HALAMAN PENGESAHAN

SISTEM LAPORAN KEUANGAN UMKM

SULAIMAN
NIM. 219 280 122

Telah dipertahankan di depan Komisi Penguji Ujian Skripsi pada tanggal 20 Agustus 2025 dan dinyatakan telah memenuhi syarat

Komisi Penguji

Ahmad Selao STP.,M.Sc. (Ketua) (.....)

Ir.Untung Suwardoyo,S.Kom,MT.,IPP ((Sekretaris) (.....)

Andi Wafiah, S.Kom.,M.Kom. (Anggota) (.....)

Sudirman Sahidin,S.T.,M.Kom. (Anggota) (.....)

Mengetahui,

Plt.Ketua Program Studi
Teknik Informatika

Mughaffir Yunus, ST., MT
NBM. 1252 809

Dekan
Fakultas Teknik

Dr. Ir. H. Hakkah, S.T., M.T
NBM. 938 317

PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertandatangan di bawah ini :

Nama	:	**Sulaiman**
NIM	:	**219280122**
Program Studi	:	Teknik Informatika
Fakultas	:	Teknik Universitas Muhammadiyah Parepare
Judul Skripsi	:	Sistem Laporan Keuangan UMKM

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambilalihan tulisan atau pemikiran orang lain. Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan skripsi ini hasil karya orang lain, saya bersedia menerima sanksi sesuai aturan yang berlaku.

Parepare,20 Agustus 2025

Yang menyatakan



SULAIMAN

NIM.219280122

Dipindai dengan CamScanner

PRAKATA



Alhamdulillahi robbil 'alamiin. Puji syukur kehadirat Allah *subhanahu wa ta'ala* atas limpahan rahmat-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul "**Sistem Laporan Keuangan Umkm**" Dengan nama Allah Yang Maha Pengasih lagi Maha Penyayang, penulis mengucapkan terima kasih kepada-Nya atas rahmat, hidayah, dan karunia-Nya yang telah membantu penulis menyelesaikan skripsi ini dengan baik. Penulisan skripsi ini adalah perjalanan yang panjang dan sulit. Penulis telah banyak mendapatkan bantuan, dukungan, dan dorongan selama proses penulisan ini. Oleh karena itu, penulis ingin mengucapkan terima kasih yang tulus kepada:

1. Keluarga terutama orang tua saya, bapak Burhan dan ibu Erni terima kasih atas cinta, dukungan, dan pengorbanan tanpa batas yang telah kalian berikan. Kalian adalah teladan yang sempurna bagi hidupku. Saya beruntung memiliki kalian sebagai orang tua. Terima kasih atas segalanya, doa, dukungan, dan inspirasi yang kalian berikan selama saya mengerjakan skripsi ini.
2. Bapak Dr. H. Hakzah, ST.,MT, selaku Dekan Fakultas Teknik Universitas Muhammadiyah Parepare.
3. Bapak Mughaffir Yunus,S.T.,M.T, selaku ketua program studi teknik informatika.
4. Bapak Ahmad Selao, S.T.P.,M.Sc., selaku pembimbing I.
5. Bapak Ir. Untung Suwardoyo, S.Kom.,MT.,IPP selaku pembimbing II.
6. Ibu Andi Wafiah,S.Kom.,M.Kom, selaku penguji I.

7. Bapak Sudirman Sahidin,S.T.,M.Kom, selaku penguji I.
8. Seluruh Dosen Prodi Teknik Informatika dan Staf Fakultas Teknik Universitas Muhammadiyah Parepare
9. Teman-teman yang selalu setia memberikan dukungan dan doa selama proses penyelesaian skripsi ini.

Penulis sadar sepenuhnya bahwa skripsi ini tidak sempurna. Oleh karena itu, penulis memohon maaf yang sebesar-besarnya apabila ada kesalahan atau kekurangan yang mungkin terjadi selama proses penyusunan Akhir kata, penulis berharap semoga Allah SWT senantiasa memberikan rahmat dan hidayah-Nya kepada kita semua. Aamiin.

Parepare, 20 Agustus 2025

Penulis

Sulaiman
NIM.219280122

ABSTRAK

Sulaiman. *Sistem Laporan Keuangan Umkm* (dibimbing oleh Ahmad Selao dan Untung Suwardoyo).

