



**KERJA PRAKTIK - IF184801**

**Pengembangan Sistem Informasi Keuangan (SIMKEU) pada  
Direktorat Riset dan Pengabdian kepada Masyarakat  
(DRPM) dengan Kerangka Kerja Laravel dan Next.js**

Gedung Pusat Riset ITS

Jl. Teknik Kimia, Kampus ITS Keputih, Sukolilo, Surabaya, Jawa  
Timur, 60111

Periode: 11 September 2023 – 17 Desember 2023

**Oleh:**

Zahra Fayyadiyah

5025201133

**Pembimbing Jurusan**

Siska Arifiani, S.Kom., M.Kom.

**Pembimbing Lapangan**

Pramandha Sosiawan Nugroho, A.Md.

DEPARTEMEN TEKNIK INFORMATIKA

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya 2023



**KERJA PRAKTIK - IF184801**

**Pengembangan Sistem Informasi Keuangan (SIMKEU) pada  
Direktorat Riset dan Pengabdian kepada Masyarakat  
(DRPM) dengan Kerangka Kerja Laravel dan Next.js**

Gedung Pusat Riset ITS

Jl. Teknik Kimia, Kampus ITS Keputih, Sukolilo, Surabaya, Jawa  
Timur, 60111

Periode: 11 September 2023 – 17 Desember 2023

Oleh:

Zahra Fayyadiyahati

5025201133

**Pembimbing Jurusan**

Siska Arifiani, S.Kom., M.Kom.

**Pembimbing Lapangan**

Pramandha Sosiawan Nugroho, A.Md.

DEPARTEMEN TEKNIK INFORMATIKA

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya 2023

*[Halaman ini sengaja dikosongkan]*

## DAFTAR ISI

|                                     |      |
|-------------------------------------|------|
| DAFTAR ISI                          | iv   |
| DAFTAR GAMBAR                       | viii |
| DAFTAR TABEL                        | x    |
| LEMBAR PENGESAHAN                   | xii  |
| KATA PENGANTAR                      | xvi  |
| BAB I PENDAHULUAN                   | 1    |
| 1.1. Latar Belakang                 | 1    |
| 1.2. Tujuan                         | 2    |
| 1.3. Manfaat                        | 3    |
| 1.4. Rumusan Masalah                | 3    |
| 1.5. Lokasi dan Waktu Kerja Praktik | 3    |
| 1.6. Metodologi Kerja Praktik       | 4    |
| 1.6.1. Perumusan Masalah            | 4    |
| 1.6.2. Studi Literatur              | 4    |
| 1.6.3. Implementasi Sistem          | 4    |
| 1.6.4. Pengujian dan Evaluasi       | 4    |
| 1.6.5. Kesimpulan dan Saran         | 5    |
| 1.7. Sistematika Laporan            | 5    |
| 1.7.1. Bab I Pendahuluan            | 5    |
| 1.7.2. Bab II Profil Perusahaan     | 5    |
| 1.7.3. Bab III Tinjauan Pustaka     | 5    |

|   |  |           |
|---|--|-----------|
| 1.7.4.  | Bab IV Analisis dan Perancangan Infrastruktur Sistem | 5         |
| 1.7.5.  | Bab V Implementasi Sistem                            | 5         |
| 1.7.6.  | Bab VI Pengujian dan Evaluasi                        | 6         |
| 1.7.7.  | Bab VII Kesimpulan dan Saran                         | 6         |
| <b>BAB II PROFIL PERUSAHAAN</b>                             |  | <b>7</b>  |
| 2.1.  | Profil   | 7         |
| 2.2.  | Lokasi   | 8         |
| <b>BAB III TINJAUAN PUSTAKA</b>                             |  | <b>9</b>  |
| 3.1.  | Sistem Informasi Berbasis Web                        | 9         |
| 3.2.  | Laravel  | 9         |
| 3.3.  | Application Programming Interface (API)              | 10        |
| 3.4.  | React  | 10        |
| 3.5.  | Next.js  | 11        |
| 3.6.  | MySQL  | 11        |
| 3.7.  | Decoupled Architecture                               | 11        |
| 3.8.  | cPanel   | 12        |
| 3.9.  | Vercel   | 12        |
| <b>BAB IV ANALISIS DAN PERANCANGAN INFRASTRUKTUR SISTEM</b> |  | <b>13</b> |
| 4.1.  | Analisis Sistem                                      | 13        |
| 4.1.1.  | Definisi Umum Aplikasi                               | 13        |
| 4.1.2.  | Analisis Kebutuhan                                   | 13        |
| 4.2.  | Perancangan Infrastruktur Sistem                     | 14        |

|  |           |
|--|-----------|
| <b>BAB V IMPLEMENTASI SISTEM</b>           | <b>15</b> |
| 5.1. Implementasi Database                 | 15        |
| 5.2. Implementasi Back-End dengan Laravel  | 21        |
| 5.3. Implementasi Front-End dengan Next.js | 26        |
| <b>BAB VI PENGUJIAN DAN EVALUASI</b>       | <b>29</b> |
| 6.1. Tujuan Pengujian                      | 29        |
| 6.2. Kriteria Pengujian                    | 29        |
| 6.3. Skenario Pengujian                    | 31        |
| 6.1. Evaluasi Pengujian                    | 33        |
| <b>BAB VII KESIMPULAN DAN SARAN</b>        | <b>35</b> |
| 7.1. Kesimpulan                            | 35        |
| 7.2. Saran                                 | 35        |
| <b>DAFTAR PUSTAKA</b>                      | <b>38</b> |
| <b>BIODATA PENULIS</b>                     | <b>40</b> |

*[Halaman ini sengaja dikosongkan]*

## DAFTAR GAMBAR

|   |    |
|---|----|
| Gambar 3.1 Arsitektur Monolithic dan Decoupled.....         | 12 |
| Gambar 5.2 Diagram CDM SIMKEU .....                         | 21 |
| Gambar 5.2 Diagram Layer pada Back-End.....                 | 21 |
| Gambar 5.3 Route API.....                                   | 22 |
| Gambar 5.4 Route API Repository .....                       | 23 |
| Gambar 5.5 Struktur Folder Controllers .....                | 23 |
| Gambar 5.6 Source Code Controller.php.....                  | 24 |
| Gambar 5.7 Struktur Folder Repositories .....               | 25 |
| Gambar 5.8 Source Code Repository.php .....                 | 25 |
| Gambar 5.9 Source Code Fungsi Inisialisasi .....            | 26 |
| Gambar 5.10 Penggunaan useEffect untuk Pemrosesan Data..... | 27 |



*[Halaman ini sengaja dikosongkan]*

## DAFTAR TABEL

|  |    |
|--|----|
| Tabel 6.1 Matriks Pengujian.....         | 31 |
| Tabel 6.2 Matriks Pengujian (Hasil)..... | 33 |

