Upload template laporan</i>	Menekan tombol <i>upload</i> pada bagian laporan	Mengunggah dokumen laporan	Berhasil
<i>Upload template HistoryKPTA</i>	Menekan tombol <i>upload</i> pada bagian HistoryKPTA	Mengunggah dokumen HistoryKPTA	Berhasil
<i>Upload template FormBimbinganTA</i>	Menekan tombol <i>upload</i> pada bagian FormBimbinganTA	Mengunggah dokumen FormBimbinganTA	Berhasil
<i>Upload template Imas</i>	Menekan tombol <i>upload</i> pada bagian Imas	Mengunggah dokumen Imas	Berhasil

6) Fitur Jenis MBKM

TABEL VI
PENGUJIAN FITUR JENIS MBKM

Aktivitas Pengujian	Case Skenario	Output	Kesimpulan
Menambahkan jenis MBKM baru.	Mengisi nama jenis MBKM	MBKM berhasil ditambahkan!	Berhasil
Menambahkan jenis MBKM baru.	Tidak mengisi nama jenis MBKM	Nama MBKM tidak boleh kosong.	Berhasil
Merubah jenis MBKM.	Mengisi nama jenis MBKM	MBKM berhasil diperbarui!	Berhasil
Merubah jenis MBKM.	Tidak mengisi nama jenis MBKM	Nama MBKM tidak boleh kosong.	Berhasil
Menghapus jenis MBKM	Menekan tombol <i>delete</i> .	MBKM berhasil dihapus!	Berhasil

7) Fitur Mata Kuliah

TABEL VII
PENGUJIAN FITUR MATA KULIAH

Aktivitas Pengujian	Case Skenario	Output	Kesimpulan
Menambahkan jenis mata kuliah baru.	Seluruh <i>field</i> terisi	Mata Kuliah berhasil ditambahkan!	Berhasil
Menambahkan jenis mata kuliah baru.	Pada <i>field</i> SKS, dan ID terisi oleh huruf	Sistem tidak mengijinkan huruf pada <i>field</i> angka	Berhasil
Menambahkan jenis mata kuliah baru.	Seluruh <i>field</i> tidak terisi	Nama, Jumlah SKS, dan Jenis Semester tidak boleh kosong.	Berhasil
Merubah jenis mata kuliah.	Seluruh <i>field</i> terisi	Mata Kuliah berhasil diperbarui!	Berhasil
Merubah jenis mata kuliah.	Tidak mengisi <i>field</i> Semester	Terjadi kesalahan	Berhasil
Menghapus jenis mata kuliah	Menekan tombol <i>delete</i> .	Mata Kuliah berhasil dihapus!	Berhasil

8) Fitur Download Dokumen Mahasiswa

TABEL VIII
PENGUJIAN FITUR DOWNLOAD DOKUMEN MAHASISWA

Aktivitas Pengujian	Case Skenario	Output	Kesimpulan
<i>Download logbook</i>	Menekan tombol <i>download</i> pada bagian <i>logbook</i>	Mengunduh dokumen <i>logbook</i>	Berhasil
<i>Download</i> laporan	Menekan tombol <i>download</i> pada bagian laporan	Mengunduh dokumen laporan	Berhasil
<i>Download</i> HistoryKPTA	Menekan tombol <i>download</i> pada bagian HistoryKPTA	Mengunduh dokumen HistoryKPTA	Berhasil
<i>Download</i> FormBimbinganTA	Menekan tombol <i>download</i> pada bagian FormBimbinganTA	Mengunduh dokumen FormBimbinganTA	Berhasil
<i>Download</i> Imas	Menekan tombol <i>download</i> pada bagian Imas	Mengunduh dokumen Imas	Berhasil

9) Fitur Konfirmasi MBKM Mahasiswa

TABEL IX
PENGUJIAN FITUR KONFIRMASI MBKM

Aktivitas Pengujian	Case Skenario	Output	Kesimpulan
Menampilkan data pada halaman MBKM	Menunggu hasil yang dikirim oleh sistem	Seluruh data mahasiswa yang mengikuti MBKM ditampilkan	Berhasil
Menuju halaman <i>detail</i> MBKM	Menekan tombol detail pada mahasiswa yang diinginkan	Menuju halaman <i>detail</i> MBKM	Berhasil