Sistem pencatatan manual yang masih digunakan oleh sebagian besar pelaku Usaha Mikro, Kecil, dan Menengah (UMKM) sering kali menimbulkan berbagai permasalahan, seperti kehilangan data, kesalahan perhitungan, serta ketidakakuratan dalam menyusun laporan keuangan. Kondisi ini dapat berdampak buruk terhadap pengambilan keputusan bisnis, terutama ketika UMKM membutuhkan pembiayaan atau ingin menarik investor. Penelitian ini bertujuan untuk merancang sistem laporan keuangan berbasis web yang dapat mempermudah pelaku UMKM dalam mencatat dan mengelola transaksi keuangan secara lebih efektif. Metode yang digunakan dalam penelitian ini adalah pengembangan sistem berbasis Progressive Web Application (PWA) dengan menggunakan bahasa pemrograman PHP dan database MySQL. Hasil penelitian menunjukkan bahwa sistem yang dibangun telah berhasil memenuhi kebutuhan dasar pelaporan keuangan UMKM, dengan fitur utama seperti jurnal umum, jurnal penyesuaian, buku besar, neraca saldo, neraca lajur, dan laporan keuangan otomatis. Sistem ini dapat diakses kapan saja melalui internet dan membantu pelaku UMKM menyusun laporan keuangan yang lebih rapi, terstruktur, dan akurat. Secara keseluruhan, sistem ini diharapkan mampu menjadi solusi praktis dalam meningkatkan kualitas administrasi keuangan UMKM dan mendorong digitalisasi usaha kecil di era teknologi saat ini.

Kata Kunci: Sistem, Laporan Keuangan, UMKM, Progressive Web Application.

ABSTRACT

Sulaiman. *MSME Financial Reporting System (supervised by Ahmad Selao and Untung Suwardoyo).*

The manual recording system still used by most Micro, Small, and Medium Enterprises (MSMEs) often leads to various problems, such as data loss, calculation errors, and inaccuracies in preparing financial reports. This situation can negatively impact business decision-making, especially when MSMEs require financing or want to attract investors. This research aims to design a web-based financial reporting system that can facilitate MSMEs in recording and managing financial transactions more effectively. The method used in this research is the development of a Progressive Web Application (PWA)-based system using the PHP programming language and a MySQL database. The results show that the system successfully meets the basic financial reporting needs of MSMEs, with key features such as general journals, adjustment journals, general ledgers, trial balances, trial balances, and automated financial reports. This system can be accessed anytime via the internet and helps MSMEs prepare more organized, structured, and accurate financial reports. Overall, this system is expected to be a practical solution for improving the quality of MSME financial administration and encouraging the digitalization of small businesses in today's technological era.

Keywords: System, Financial Reports, MSMEs, Progressive Web Application.

DAFTAR ISI

HALAMAN JUDUL i
HALAMAN PENGESAHAN ii
HALAMAN PERNYATAAN KEASLIAN SKRIPSI iii
HALAMAN MOTTO iv
PRAKATA v
ABSTRAK vii
ABSTRACT viii
DAFTAR GAMBAR xi
BAB I PENDAHULUAN 1
A. Latar Belakang 1
B. Rumusan Masalah 2
C. Tujuan Penelitian 3
D. Batasan Masalah 3
E. Manfaat Penelitian 3
F. Sistematika Penulisan 3
BAB II TINJAUAN PUSTAKA 5
A. Kajian Teori 5
1. Usaha Mikro Kecil dan Menengah UMKM 5
2. Laporan Keuangan 8
3. *Progressive Web Application* (PWA) 15
4. Sistem Informasi 18
5. *Website* 19
6. Pemrograman Web 20
7. PHP (*Hypertext Preprocessor*) 21
8. HTML 22
9. MySQL 24
10. *Flowchart* 26
11. Metode Pengembangan Perangkat Lunak 28
12. Metode Pengujian 37

B. Kajian Hasil Penelitian Terdahulu 44
C. Kerangka Pikir 47
BAB III METODE PENELITIAN 55
A. Jenis Penelitian 55
B. Waktu dan Tempat Penelitian 55
C. Alat dan Bahan Penelitian 56
D. Tahap Penelitian 57
E. Analisis Sistem 58
1. *Flowchart* Aplikasi 58
2. Sistem Yang Berjalan 59
3. Sistem Yang Diusulkan 60
BAB IV HASIL DAN PEMBAHASAN 62
A. Hasil Penelitian 62
B. Pembahasan 70
1. Analisis Aliran Data UML 70
2. *Activity Diagram* 75
3. *Sequence Diagram* 82
4. Kamus Data 89
5. Perancangan Sistem 91
6. Pengujian Sistem 125
BAB V PENUTUP 107
A. Kesimpulan 107
B. Saran 108
DAFTAR PUSTAKA 102