*[Halaman ini sengaja dikosongkan]*

**LEMBAR PENGESAHAN  
KERJA PRAKTIK**

**Pengembangan Sistem Informasi Keuangan (SIMKEU) pada  
Direktorat Riset dan Pengabdian kepada Masyarakat  
(DRPM) dengan Kerangka Kerja Laravel dan Next.js**

Oleh:

Zahra Fayyadiyah

5025201133

Disetujui oleh Pembimbing Kerja Praktik:

1. Siska Arifiani, S.Kom.,  
M.Kom.  
NIP. 1990202012034



(Pembimbing Departemen)

2. Pramandha Sosiawan  
Nugroho, A.Md  
NIP. 1991201821405



(Pembimbing Lapangan)

*[Halaman ini sengaja dikosongkan]*

# **Pengembangan Sistem Informasi Keuangan (SIMKEU) pada Direktorat Riset dan Pengabdian kepada Masyarakat (DRPM) dengan Kerangka Kerja Laravel dan Next.js**

Nama Mahasiswa : Zahra Fayyadiyah  
NRP : 5025201133  
Departemen : Teknik Informatika FTEIC-ITS  
Pembimbing Departemen : Siska Arifiani, S.Kom., M.Kom.  
Pembimbing Lapangan : Pramandha Sosiawan N., A.Md.

## **ABSTRAK**

*Direktorat Riset dan Pengabdian kepada Masyarakat (DRPM) ITS adalah salah satu birokrasi dalam naungan ITS yang mengurus perihal pelaksanaan riset dan pengabdian kepada masyarakat. Supaya setiap instansi di bawah DRPM dapat melaksanakan kegiatan-kegiatan tersebut dengan lancar, dilaksanakan pengembangan Sistem Informasi Keuangan (SIMKEU) yang dapat melayani kebutuhan administrasi keuangan. SIMKEU menggunakan alur alokasi keuangan yang mirip dengan perusahaan/instansi/organisasi pada umumnya, yang mana alur tersebut dimulai dari alokasi uang (pagu), kemudian pembuatan kegiatan beserta rancangan anggaran biayanya (RBA) dan diikuti oleh monitoring transaksi pengeluaran, pemasukan, pengembalian untuk pengurusan surat pertanggungjawaban (SPJ). Sistem informasi ini dirancang dengan decoupled architecture menggunakan tech stack Laravel dan Next.js.*

***Kata Kunci: Sistem Informasi Keuangan, Decoupled Architecture, Laravel, Next.js, Administrasi Keuangan, Monitoring Transaksi Keuangan, Alokasi Keuangan***

*[Halaman ini sengaja dikosongkan]*

## **KATA PENGANTAR**

Dengan menyebut nama Allah SWT yang Maha Pengasih lagi Maha Penyayang, kami panjatkan segala puji syukur atas kehadiran-Nya sehingga saya dapat menyelesaikan salah satu kewajiban dalam menulis laporan dan menunaikan Kerja Praktik salah satu persyaratan kelulusan mahasiswa Departemen Teknik Informatika ITS. Kerja Praktik ini saya lakukan dalam cakupan tajuk Perancangan dan Implementasi Sistem Informasi Keuangan (SIMKEU) Direktorat Riset dan Pengabdian kepada Masyarakat (DRPM).

Kerja Praktik dan penyusunan laporan ini telah saya lakukan dengan maksimal. Akan tetapi, saya menyadari akan adanya kekurangan dalam pelaksanaan kedua hal tersebut. Meskipun begitu, saya berharap laporan ini dapat dijadikan referensi untuk pengembangan berikutnya.

Melalui buku laporan ini saya juga ingin menyampaikan rasa terima kasih kepada orang-orang yang telah membantu saya selama kerja praktik ini baik secara langsung maupun tidak langsung.

1. Kedua orang tua penulis.
2. Ibu Siska Arifiani, S.Kom., M.Kom. selaku dosen pembimbing kerja praktik sekaligus koordinator kerja praktik.
3. Bapak Pramandha Sosiawan Nugroho, A.Md. selaku pembimbing lapangan selama kerja praktik berlangsung.
4. Teman-teman penulis yang senantiasa memberikan semangat ketika penulis melaksanakan KP.

Surabaya, 11 Desember 2023

Zahra Fayyadiyah



*[Halaman ini sengaja dikosongkan]*

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Sejak terciptanya teknologi komputer pada revolusi industri ke-3, banyak kebutuhan administrasi yang biasa dilakukan dengan kertas dan pulpen dipindahkan, dikelola dan disimpan dalam media digital menggunakan komputer. Pemindahan media ini memberikan banyak keuntungan pada peradaban manusia, seperti proses administrasi yang lebih cepat, pengelolaan yang lebih baik dan peningkatan efektivitas dalam aspek lainnya. Selain itu, proses kalkulasi pun juga menjadi akurat dan cepat. Hal ini dapat mengatasi masalah *human error* yang terkadang terjadi dalam proses-proses perekapan atau administrasi.

Pada era sekarang, hampir seluruh urusan administrasi sudah dilaksanakan secara digital. Namun, tidak seluruh proses administrasi memanfaatkan otomatisasi proses bisnis dikarenakan *platform-platform* perekapan seperti Microsoft Excel memiliki batasan-batasan tertentu dalam kustomisasi proses bisnis atau alur administrasi yang terjadi. Pada saat-saat seperti inilah seorang *developer* diperlukan untuk melakukan rancang bangun suatu aplikasi atau perangkat lunak yang dapat memenuhi kebutuhan alur administrasi atau proses bisnis yang spesifik.

Dalam struktur hierarki atas kumpulan organisasi atau instansi, terdapat *chain of commands* beserta alur koordinasi yang mengikat dan mengatur berbagai proses bisnis tertentu. Direktorat Riset dan Pengabdian kepada Masyarakat (DRPM) ITS adalah lembaga direktorat yang mengatur riset dan pengabdian masyarakat yang diadakan dalam lingkup kampus ITS. Banyaknya pihak serta

instansi-instansi yang ada di bawah naungan ITS menjadikan pelaksanaan kegiatan riset dan pengabdian masyarakat dijalankan dalam sistem desentralisasi. Hal ini berarti DRPM membina dan mengatur tata pelaksanaan kepada seluruh instansi mandiri yang mana tiap-tiap instansi tersebut menjalankan program kerja riset atau pengabdian masyarakatnya masing-masing.

Pelaksanaan riset dan pengabdian masyarakat tentunya membutuhkan sumber daya atau *resource* dari ITS, salah satunya ialah perihal keuangan. DRPM, sebagai direktorat yang menaungi proses pelaksanaan riset dan pengabdian masyarakat dalam ITS, secara berkala akan mengalokasikan dana kepada lembaga pusat dalam ITS yang berkaitan. Akan tetapi, proses alokasi keuangan ini memiliki proses bisnis yang cukup rumit sehingga penggunaan Microsoft Excel saja tidak bisa mengintegrasikan seluruh aspek yang ada dalam alur keuangan ini. Hal ini disebabkan karena proses keuangan yang terdiri dari tahap pemberian pagu, permintaan transaksi pemasukan, laporan transaksi pengeluaran serta transaksi pengembalian. Selain proses utama tersebut, DRPM juga harus melakukan *monitoring cash opname*, tahap status keuangan dalam tiap kegiatan beserta kepentingan keuangan lainnya.