10) Fitur Detail Konfirmasi MBKM Mahasiswa

TABEL X
PENGUJIAN DETAIL FITUR KONFIRMASI MBKM

Aktivitas Pengujian	Case Skenario	Output	Kesimpulan
Menampilkan data pada halaman MBKM	Menunggu hasil yang dikirim oleh sistem	Seluruh data MBKM dan Konversi SKS mahasiswa yang dipilih ditampilkan	Berhasil
Konversi SKS sesuai dengan 20 SKS	Input: Total konversi mencapai 20 SKS	Pengajuan konversi SKS berhasil dilakukan, menampilkan status pengajuan	Berhasil
Konversi SKS kurang dari 20 SKS	Input: Total konversi kurang dari 20 SKS	Tombol <i>submit</i> tidak dapat ditekan	Berhasil
Konversi SKS lebih dari 20 SKS	Input: Total konversi lebih dari 20 SKS	Tombol <i>submit</i> tidak dapat ditekan	Berhasil

VI. KESIMPULAN

Penambahan serta perubahan fitur pada website Kerja Praktik digital meningkatkan kemudahan dan efisiensi bagi penggunaannya. Fitur jalur MBKM yang ditambahkan memungkinkan mahasiswa untuk melakukan pendaftaran MBKM, konversi SKS, pengajuan topik, hingga sidang secara digital. Koordinator kerja praktik juga terbantu dengan fitur rekap data mahasiswa serta unggah dan unduh dokumen seperti laporan, *logbook*, *history* KPTA, Form Imas, dan *Form* Bimbingan. Selain itu, fitur notifikasi email mempermudah pengingat bagi dosen penguji, dosen pembimbing, dan mahasiswa terkait jadwal sidang.

Pengembangan jalur MBKM menggunakan *Vue.js* sebagai *framework frontend* untuk menciptakan antarmuka yang interaktif dan dinamis, serta *Laravel* sebagai *backend* untuk mengelola logika sistem, database, dan keamanan. Kombinasi kedua teknologi ini menghasilkan website Kerja Praktik digital yang efisien, responsif, dan memiliki pengalaman pengguna yang lebih baik. *Vue.js* memungkinkan tampilan modern yang mudah diinteraksikan, sementara *Laravel* memastikan pengelolaan data yang optimal serta keamanan sistem, memberikan dasar yang kuat bagi implementasi jalur MBKM yang terstruktur.

DAFTAR PUSTAKA

- [1] Muhtar, "Mengenal Website, Manfaat, dan Jenis-jenisnya," 2024.
- [2] S. A. Kinanti, "Maintenance Website : Kapan Perlu Dilakukan dan Apa Manfaatnya?," jagoweb, 2022.
- [3] S. S. Indonesia, "Website Bisnis Anda Perlu Revamp? Inilah 4 Tahapannya!," 2023.
- [4] T. R. Rivanthio, "PERANCANGAN PENGAJUAN SIDANG LAPORAN PRAKTEK KERJA LAPANGAN MAHASISWA BERBASIS WEBSITE PADA SEKOLAH TINGGI ANALIS BAKTI ASIH BANDUNG," *Jurnal Teknologi Informasi Dan Komunikasi*, vol. 7, p. 109, 2020.
- [5] E. S. Alim, "Sistem Penilaian Sidang Skripsi Berbasis Website," *Jurnal Penerapan Sistem Informasi (Komputer & Manajemen)*, vol. 4, no. 4, pp. 996-1005, 2023.
- [6] S. P. S. SKRIPSI, "Aditya Yuliskandar," *Teknik Informatika Universitas Muhammadiyah Yogyakarta*, 2018.
- [7] A. R. Mariana, "Sistem Informasi Aplikasi Penilaian Sidang Skripsi," *JURNAL SISFOTEK GLOBAL*, vol. 3, no. 2, p. 19, 2013.
- [8] N. J., "Designing Web Usability: The Practice of Simplicity," *IEEE Software*, vol. 17, pp. 75-77.
- [9] E. Castro, *HTML for the World Wide Web*, Peachpit Press, 2024.
- [10] S. M. A. T. P. Jensen, "Type Analysis for JavaScript," *Lecture Notes in Computer Science*, vol. 5673, 2009.
- [11] R. Setiawan, "Apa itu Framework? Developer Wajib Tahu," Dicoding Indonesia, 2021.
- [12] J. Armel, *Web application development with Laravel PHP Framework version 4*, 2014.
- [13] K. Kyoreva, *State of the Art JavaScript Application Development*, Sofia: Proceedings of International Conference on Application of Information and Communication Technology and Statistics in Economy and Education, 2017.