DAFTAR TABEL

Tabel 2.1 Simbol *Usecase Diagram* 23

Tabel 2.2 Simbol *Class Diagram* 24

Tabel 2.3 Simbol *Activity Diagram* 25

Tabel 2.4 Simbol *Sequence Diagram* 26

Tabel 2.5 Simbol *StateChart Diagram* 27

Tabel 2.6 Simbol *Flowchart* 29

Tabel 2.7 *White Box* Perhituangan 33

Tabel 2.8 Contoh Tabel Pengujian *Black Box* 36

Tabel 3.1 Jadwal Penelitian 40

Tabel 3.2 Definisi Sistem yang Berjalan 53

Tabel 3.3 Definisi Aktor dalam Sistem 54

Tabel 3.4 Definisi Sistem yang diusulkan 55

DAFTAR GAMBAR

Gambar 2 1 Logo Java 14

Gambar 2 2 Contoh Coding Bahasa Pemrograman Java 14

Gambar 2 3 Logo *XML* 15

Gambar 2 4 *SQLite* 15

Gambar 2 5 Logo *Kotlin* 16

Gambar 2 6 Contoh Coding *Kotlin* 17

Gambar 2 7 Logo *MySQL* 18

Gambar 2 8 Contoh *Flowchart* 19

Gambar 2 9 *Prototype* Paradigma 20

Gambar 2 1 0 *White Box Testing* 30

Gambar 2 11 Contoh *White box* 32

Gambar 2 12 *Black Box Testing* 33

Gambar 2. 13 Contoh *Black Box* 33

Gambar 2 14 Kerangka Pikir 39

Gambar 2. 15 Kerangka Pikir 39

Gambar 3 1 Sistem Yang Berjalan 43

Gambar 3 2 Sistem Yang Dusulkan 44

LAMPIRAN

Lampiran-1 Login

```
<script setup lang="ts">
import InputError from
'@/components/InputError.vue';
import TextLink from
'@/components/TextLink.vue';
import { Button } from
'@/components/ui/button';
import { Checkbox } from
'@/components/ui/checkbox';
import { Input } from
'@/components/ui/input';
import { Label } from
'@/components/ui/label';
import AuthBase from
'@/layouts/AuthLayout.vue';
import { Head, useForm } from
'@inertiajs/vue3';
import { LoaderCircle } from
'lucide-vue-next';

defineProps<{
    status?: string;
    canResetPassword: boolean;
}>();

const form = useForm({
    email: '',
    password: '',
    remember: false,
});

const submit = () => {
    form.post(route('login'),
{
        onFinish: () => {
            form.password =
'';

console.log('oke');
        },
    });
};
</script>

<template>
    <AuthBase title="Masuk ke
Akun Anda"
description="Masukkan email
dan kata sandi Anda di bawah
ini untuk masuk">
        <Head title="Log in"
/>

        <div v-if="status"
class="mb-4 text-center text-
sm font-medium text-green-
600">
            {{ status }}
        </div>

        <form
@submit.prevent="submit"
class="flex flex-col gap-6">
            <div class="grid
gap-6">
                <div
class="grid gap-2">
                    <Label
for="email">Email</Label>
                    <Input

id="email"

type="email"

required

autofocus

:tabindex="1"

autocomplete="off"
                        v-
model="form.email"

placeholder="email@example.com
"
                    />

<InputError
:message="form.errors.email"
/>
                </div>

                <div
class="grid gap-2">
                    <div
class="flex items-center
justify-between">
                        <Label
for="password">Password</Label
>
                        <!--
<TextLink v-
if="canResetPassword"
:href="route('password.request
')" class="text-sm"
:tabindex="5">
```

```
Forgot password?

</TextLink> -->
                    </div>
                    <Input

id="password"

type="password"

required

:tabindex="2"

autocomplete="current-
password"
                           v-
model="form.password"

placeholder="Password"
                    />

<InputError
:message="form.errors.password
" />
                </div>

                <div
class="flex items-center
justify-between"
:tabindex="3">
                    <Label
for="remember" class="flex
items-center space-x-3">

<Checkbox id="remember" v-
model="form.remember"
:tabindex="4" />