Maka dari itu, DRPM melaksanakan proyek pengembangan perangkat lunak sistem informasi keuangan berbasis *website* yang dapat mengakomodasi semua proses keuangan yang dibutuhkan DRPM dalam melaksanakan tanggung jawabnya. Proyek ini dilaksanakan dibawah kerja sama dengan dosen departemen Teknik Informatika ITS.

## **1.2. Tujuan**

Tujuan kerja praktik ini adalah menyelesaikan kewajiban nilai kerja praktik sebesar 4 sks. Selain itu, kerja

praktik ini juga dilaksanakan untuk membantu Direktorat Riset dan Pengabdian Masyarakat ITS (DRPM) serta instansi-instansi yang berada di bawah naungannya untuk melaksanakan skema keuangan mereka secara terkontrol, terintegrasi, efisien dan akurat.

### **1.3. Manfaat**

Manfaat yang diperoleh dengan pembuatan perangkat lunak SIMKEU berbasis website yakni mengintegrasikan seluruh aspek pada alur administrasi keuangan untuk kegiatan dari lembaga-lembaga yang berada di dalam naungan tanggung jawab DRPM. Selain itu, juga meningkatkan efisiensi dalam perhitungan dan rekapan dari seluruh pemasukan, pengembalian, pengeluaran dan *monitoring* keuangan.

### **1.4. Rumusan Masalah**

Rumusan masalah dari kerja praktik ini adalah sebagai berikut.

1. Bagaimana rancangan proses bisnis dari administrasi keuangan DRPM ITS dalam SIMKEU?
2. Bagaimana arsitektur beserta teknologi (*tech stack*) yang digunakan dalam pengembangan SIMKEU?
3. Bagaimana implementasi dari SIMKEU yang sesuai dengan rancangan dapat melayani kebutuhan administrasi keuangan DRPM ITS?

### **1.5. Lokasi dan Waktu Kerja Praktik**

Sehubungan karena proyek pengembangan perangkat lunak yang dapat dilaksanakan secara jauh, lokasi kerja praktik bersifat *hybrid*, yakni secara *remote* (*online*) dan secara *offline* di Gedung Pusat Riset kantor

DRPM ITS. Adapun kerja praktik dimulai pada tanggal 11 September 2023 hingga 17 Desember 2023.

## **1.6. Metodologi Kerja Praktik**

Metodologi dalam pembuatan buku kerja praktik meliputi sebagai berikut.

### **1.6.1. Perumusan Masalah**

Sebelum memulai pengembangan perangkat lunak SIMKEU, dilaksanakan wawancara beserta rapat dengan DRPM terkait kebutuhan apa saja yang perlu dilayani oleh SIMKEU. Dari hasil wawancara tersebut, akan dibuat perumusan masalah dari kerja praktik ini.

### **1.6.2. Studi Literatur**

Setelah mendapat gambaran besar terkait sistem yang dibutuhkan oleh pihak DRPM, dilakukan studi literatur terkait teknologi beserta alur algoritma yang dapat menyelesaikan masalah yang telah didapatkan dari perumusan masalah.

### **1.6.3. Implementasi Sistem**

Pada tahap metodologi ini, akan dilakukan implementasi sistem menggunakan rancangan, teknologi dan algoritma yang telah dikaji pada studi literatur.

### **1.6.4. Pengujian dan Evaluasi**

Setelah semua fitur SIMKEU selesai diimplementasikan, akan dilakukan pengujian fitur untuk menilai apakah *website* SIMKEU sudah dapat menjalankan semua fitur yang direncanakan atau diperlukan perbaikan.

### **1.6.5. Kesimpulan dan Saran**

Bagian ini adalah kesimpulan dari aplikasi SIMKEU yang telah diimplementasikan dan diuji beserta saran bagi SIMKEU untuk pengembangan kedepannya.

## **1.7. Sistematika Laporan**

Sistematika laporan dalam buku kerja praktik ini adalah sebagai berikut.

### **1.7.1. Bab I Pendahuluan**

Bagian ini mencakup pengantar, maksud, kegunaan, perumusan permasalahan, lokasi dan jangka waktu praktik, metode, serta struktur laporan.

### **1.7.2. Bab II Profil Perusahaan**

Bagian ini berisi deskripsi umum dari Direktorat Riset dan Pengabdian kepada Masyarakat (DRPM) ITS dalam perihal profil beserta lokasi perusahaan.

### **1.7.3. Bab III Tinjauan Pustaka**

Bagian ini mencakup fondasi konseptual dari teknologi yang digunakan dalam mengimplementasikan proyek dari kerja praktik ini. Hal tersebut berisi dasar teori dan teknologi yang digunakan dalam menyelesaikan proyek kerja praktik.

### **1.7.4. Bab IV Analisis dan Perancangan Infrastruktur Sistem**

Bab ini berisi mengenai analisis dan rancangan dari aplikasi *website* SIMKEU.

### **1.7.5. Bab V Implementasi Sistem**

Bagian ini berisi uraian perihal tahap-tahap yang dilakukan dalam proses implementasi aplikasi.

**1.7.6. Bab VI Pengujian dan Evaluasi**

Bagian ini berisi hasil uji coba dan evaluasi dari aplikasi yang telah dikembangkan selama pelaksanaan kerja praktik.

**1.7.7. Bab VII Kesimpulan dan Saran**

Bab ini berisi kesimpulan dan saran terhadap pengembangan SIMKEU dan juga proses pelaksanaan kerja praktik.

## **BAB II**

### **PROFIL PERUSAHAAN**

#### **2.1. Profil**

Direktorat Riset dan Pengabdian kepada Masyarakat (DRPM) ITS adalah instansi di bawah naungan Institut Teknologi Sepuluh Nopember (ITS) yang mengatur bidang riset dan pengabdian Masyarakat yang berjalan di dalam ITS. Antara lain, fungsi dari Lembaga DRPM ITS sendiri adalah sebagai berikut.

1. Menyusun Rencana Strategis, Rencana Operasional, Rencana Tahunan dan Rencana Anggaran Pendapatan dan Belanja Lembaga yang terkait dengan Penelitian dan Pelayanan masyarakat sebagai penjabaran Renstra ITS
2. Mengembangkan dan Melaksanakan program peningkatan kinerja dalam bidang penelitian dan pengabdian kepada masyarakat secara berkelanjutan berdasarkan prinsip ekonomi dan akuntabilitas
3. Mengembangkan dan melaksanakan sistem penjaminan dan pengendalian mutu penelitian dan pelayanan masyarakat
4. Mengkoordinasikan, mengintegrasikan dan mengendalikan mutu semua kegiatan penelitian dan pelayanan masyarakat dari pusat studi
5. Meningkatkan hubungan eksternal dalam rangka pemenuhan kebutuhan sumber daya DRPM



## **2.2. Lokasi**

Lokasi dari kantor DRPM ITS berada di Gedung Pusat Riset ITS yang berada di Jl. Teknik Kimia, Kampus ITS Keputih, Sukolilo, Surabaya, Jawa Timur Lt. 2.