<span>Remember me</span>
                    </Label>
                </div>

                <Button
type="submit" class="mt-4 w-
full" :tabindex="4"
:disabled="form.processing">

<LoaderCircle v-
if="form.processing" class="h-
4 w-4 animate-spin" />
                    Log in
                </Button>
                <div
class="text-muted-foreground
text-center text-sm">
                    Belum
punya akun?
                    <TextLink
:href="route('register')"
class="underline underline-
offset-4" :tabindex="6">Daftar
Sekarang</TextLink>
                </div>
            </div>
        </form>
    </AuthBase>
</template>

Lampiran-2 Home <script setup
lang="ts">
import { Card, CardContent }
from '@/components/ui/card';
import AppLayout from
'@/layouts/AppLayout.vue';
import { SharedData, type
BreadcrumbItem } from
'@/types';
import { Head, usePage } from
'@inertiajs/vue3';
import { StoreIcon } from
'lucide-vue-next';

const { data } = defineProps<{
data: { umkmCount: number }
}>();
const { auth } =
usePage<SharedData>().props;

const breadcrumbs:
BreadcrumbItem[] = [
    {
        title: 'Dashboard',
        href: '/dashboard',
    },
];
</script>

<template>
    <Head title="Dashboard" />

    <AppLayout
:breadcrumbs="breadcrumbs">
        <div class="flex h-
full flex-1 flex-col gap-4
rounded-xl p-4">
            <div v-
if="auth.roles?.includes('supe
r-admin')" class="grid auto-
```

```
rows-min gap-4 md:grid-cols-
3">
                <Card
class="rounded-xl bg-gray-100
p-6 shadow-md dark:bg-black
dark:text-white">

<CardContent class="flex
items-center justify-between">
                        <div>
                            <p
class="mb-1 text-sm text-gray-
400">UMKM</p>

<h2 class="text-3xl font-
bold">{{ data.umkmCount
}}</h2>

<!-- <p class="mt-1 text-sm
text-green-400">+180.1% from
last month</p> -->
                        </div>

<StoreIcon class="h-6 w-6
text-gray-400" />

</CardContent>
                </Card>
            </div>
            <div v-else>
                <h2
class="text-xl font-bold
tracking-wide md:text-
3xl">Selamat Datang Kembali di
Aplikasi Pengelolaan <br
/>Laporan Keuangan!</h2>

                <!-- cara
penggunanaan -->
                <div
class="mt-2 w-full max-w-3xl
text-sm tracking-wide text-
gray-500">
                    Aplikasi
ini digunakan untuk membantu
UMKM dalam mengelola laporan
keuangan mereka. Anda dapat
melakukan berbagai hal seperti
                    membuat
laporan keuangan. Aplikasi ini
juga dilengkapi dengan fitur
untuk mencetak laporan
keuangan dalam format PDF.
                    <span
class="text-red-500"
                        >Hanya
dengan membuat jurnal umum
data yang lain akan
tergenerate sesuai pedoman
pembuatan laporan
keuangan</span
                    >
                </div>
            </div>
        </div>
    </AppLayout>
</template>
```

Lampiran 3 – controller DashboardController.php

```
<?php

namespace
App\Http\Controllers;

use Illuminate\Http\Request;
use
Illuminate\Support\Facades\Aut
h;
use Inertia\Inertia;

class DashboardController
extends Controller
{
    /**
     * Handle the incoming
request.
     */
    public function
__invoke(Request $request)
    {
        // hitung jumlah umkm
        $user = Auth::user();
        $umkmCount = $user-
>hasRole('super-admin') ?
\App\Models\Umkm::count() : 0;

        return
Inertia::render('Dashboard', [
            'title' =>
'Dashboard',
            'description' =>
'Halaman untuk mengelola
dashboard',
            'data' =>
compact('umkmCount'),
        ]);
    }
}
```

JurnalUmumController.php

```
<?php

namespace
App\Http\Controllers;

use App\Models\Akun;
use App\Models\JurnalUmum;
use Illuminate\Http\Request;
use
Illuminate\Support\Facades\Aut
h;
use
Illuminate\Support\Facades\Cac
he;
use Inertia\Inertia;

class JurnalUmumController
extends Controller
{
    protected $month;
    public function
__construct()
    {
        $this->month =
request('month') ?? date('Y-
m');

        Inertia::share([
            'title' =>
'Jurnal Umum',
            'month' => $this-
>month,
        ]);
    }


    public function index()
    {
        $umkm_id =
Auth::user()->umkm_id;
        $data =
JurnalUmum::with('akun')-
>orderBy('tanggal')-
>orderBy('jurnal_id')-
>orderBy('debit', 'desc')
            ->where('umkm_id',
$umkm_id)
            ->where('tanggal',
'like', $this->month . '%')
            ->orderBy('id')-
>get();

        return
Inertia::render('JurnalUmum/In
dex', ['data' => $data]);
    }

    public function create()
    {

        return
Inertia::render(

'JurnalUmum/Create',
            [
                'akuns' =>
Akun::orderBy('kode_akun')-
>get()
            ]
        );
    }

    public function
store(Request $request)
    {


        $umkm_id =
Auth::user()->umkm_id;


        foreach ($request-
>jurnal as $data) {
            $kredit =
$data['kredit'] ?? [];
            $debit =
$data['debit'] ?? [];

            $JurnalUmum =
JurnalUmum::create([
                'kode_akun' =>
$debit['kode_akun'],
                'tanggal' =>
$data['tanggal'],
                'ref' =>
$debit['ref'] ?? null,
                'debit' =>
$debit['total'] ?? 0,
                'kredit' => 0,
                'umkm_id' =>
$umkm_id
            ]);

            $JurnalUmum-
>update([
                'jurnal_id' =>
$JurnalUmum->id,
            ]);
```

```
JurnalUmum::create([
                'tanggal' =>
$JurnalUmum->tanggal,
                'umkm_id' =>
$JurnalUmum->umkm_id,
                'kode_akun' =>
$kredit['kode_akun'],
                'debit' => 0,
                'kredit' =>
$kredit['total'] ?? 0,
                'jurnal_id' =>
$JurnalUmum->id,
            ]);
        }



        if ($request->action
=== 'save_and_create_another')
{
            return redirect()-
>back()->with($this-
>flash('Data berhasil
disimpan. Silakan tambah data
baru.'));
        }

        return redirect()-
>route('jurnal.index', [
            'month' => $this-
>month
        ])->with($this-
>flash('Data berhasil
disimpan.'));
    }

    public function edit($id)
    {

        $debit =
JurnalUmum::findOrFail($id);
        $kredit = $debit-
>jurnalChildren;
        $akuns =
Akun::orderBy('kode_akun')-
>get();

        return
Inertia::render('JurnalUmum/Ed
it', compact('debit',
'kredit', 'akuns'));
    }

    public function
update(Request $request, $id)
    {

        $jurnal = (object)
$request->jurnal;


        $model =
JurnalUmum::findOrFail($id);
        $model->update([
            'tanggal' =>
$jurnal->tanggal,
            'kode_akun' =>
$jurnal->debit['kode_akun'],
            'ref' => $jurnal-
>debit['ref'] ?? null,
            'debit' =>
$jurnal->debit['total'] ?? 0,
            'kredit' =>  0,
        ]);


JurnalUmum::where('jurnal_id',
$model->id)->whereNot('id',
$model->id)->update([
            'tanggal' =>
$jurnal->tanggal,
            'kode_akun' =>
$jurnal->kredit['kode_akun'],
            'debit' => 0,
            'kredit' =>
$jurnal->kredit['total'] ?? 0,
        ]);

        return redirect()-
>route('jurnal.index', [
            'month' => $this-
>month
        ])->with($this-
>flash('Data berhasil
diperbarui.'));
    }

    public function
destroy(Request $request, $id)
    {

JurnalUmum::where('id', $id)-
>orWhere('jurnal_id', $id)-
>delete();
        return redirect()-
>route('jurnal.index',
['month' => $request->month])-
>with($this->flash('Data
berhasil dihapus.'));
```

```
    }
}

/** end
JurnalUmumController.php */


/** start
JurnalPenyesuaianController */

<?php

namespace
App\Http\Controllers;

use App\Models\Akun;
use
App\Models\JurnalPenyesuaian;
use Illuminate\Http\Request;
use
Illuminate\Support\Facades\Aut
h;
use
Illuminate\Support\Facades\Cac
he;
use Inertia\Inertia;

class
JurnalPenyesuaianController
extends Controller
{