## **BAB III**

### **TINJAUAN PUSTAKA**

#### **3.1. Sistem Informasi Berbasis Web**

Sistem informasi adalah perangkat lunak yang dibutuhkan dalam suatu organisasi atau perusahaan untuk memproses kumpulan informasi menjadi suatu hasil atau kesimpulan yang dapat digunakan untuk kebutuhan perusahaan (Sulistiani et al., 2022). Sementara itu, pada tahun 2023, pengembangan perangkat lunak berbasis website sedang menjadi tren karena adanya nodejs yang membawa *runtime environment* V8, yang merupakan *runtime environment* dari Google Chrome, kepada CPU. Hal ini menjadikan aplikasi yang dibangun di atas JavaScript menggunakan *runtime* nodejs dapat dijalankan oleh berbagai macam komputer, terlepas dari OS komputer tersebut pada *browser* komputer masing-masing.

Karena adanya tren tersebut, sistem informasi yang dikembangkan akhir-akhir ini cenderung merupakan sistem informasi berbasis website yang dibuat di atas *runtime* nodejs. Dengan sistem informasi berbasis *website*, data atau informasi yang ada pada sistem informasi akan sinkron pada setiap *device* karena terhubung dengan server yang ada pada jaringan internet. Selain itu, penggunaan dari perspektif pengguna juga menjadi lebih mudah karena pengguna tidak perlu melakukan instalasi perangkat lunak, cukup mengakses sistem informasi tersebut melalui URL di *browser*.

#### **3.2. Laravel**

Laravel merupakan kerangka kerja atau *framework* Bahasa pemrograman PHP. Laravel banyak digunakan dalam pengembangan *website*, salah satunya adalah SIMKEU versi sebelumnya. Laravel digemari oleh

banyak pengembang karena teknologi dalam Laravel yang dapat membuat pengembangan *website* jadi lebih cepat dibandingkan dengan penggunaan *framework* lainnya. Laravel dapat digunakan untuk *full-stack*, yang berarti Laravel akan mengurus tampilan serta logika program dan juga sebagai *back-end*, yang berarti sistem Laravel menyediakan *service* berupa API ke bagian *front-end* yang terpisah.

### 3.3. Application Programming Interface (API)

API atau *application programming interface* adalah suatu layanan yang menjalankan berbagai fungsi tertentu pada suatu *interface* yang memiliki hibingan dengan API tersebut. *Interface* dari API dapat berupa berbagai macam, baik itu *front-end* dari suatu *website* ataupun suatu aplikasi tertentu. Pada pengembangan *website*, API akan disediakan oleh *back-end* yang kemudian akan berkomunikasi dengan suatu *front-end*. API yang biasa digunakan dalam aplikasi *website* adalah REST API yang menyediakan metode komunikasi berupa GET, POST, PUT dan DELETE.

### 3.4. React

React adalah *library* dari JavaScript yang biasa digunakan untuk pengembangan *front-end* suatu *website*. Yang menyebabkan React mejadi salah satu *library* JavaScript yang digemari banyak pengembang adalah teknologi *hook* yang dapat mengatur *state* dalam *interface* *website*. Selain itu, *hook* juga dapat mendeteksi adanya perubahan *variable* yang ada dalam *interface* apabila pengguna melakukan interaksi dengan komponen-komponen yang berhubungan dengan *variable* tersebut.

### 3.5. Next.js

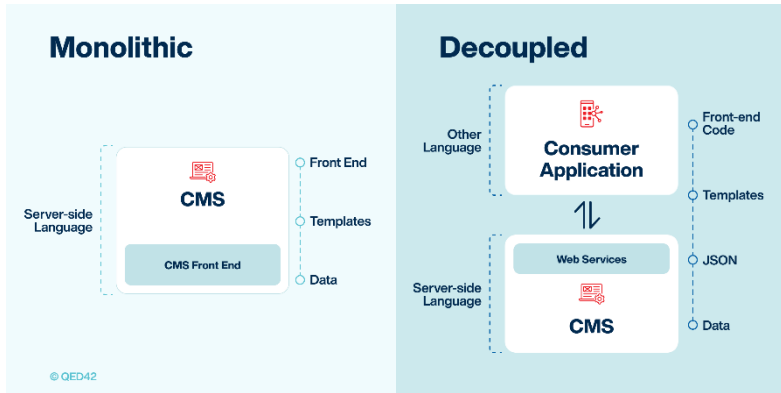
Next.js adalah kerangka kerja atau *framework* yang mengutilisasi library React. Next.js memungkinkan pengembang mengatur mana saja bagian yang akan menggunakan pendekatan *server side rendering* (SSR), *client side rendering* (CSR) atau *static site generation* (SSG). Selain itu, navigasi URL pada *interface* yang dibangun di atas Next.js akan sesuai dengan nama *file* dan lokasi direktori dari laman *file* tersebut di dalam struktur folder Next.js.

### 3.6. MySQL

MySQL adalah salah satu database management system (DBMS) yang banyak digunakan oleh *website* termasuk SIMKEU pada versi sebelumnya.

### 3.7. Decoupled Architecture

*Decoupled architecture* adalah arsitektur perangkat lunak yang memisahkan antara *interface*, API *back-end* sehingga *interface* dapat berjalan tanpa dependensi dari layanan API aplikasi tersebut. Hal ini menjadikan pengembangan *website* lebih fleksibel dan lebih kokoh karena tidak adanya dependensi. Berbeda dengan *headless architecture* yang mana suatu aplikasi memiliki banyak *interface* yang bergantung dengan *device* yang digunakan, *decoupled architecture* menjadikan API *back-end* diciptakan untuk hanya melayani satu *interface* saja. Arsitektur ini juga berbeda dengan arsitektur *monolithic* yang mana aplikasi bertugas dalam mengurus tampilan sekaligus manajemen data pada aplikasi sehingga tidak dibutuhkan komunikasi API dengan aplikasi lain.



Gambar 3.1 Arsitektur Monolithic dan Decoupled

### 3.8. cPanel

Untuk melakukan *deployment* aplikasi supaya dapat diakses melalui internet, diperlukan penyewaan *web server*. Aplikasi SIMKEU sebelumnya sudah di-*host* pada Niagahoster. Layanan Niagahoster menggunakan cPanel sebagai GUI untuk berinteraksi dengan servernya.