    protected $month;
    public function
__construct()
    {
        $this->month =
request('month') ?? date('Y-
m');
        Inertia::share([
            'title' =>
'Jurnal Penyesuaian',
            'month' => $this-
>month,
        ]);
    }


    public function index()
    {
        $umkm_id =
Auth::user()->umkm_id;
        $data =
JurnalPenyesuaian::with('akun'
)
            ->where('umkm_id',
$umkm_id)
            ->where('tanggal',
'like', $this->month . '%')
            -
>orderBy('tanggal')
            -
>orderBy('jurnal_id')
            ->orderBy('debit',
'desc')
            ->orderBy('id')-
>get();
        return
Inertia::render('JurnalPenyesu
aian/Index', ['data' =>
$data]);
    }

    public function create()
    {

        return
Inertia::render(

'JurnalPenyesuaian/Create',
            ['akuns' =>
Akun::orderBy('kode_akun')-
>get()]
        );
    }

    public function
store(Request $request)
    {


        $umkm_id =
Auth::user()->umkm_id;


        foreach ($request-
>jurnal as $data) {
            $kredit =
$data['kredit'] ?? [];
            $debit =
$data['debit'] ?? [];

            $JurnalPenyesuaian
= JurnalPenyesuaian::create([
                'kode_akun' =>
$debit['kode_akun'],
                'tanggal' =>
$data['tanggal'],
                'ref' =>
$debit['ref'] ?? null,
```

```
                'debit' =>
$debit['total'] ?? 0,
                'kredit' => 0,
                'umkm_id' =>
$umkm_id
            ]);


$JurnalPenyesuaian->update([
                'jurnal_id' =>
$JurnalPenyesuaian->id,
            ]);


JurnalPenyesuaian::create([
                'tanggal' =>
$JurnalPenyesuaian->tanggal,
                'umkm_id' =>
$JurnalPenyesuaian->umkm_id,
                'kode_akun' =>
$kredit['kode_akun'],
                'debit' => 0,
                'kredit' =>
$kredit['total'] ?? 0,
                'jurnal_id' =>
$JurnalPenyesuaian->id,
            ]);
        }



        if ($request->action
=== 'save_and_create_another')
{
            return redirect()-
>back()->with($this-
>flash('Data berhasil
disimpan. Silakan tambah data
baru.'));
        }

        return redirect()-
>route('jurnal-
penyesuaian.index', [
            'month' => $this-
>month
        ])->with($this-
>flash('Data berhasil
disimpan.'));
    }

    public function edit($id)
    {
        $debit =
JurnalPenyesuaian::findOrFail(
$id);
        $kredit = $debit-
>jurnalChildren;
        $akuns =
Akun::orderBy('kode_akun')-
>get();
        return
Inertia::render('JurnalPenyesu
aian/Edit', compact('debit',
'kredit', 'akuns'));
    }

    public function
update(Request $request, $id)
    {

        $jurnal = (object)
$request->jurnal;



        $model =
JurnalPenyesuaian::findOrFail(
$id);
        $model->update([
            'tanggal' =>
$jurnal->tanggal,
            'kode_akun' =>
$jurnal->debit['kode_akun'],
            'ref' => $jurnal-
>debit['ref'] ?? null,
            'debit' =>
$jurnal->debit['total'] ?? 0,
            'kredit' =>  0,
        ]);


JurnalPenyesuaian::where('jurn
al_id', $model->id)-
>whereNot('id', $model->id)-
>update([
            'tanggal' =>
$jurnal->tanggal,
            'kode_akun' =>
$jurnal->kredit['kode_akun'],
            'debit' => 0,
            'kredit' =>
$jurnal->kredit['total'] ?? 0,
        ]);

        return redirect()-
>route('jurnal-
penyesuaian.index', [
            'month' => $this-
>month
        ])->with($this-
>flash('Data di perbarui.'));
    }
```

```
    public function
destroy($id)
    {

JurnalPenyesuaian::where('id',
$id)->orWhere('jurnal_id',
$id)->delete();
        return redirect()-
>route('jurnal-
penyesuaian.index', [
            'month' => $this-
>month
        ])->with($this-
>flash('Data berhasil
dihapus.'));
    }
}


/** end
JurnalPenyesuaianController.ph
p */

Lampiran 4 – Kelola Pengguna
UMKM
<template>
    <AppLayout
:breadcrumbs="breadcrumbs">
        <Head title="Data
Pengguna UMKM" />
        <div class="w-full p-
4">
            <div class="mb-4
flex items-center justify-
between">
                <h1
class="text-2xl font-
bold">Data Pengguna UMKM</h1>
                <Link
:href="route('users.create')">
                    <Button>+
Tambah Data</Button>
                </Link>
            </div>

            <Table v-
if="data.length > 0">
                <TableHeader>
                    <TableRow>

<TableHead>Nama
Pengguna</TableHead>

<TableHead>Nama
UMKM</TableHead>

<TableHead>Email</TableHead>

<TableHead>Status</TableHead>

<TableHead>Aksi</TableHead>

</TableRow>
                </TableHeader>
                <TableBody>
                    <TableRow
v-for="item in data"
:key="item.id">

<TableCell>{{ item.name || '-'
}}</TableCell>

<TableCell>{{ item.umkm.name
}}</TableCell>

<TableCell>{{ item.email
}}</TableCell>

<TableCell>

<div class="flex items-center
space-x-2">

<Label>

<Switch

@update:model-value="(status)
=> handleChange(status,
item.id)"

:default-
value="item.umkm.status ==
'active'"

id="airplane-mode"

:disabled="isDisabled"

/>

<span :id="String(item.id)"
class="text-sm font-medium
text-gray-900">

{{ item.umkm.status ==
'active' ? 'Aktif' : 'Non-
Aktif' }}

</span>
```

```
</Label>

</div>

</TableCell>

<TableCell>

<div class="flex items-center
gap-2">

<Link
:href="route('users.edit',
item.id)">

<Button size="icon"><Pen
/></Button>

</Link>

<Button @click="() =>
deleteItem(item.id)"
variant="destructive"
size="icon"><Trash /></Button>

</div>

</TableCell>

</TableRow>
                </TableBody>
            </Table>
            <div v-else
class="mt-20 text-center">
                <p
class="text-gray-500">Tidak
ada data pengguna.</p>
            </div>
        </div>
    </AppLayout>
</template>

<script setup lang="ts">
import { Button } from
'@/components/ui/button';
import { Label } from
'@/components/ui/label';
import { Switch } from
'@/components/ui/switch';
import { Table, TableBody,
TableCell, TableHead,
TableHeader, TableRow } from
'@/components/ui/table';
import AppLayout from
'@/layouts/AppLayout.vue';
import { BreadcrumbItem } from
'@/types';
import { Head, Link, router }
from '@inertiajs/vue3';
import { Pen, Trash } from
'lucide-vue-next';
import { ref } from 'vue';
import { User } from
'./types';

const { data } = defineProps<{
data: User[] }>();

const breadcrumbs:
BreadcrumbItem[] = [
    {
        title: 'Dashboard',
        href: '/dashboard',
    },
    {
        title: 'Kelola
Pengguna UMKM',
        href:
route('users.index'),
    },
];
const isDisabled = ref(false);

const handleChange = (status:
boolean, id: number) => {
    const user =
data.find((user) => user.id
=== id) as User;
    if (user) {
        isDisabled.value =
true;
        const label =
document.getElementById(String
(id)) as HTMLAnchorElement;
        label.innerHTML =
'<span class="animate-
pulse">Loading...</span>';
        router.put(

route('users.update.status',
user.umkm.id),
            {
                status:
!status ? 'inactive' :
'active',
            },
            {
                onFinish: ()
=> {

isDisabled.value = false;
```

```
label.innerHTML = status ?
'Aktif' : 'Non-Aktif';
                },

preserveScroll: true,
                preserveState:
true,
            },
        );
    }
};

const deleteItem = (id:
number) => {
    if (confirm('Apakah Anda
yakin ingin menghapus pengguna
ini?')) {

router.delete(route('users.des
troy', id), {
            onSuccess: () => {
                // Optionally,
you can show a success message
or refresh the page
            },
            onError: (error)
=> {

console.error('Error deleting
user:', error);
            },
        });
    }
};
</script>

Lampiran 5 – Jurnal Umum
// □ File:
resources/js/Pages/JurnalUmum/
Index.vue
<template>
    <AppLayout
:breadcrumbs="breadcrumbs">
        <div class="p-4">
            <div class="mb-4
flex items-center justify-
between">
                <h1
class="text-2xl font-bold">
                    Jurnal
Umum <span class="hidden
print:inline">- {{
formatMonth(month as string)
}}</span>
                </h1>
                <div
class="flex items-center gap-
2">

<PrintButton />
                    <Link

:href="

route('jurnal.create', {

month: month,
                                })
                            "

class="print:hidden"
                    >

<Button>+ Tambah Jurnal
Umum</Button>
                    </Link>
                </div>
            </div>

            <div class="mt-4
flex items-center gap-2
print:hidden">
                <Input