### 3.9. Vercel

Salah satu layanan *web deployment* yang mudah digunakan untuk aplikasi berbasis React atau Nexxt.js adalah Vercel. Dengan beberapa tahap saja, *website* bisa langsung ter-*deploy*. Layanan Vercel ini dapat diperoleh secara gratis apabila memiliki akun GitHub Pro.

## **BAB IV**

### **ANALISIS DAN PERANCANGAN INFRASTRUKTUR SISTEM**

#### **4.1. Analisis Sistem**

Pada bab ini akan dijelaskan mengenai analisis sistem informasi keuangan (SIMKEU). Hal tersebut dijelaskan ke dalam dua bagian, definisi umum aplikasi dan analisis kebutuhan.

##### **4.1.1. Definisi Umum Aplikasi**

Secara umum, aplikasi SIMKEU merupakan sistem informasi keuangan berbasis *website* yang berfungsi untuk memberikan informasi seputar keuangan beserta alurnya yang terjadi dalam DRPM. Sistem ini memiliki tiga jenis pengguna. Yakni *super admin*, *admin* dan *user*.

##### **4.1.2. Analisis Kebutuhan**

Berdasarkan riset pengguna, DRPM membutuhkan suatu sistem informasi yang dapat mengakomodasi kebutuhan berikut.

1. Beranda: tampilan informasi tentang alokasi pagu pada tiap unit dan *tracking* tahap jadwal pada kegiatan
2. Perencanaan: pembuatan MAK, kegiatan beserta RBA
3. Transaksi: pembuatan struk transaksi pemasukan, pengeluaran dan pengembalian
4. Monitoring: penyimpanan serta perhitungan dari serapan MAK per unit, *cash opname* serta rekapitulasi semua transaksi
5. Kelola: pengelolaan unit dan juga akun pengguna

## 4.2. Perancangan Infrastruktur Sistem

Aplikasi SIMKEU dikembangkan dengan menggunakan basis *decoupled architecture*. Hal ini berarti SIMKEU memiliki bagian *front-end* yang mengurus perihal tampilan dan juga bagian *back-end* yang berisi sekumpulan logika yang bisa diakses melalui API. Arsitektur ini berbeda dari arsitektur aplikasi SIMKEU sebelumnya yang menggunakan arsitektur *monolithic*, yang mana SIMKEU hanya dibuat dari Laravel saja. Pada kerja praktik ini, aplikasi SIMKEU dirombak sehingga Laravel tidak lagi membuka *route* pada *web.php* dan seluruh layanan di-*refactor* menjadi layanan API. Sementara itu, untuk bagian *front-end*, digunakan *framework* Next.js untuk pengembangannya.

Pertimbangan ini diambil karena arsitektur *decoupled* yang menjadikan bagian *front-end* dan *back-end* tidak tergantung dengan satu sama lain lebih mudah untuk dikembangkan, dites dan juga di-*maintain*. Selain itu, pertimbangan ini juga diambil berdasarkan kemampuan penulis dalam mengembangkan *website* menggunakan *decoupled architecture*.

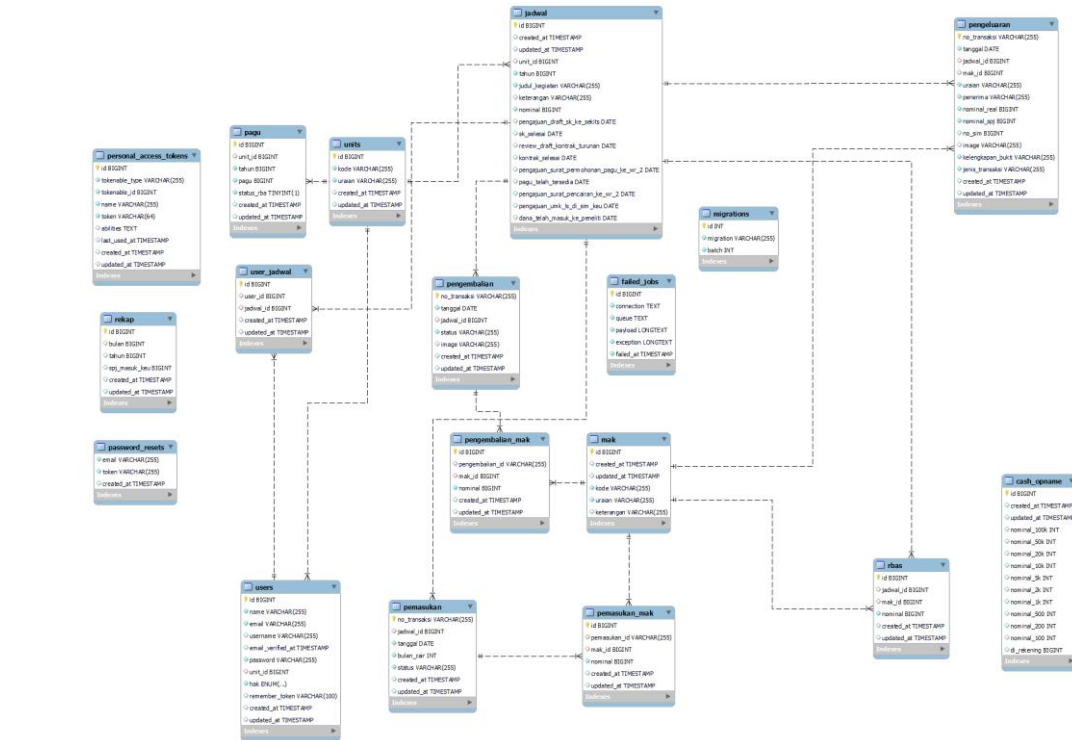
## **BAB V**

### **IMPLEMENTASI SISTEM**

#### **5.1. Implementasi Database**

Berdasarkan analisis kebutuhan yang didapatkan dari elisitasi dengan pihak DRPM, terdapat beberapa objek yang dapat dijadikan tabel yang pula memiliki hubungan dengan satu sama lain. Selain itu, pertimbangan ini juga diambil dari hasil analisis SIMKEU versi sebelumnya. Berikut ini adalah CDM dari aplikasi SIMKEU yang dikembangkan dalam kerja praktik ini.



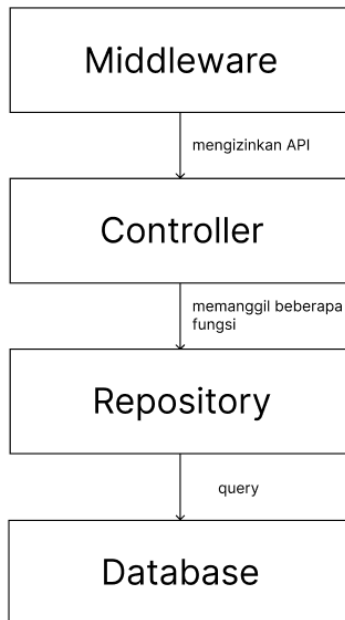


Gambar 5.1 Diagram CDM SIMKEU

## 5.2. Implementasi Back-End dengan Laravel

Sesuai dengan pembahasan sebelumnya, aplikasi Laravel yang dikembangkan dalam SIMKEU merupakan sekumpulan layanan API. Untuk dapat mengakses API Laravel ini, digunakan *middleware* otentifikasi menggunakan Laravel Sanctum. Dengan *middleware* ini, pengakses API memerlukan token ketika melakukan *request* REST API.

Untuk melakukan optimasi *request* REST API yang dilakukan oleh *front-end*, dibuat *layer* baru pada implementasi API di Laravel bernama *repository*. *Layer repository* dalam kasus ini bertindak sebagai penyedia fungsi *query* tabel pada *database* yang diperlukan.



Gambar 5.2 Diagram Layer pada Back-End

Tidak seperti layanan API Laravel biasanya yang sepenuhnya diatur logikanya oleh *controller*, *controller* pada *back-end* SIMKEU akan mengirimkan hasil dari beberapa *query* database menggunakan fungsi-fungsi yang ada pada *repository* yang dibutuhkan dalam satu laman *web*. Maka dari itu, *route* API yang dibuat cenderung dipanggil berdasarkan nama dari laman *web* di bagian *front-end*.

```
Route::middleware('auth:sanctum')->group(function () {
    Route::post('/logout', 'AuthController@logout');

    Route::get('/beranda', 'BerandaController@index');

    Route::get('/mak', 'MakController@index');
    Route::get('/mak/{id}', 'MakController@show');
    Route::post('/mak', 'MakController@create');
    Route::put('/mak', 'MakController@update');
    Route::delete('/mak/{id}', 'MakController@delete');

    Route::get('/kegiatan', 'KegiatanController@index');
    Route::get('/kegiatan/create', 'KegiatanController@indexCreateJadwal');
    Route::get('/kegiatan/create/rba/{unit_id}', 'KegiatanController@indexCreateRBA');
```

Gambar 5.3 Route API

Akan tetapi, dibuat juga beberapa layanan API yang menembak langsung ke *repository* yang dibuat dengan pertimbangan kemudahan ketika melakukan *testing* atau implementasi suatu laman saat pengembangan. Dalam kasus ini, penamaan *route* API disesuaikan dengan nama tabel dari *request* yang diinginkan.

```
// call directly to the repositories
Route::middleware('auth:sanctum')->group(function () {
    Route::get('/user', [UserRepository::class, 'getUserByToken']);
    Route::get('/user_all', [UserRepository::class, 'getUser']);
    Route::get('/user/{id}', [UserRepository::class, 'getUserById']);
    Route::get('/user/unit/{id}', [UserRepository::class, 'getUserByUnitId']);
    Route::post('/user', [UserRepository::class, 'createUser']);
    Route::put('/user/{id}', [UserRepository::class, 'updateUser']);
    Route::delete('/user/{id}', [UserRepository::class, 'deleteUser']);

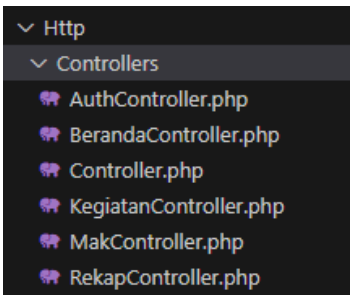
    Route::get('/unit', [UnitRepository::class, 'getUnit']);
    Route::get('/unit/{id}', [UnitRepository::class, 'getUnitById']);
    Route::post('/unit/search', [UnitRepository::class, 'getUnitByName']);
    Route::post('/unit', [UnitRepository::class, 'createUnit']);
    Route::put('/unit', [UnitRepository::class, 'updateUnit']);
    Route::delete('/unit/{id}', [UnitRepository::class, 'deleteUnit']);

    Route::get('/pagu', [PaguRepository::class, 'getPagu']);
    Route::get('/pagu/{unit_id}', [PaguRepository::class, 'getPaguByUnitId']);
    Route::post('/pagu', [PaguRepository::class, 'createPagu']);
    Route::put('/pagu', [PaguRepository::class, 'updatePagu']);
    Route::delete('/pagu/{id}', [PaguRepository::class, 'deletePagu']);

    Route::get('/jadwal', [JadwalRepository::class, 'getJadwal']);
    Route::get('/jadwal/{id}', [JadwalRepository::class, 'getJadwalById']);
    Route::post('/jadwal/search', [JadwalRepository::class, 'getJadwalByParameter']);
    Route::post('/jadwal', [JadwalRepository::class, 'createJadwal']);
    Route::post('/jadwal/unit_tahun', [JadwalRepository::class, 'getPaguSerapanByUnitTahun']);
    Route::put('/jadwal', [JadwalRepository::class, 'updateJadwal']);
    Route::delete('/jadwal/{id}', [JadwalRepository::class, 'deleteJadwal']);
});
```

Gambar 5.4 Route API Repository

Seperti aplikasi Laravel pada umumnya, setiap *class* pada *controller* merupakan *extend* dari *Controller.php* yang juga merupakan *extend* dari *BaseController*, yakni fitur *routing* yang sudah tersedia dalam Laravel.

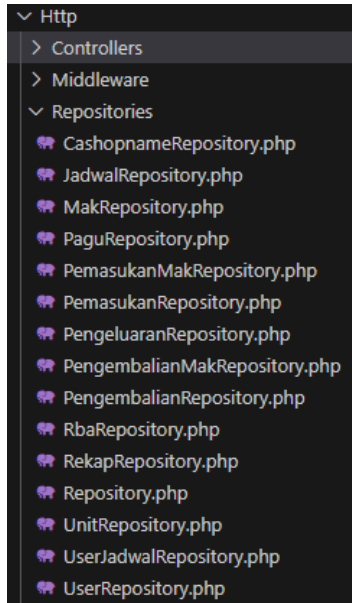


Gambar 5.5 Struktur Folder Controllers

```
backend-laravel > app > Http > Controllers > Controller.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Foundation\Auth\Access\AuthorizesRequests;
6  use Illuminate\Foundation\Bus\DispatchesJobs;
7  use Illuminate\Foundation\Validation\ValidatesRequests;
8  use Illuminate\Routing\Controller as BaseController;
9
10 class Controller extends BaseController
11 {
12     use AuthorizesRequests, DispatchesJobs, ValidatesRequests;
13 }
```

Gambar 5.6 Source Code Controller.php

Meniru *desain* implementasi yang sama dengan *controller*, setiap *class* yang ada pada segmen *repository* juga merupakan *extend* dari *Repository.php*. Dalam kasus ini, *Repository.php* dibuat sama persis seperti *Controller.php* supaya *repository* dapat melakukan *request query* ke MySQL.