type="month"

placeholder="Cari berdasarkan
keterangan, ref, debit, atau
kredit"
                    class="mb-
4 w-max"
                    v-
model="month"

@input="filterMonth"
                />
            </div>

            <Table v-
if="items.length > 0"
class="w-full">
                <TableHeader>
                    <TableRow>

<TableHead>Tanggal</TableHead>

<TableHead>Keterangan</TableHe
ad>

<TableHead>Ref</TableHead>
```

```
<TableHead class="text-
right">Debit</TableHead>

<TableHead class="text-
right">Kredit</TableHead>

<TableHead class="text-center
print:hidden">Aksi</TableHead>

</TableRow>
                </TableHeader>
                <TableBody>
                    <TableRow
v-for="(item, index) in items"
:key="item.id">

<TableCell rowspan="2" v-
if="index % 2 == 0">{{
item.formatted_date
}}</TableCell>

<TableCell :class="{ 'text-
center': index % 2 == 1 }">{{
item.keterangan ??
item.akun?.nama_akun ?? '-'
}}</TableCell>

<TableCell>{{ item.kode_akun
|| '-' }}</TableCell>

<TableCell class="text-
right">{{
formatRupiah(item.debit)
}}</TableCell>

<TableCell class="text-
right">{{
formatRupiah(item.kredit)
}}</TableCell>

<TableCell rowspan="2" v-
if="index % 2 == 0"
class="text-center
print:hidden">

<div class="flex justify-
center gap-2">

<Link
:href="route('jurnal.edit',
item.jurnal_id ?? item.id) +
`?month=${month}`">

<Button size="icon"><Pen
/></Button>
</Link>

<Button
@click="hapus(item.jurnal_id
?? item.id)"
variant="destructive"
size="icon"><Trash /></Button>

</div>

</TableCell>

</TableRow>
                    <TableRow>

<TableCell></TableCell>

<TableCell class="text-center
font-bold">Total</TableCell>

<TableCell></TableCell>

<TableCell class="text-right
font-bold">{{
formatRupiah(sumJurnalUmum('de
bit')) }}</TableCell>

<TableCell class="text-right
font-bold">{{
formatRupiah(sumJurnalUmum('kr
edit')) }}</TableCell>

<TableCell
class="print:hidden"></TableCe
ll>

</TableRow>
                </TableBody>
            </Table>
            <Alert v-else
variant="destructive">
                <InfoIcon
class="h-4 w-4" />

<AlertTitle>Tidak ada data
Jurnal Umum.</AlertTitle>

<AlertDescription> Silakan
tambahkan transaksi jurnal
umum baru untuk bulan {{
formatMonth(month as string)
}}. </AlertDescription>
            </Alert>
        </div>
    </AppLayout>
```

```
</template>

<script setup lang="ts">
import { Alert,
AlertDescription, AlertTitle }
from '@/components/ui/alert';
import { Button } from
'@/components/ui/button';
import PrintButton from
'@/components/ui/button/PrintB
utton.vue';
import { Input } from
'@/components/ui/input';
import { Table, TableBody,
TableCell, TableHead,
TableHeader, TableRow } from
'@/components/ui/table';
import AppLayout from
'@/layouts/AppLayout.vue';
import { formatMonth } from
'@/lib/utils';
import { BreadcrumbItem } from
'@/types';
import { Link, router } from
'@inertiajs/vue3';
import { InfoIcon, Pen, Trash
} from 'lucide-vue-next';
import { onMounted, ref } from
'vue';
import { JurnalUmum } from
'./types';

const props = defineProps<{
data: any[]; month: string
}>();

const formatRupiah = (val:
number | string): string => {
    const n =
parseInt(String(val).replace(/
[^\d]/g, '')) || 0;
    return 'Rp. ' +
n.toLocaleString('id-ID');
};

const hapus = (id: string) =>
{
    if (confirm('Yakin ingin
menghapus data ini?')) {

router.delete(route('jurnal.de
stroy', id), {
            data: {
                month:
month.value,
            },
            preserveScroll:
true,
            preserveState:
false,
        });
    }
};

const breadcrumbs:
BreadcrumbItem[] = [
    {
        title: 'Dashboard',
        href: '/dashboard',
    },
    {
        title: 'Jurnal Umum',
        href: '/jurnal-umum',
    },
];

const month =
ref<string>(props.month);

const filterMonth = () => {
    // update path url
    router.get(
        route('jurnal.index'),
        { month: month.value
},
        {
            preserveScroll:
true,
        },
    );
};
const items =
ref<JurnalUmum[]>([]);

const sumJurnalUmum = (column:
'kredit' | 'debit') => {
    return
items.value.reduce((acc, item)
=> acc + parseInt(item[column]
?? 0), 0);
};

onMounted(() => {
    items.value = props.data;
});
</script>
```