*Gambar 5.7 Struktur Folder Repositories*

```
backend-laravel > app > Http > Repositories > Repository.php
1  <?php
2
3  namespace App\Http\Repositories;
4
5  use Illuminate\Foundation\Auth\Access\AuthorizesRequests;
6  use Illuminate\Foundation\Bus\DispatchesJobs;
7  use Illuminate\Foundation\Validation\ValidatesRequests;
8  use Illuminate\Routing\Controller as BaseController;
9
10 class Repository extends BaseController
11 {
12     use AuthorizesRequests, DispatchesJobs, ValidatesRequests;
13 }
```

*Gambar 5.8 Source Code Repository.php*

### 5.3. Implementasi Front-End dengan Next.js

Alur utama logika tiap laman *front-end* pada Next.js di SIMKEU mengutilisasikan fitur *hook* yang tersedia pada *library* React. Setiap kali laman pertama kali diakses, akan dipanggil `useEffect` dengan variable kosong yang akan memanggil fungsi `initializeApp()`. Fungsi ini adalah fungsi yang akan melakukan pemanggilan API.

```
useEffect(() => {
  initializeApp();
}, [])

const initializeApp = async () => {
  const res1 = await getUnit();
  await processData(res1);

  const res2 = await getPagu();
  if (res2[0].length === 0) setPagus(res2);
  else setPagus(res2[0]);

  const res6 = await getUser();
  setCurrentUser(res6.user);
}
```

Gambar 5.9 Source Code Fungsi Inisialisasi

Ada beberapa laman yang mengharuskan mereka untuk juga memanggil `initializeTahun()` karena terdapat *interface* pilihan (*dropdown*) tahun pada lamannya. Kemudian, apabila semua API *request* telah diterima, akan di-*trigger* `useEffect` lain yang memastikan bahwa data-data telah diterima dengan baik. Fungsi dalam `useEffect` ini kemudian akan menjalankan fungsi atau logika yang memproses data-data dari API sesuai kebutuhan pada inisiasi laman (apabila ada). Beberapa laman akan memiliki penggunaan `useEffect` yang lebih banyak dari

laman lainnya. Hal ini berarti laman tersebut butuh memproses data yang sebelumnya telah diproses.

```
useEffect(() => {
  const fetchData = async () => {
    if (allJadwal.length !== 0 && units.length !== 0 && users.length !== 0 && isLoading && currentUser !== null) {
      setSelectedUnit(units[0].id);
      if (currentUser?.hak !== "user")
        setSelectedUser("semua");
      else
        setSelectedUser(users[0].id);
      setSelectedYear(new Date().getFullYear());
    }
  };

  fetchData();
}, [allJadwal, users, units, currentUser]);

useEffect(() => {
  const fetchData = async () => {
    if (selectedUnit !== '' && selectedUser !== '' && selectedYear !== 0 && isLoading) {
      await filter(selectedUnit, selectedYear, selectedUser);
    }
  };

  fetchData();
}, [selectedUnit, selectedUser, selectedYear])
```

*Gambar 5.10 Penggunaan useEffect untuk Pemrosesan Data*



*[Halaman ini sengaja dikosongkan]*

## **BAB VI**

### **PENGUJIAN DAN EVALUASI**

#### **6.1. Tujuan Pengujian**

Pengujian dilakukan terhadap SIMKEU untuk memastikan fungsionalitas dan kesesuaian hasil implementasi aplikasi dengan analisis kebutuhan yang telah dilakukan terhadap pihak DRPM.

#### **6.2. Kriteria Pengujian**

Penilaian dari pengujian akan dilaksanakan dengan memperhatikan ketercapaian dan kapabilitas fungsionalitas-fungsionalitas yang telah diimplementasikan berdasarkan rancangan dan harapan yang telah digagas sebelumnya.

- a. Beranda
  - 1) Menampilkan tahap kegiatan sesuai dengan pilihan kegiatan yang diinginkan
  - 2) Menunjukkan diagram alokasi pagu setiap unit per tahun
- b. Kegiatan
  - 1) Membuat, mengedit atau menghapus kegiatan
  - 2) Melihat detil kegiatan beserta RBA-nya
  - 3) Membuat RBA dalam kegiatan sesuai alokasi pagu yang tersisa pada unit dan tahun tersebut
  - 4) Kemampuan untuk filter pada tampilan daftar kegiatan
- c. MAK
  - 1) Kemampuan aplikasi dalam membuat, mengedit atau menghapus MAK
  - 2) Mencari MAK
- d. Transaksi Pemasukan

- 1) Membuat, mengedit atau menghapus transaksi pemasukan sesuai RBA dari MAK yang telah dibuat
  - 2) Melihat detail transaksi pemasukan
  - 3) Mengubah status “Draft” menjadi “Diajukan”
  - 4) Mengubah status “Diajukan” menjadi “Disetujui” atau “Ditolak”
  - 5) Kemampuan untuk filter pada tampilan daftar kegiatan
- e. Transaksi Pengeluaran
- 1) Membuat, mengedit atau menghapus transaksi pengeluaran
  - 2) Melihat detail transaksi pemasukan
  - 3) Kemampuan untuk filter pada tampilan daftar kegiatan
- f. Transaksi Pengembalian
- 1) Membuat, mengedit atau menghapus transaksi pengembalian sesuai RBA dari MAK yang telah dibuat
  - 2) Melihat detail transaksi pengembalian
  - 3) Mengubah status “Draft” menjadi “Diajukan”
  - 4) Mengubah status “Diajukan” menjadi “Disetujui” atau “Ditolak”
  - 5) Kemampuan untuk filter pada tampilan daftar kegiatan
- g. Serapan
- 1) Mencari MAK
  - 2) Melihat alokasi MAK setiap unit per tahun dengan nominal yang sesuai dengan data yang ada
- h. Cash Opname
- 1) Mengubah dan menyimpan cash opname per unit
  - 2) Melihat rekap perhitungan cash opname dengan hasil yg sesuai/akurat
- i. Rekapitulasi

- 1) Melihat rekap total transaksi setiap unit per bulan dan per tahun
  - 2) Kemampuan mengubah “SPJ Masuk KEU” dan mendapatkan keterangan yang sesuai dengan data
- j. Kelola Akun
- 1) Mencari akun
  - 2) Mengubah *password* akun sendiri
  - 3) Melihat akun dalam unit yang sama
  - 4) Membuat dan menghapus akun
  - 5) Mengubah *password* akun orang lain
  - 6) Mengedit akun (selain *password*)
- k. Kelola Unit
- 1) Mencari unit
  - 2) Melihat unit selain diri sendiri
  - 3) Membuat, mengedit dan menghapus unit
  - 4) Kemampuan menambahkan pagu untuk suatu tahun pada suatu unit

### 6.3. Skenario Pengujian

Karena setiap fungsionalitas yang telah dijabarkan pada 6.2 hanya dapat dilakukan oleh akun dengan peran tertentu yang dapat berbeda untuk tiap fungsionalitas yang ada, pengujian akan dilakukan dengan mengetes apakah kotak putih yang dalam tabel 6.1 (yang simbolnya sesuai dengan fungsionalitas pada 6.2) terpenuhi sehingga dapat ditambahkan tanda “✓”.

Tabel 6.1 Matriks Pengujian

| Laman | Poin Fungsi | Peran       |       |      |
|-------|-------------|-------------|-------|------|
|       |             | Super Admin | Admin | User |
| a     | 1           |             |       |      |
|       | 2           |             |       |      |
| b     | 1           |             |       |      |

|   |   |  |  |  |
|---|---|--|--|--|
|   | 2 |  |  |  |
|   | 3 |  |  |  |
|   | 4 |  |  |  |
| c | 1 |  |  |  |
|   | 2 |  |  |  |
| d | 1 |  |  |  |
|   | 2 |  |  |  |
|   | 3 |  |  |  |
|   | 4 |  |  |  |
|   | 5 |  |  |  |
| e | 1 |  |  |  |
|   | 2 |  |  |  |
|   | 3 |  |  |  |
| f | 1 |  |  |  |
|   | 2 |  |  |  |
|   | 3 |  |  |  |
|   | 4 |  |  |  |
|   | 5 |  |  |  |
| g | 1 |  |  |  |
|   | 2 |  |  |  |
| h | 1 |  |  |  |
|   | 2 |  |  |  |
| i | 1 |  |  |  |
|   | 2 |  |  |  |
| j | 1 |  |  |  |
|   | 2 |  |  |  |
|   | 3 |  |  |  |
|   | 4 |  |  |  |
|   | 5 |  |  |  |
|   | 6 |  |  |  |
| k | 1 |  |  |  |
|   | 2 |  |  |  |
|   | 3 |  |  |  |
|   | 4 |  |  |  |

## 6.1. Evaluasi Pengujian

Hasil pengujian dijabarkan pada matriks pengujian yang telah dibahas pada 6.3.

*Tabel 6.2 Matriks Pengujian (Hasil)*

| Laman | Poin Fungsi | Peran       |       |      |
|-------|-------------|-------------|-------|------|
|       |             | Super Admin | Admin | User |
| a     | 1           | ✓           | ✓     | ✓    |
|       | 2           | ✓           | ✓     | ✓    |
| b     | 1           | ✓           | ✓     |      |
|       | 2           | ✓           | ✓     | ✓    |
|       | 3           | ✓           | ✓     |      |
|       | 4           | ✓           | ✓     | ✓    |
| c     | 1           | ✓           |       |      |
|       | 2           | ✓           | ✓     | ✓    |
| d     | 1           | ✓           | ✓     |      |
|       | 2           | ✓           | ✓     | ✓    |
|       | 3           | ✓           | ✓     |      |
|       | 4           |             |       | ✓    |
|       | 5           | ✓           | ✓     | ✓    |
| e     | 1           | ✓           | ✓     |      |
|       | 2           | ✓           | ✓     | ✓    |
|       | 3           | ✓           | ✓     | ✓    |
| f     | 1           | ✓           | ✓     |      |
|       | 2           | ✓           | ✓     | ✓    |
|       | 3           | ✓           | ✓     |      |
|       | 4           |             |       | ✓    |
|       | 5           | ✓           | ✓     | ✓    |
| g     | 1           | ✓           | ✓     | ✓    |

|   |   |   |   |   |
|---|---|---|---|---|
|   | 2 | ✓ | ✓ | ✓ |
| h | 1 | ✓ | ✓ |   |
|   | 2 | ✓ | ✓ | ✓ |
| i | 1 | ✓ | ✓ | ✓ |
|   | 2 | ✓ | ✓ |   |
| j | 1 | ✓ | ✓ | ✓ |
|   | 2 | ✓ | ✓ | ✓ |
|   | 3 | ✓ | ✓ | ✓ |
|   | 4 | ✓ | ✓ |   |
|   | 5 | ✓ |   |   |
|   | 6 | ✓ |   |   |
| k | 1 | ✓ | ✓ | ✓ |
|   | 2 | ✓ |   |   |
|   | 3 | ✓ |   |   |
|   | 4 | ✓ |   |   |

## **BAB VII**

### **KESIMPULAN DAN SARAN**

#### **7.1. Kesimpulan**

Kesimpulan yang didapat setelah melakukan perancangan, implementasi dan pengujian dari SIMKEU pada kerja praktik ini adalah sebagai berikut.

- a. Rancangan proses bisnis dari administrasi keuangan DRPM pada SIMKEU dimulai dari pembuatan MAK. Setelah itu, dibutuhkan unit beserta pagu per tahun. Dari ketiga hal tersebut, bisa dibuat kegiatan dari unit tersebut beserta RBA berdasarkan MAK yang sudah dibuat. Dari RBA-RBA berbagai kegiatan yang ada, bisa dibuat transaksi pemasukan. Lalu, dari transaksi pemasukan yang diterima oleh unit yang bersangkutan, unit tersebut dapat membuat transaksi pengeluaran. Setelah itu, apabila masih tersisa uang pada unit tersebut, unit tersebut dapat membuat transaksi pengembalian. Dari alur tersebut, bisa dilakukan *monitoring* pada segmen *monitoring* di SMIKEU.
- b. Arsitektur dari SIMKEU adalah *decoupled architecture*. Sementara itu, *tech stack* yang digunakan dalam SIMKEU adalah Laravel untuk *back-end* dan Next.js untuk *front-end*
- c. Berdasarkan evaluasi, SIMKEU dapat melayani kebutuhan administrasi keuangan DRPM ITS berdasarkan analisis kebutuhan yang sudah dilakukan.

#### **7.2. Saran**

Saran untuk perancangan pengembangan SIMKEU kedepannya dijabarkan sebagai berikut.



- a. Pengimplementasian *upload* gambar dalam transaksi pengeluaran.
- b. Pengimplementasian fitur *generate* rekap dengan format *file* .csv untuk perihal percetakan (*print*) atau pengarsipan kantor.

*[Halaman ini sengaja dikosongkan]*

## DAFTAR PUSTAKA

- Sulistiani, H., Nuriansah, A., & Wahyuni, E. D. (2022).  
Pengembangan Sistem Informasi Perhitungan Upah Lembur  
Karyawan Berbasis Web Pada PT Sugar Labinta. *Jurnal  
Ilmiah Sistem Informasi Akuntansi*, 2(2), 69–76.  
<https://doi.org/10.33365/jimasia.v2i2.2015>

*[Halaman ini sengaja dikosongkan]*

## **BIODATA PENULIS**

Nama : Zahra Fayyadiyah  
Tempat, Tanggal Lahir : Bandung, 25 Juli 2002  
Jenis Kelamin : Perempuan  
Telepon : +6281222553205  
Email : zahrafayyadiyah.ac@gmail.com

## **AKADEMIS**

Kuliah : Departemen Teknik Informatika –  
FTEIC , ITS  
Angkatan : 2020  
Semester : 7 (Tujuh)