

SI Pengajuan Judul skripsi

1. Halaman login, register dan Role

Buat migration table user

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('users', function (Blueprint $table) {
            $table->id();
            $table->string('nama');
            $table->bigInteger('nim')->unique();
            $table->string('password');
            $table->string('role')->default('mahasiswa');
            $table->rememberToken();
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('users');
    }
};
```

Atur Model user

```
<?php

namespace App\Models;

// use Illuminate\Contracts\Auth\MustVerifyEmail;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Foundation\Auth\User as Authenticatable;
use Illuminate\Notifications\Notifiable;
```

```

use Laravel\Sanctum\HasApiTokens;

class User extends Authenticatable
{
    use HasApiTokens, HasFactory, Notifiable;

    /**
     * The attributes that are mass assignable.
     *
     * @var array<int, string>
     */
    protected $fillable = [
        'nama',
        'nim',
        'password',
        'role',
    ];

    /**
     * The attributes that should be hidden for serialization.
     *
     * @var array<int, string>
     */
    protected $hidden = [
        'password',
        'remember_token',
    ];

    /**
     * The attributes that should be cast.
     *
     * @var array<string, string>
     */
    protected $casts = [
        'email_verified_at' => 'datetime',
    ];

    public function judul()
    {
        return $this->hasMany(judul::class);
    }
}

```

Atur env

```

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=sipengjudskripsi
DB_USERNAME=root
DB_PASSWORD=

```

Jalankan php artisan migrate

Jalankan php artisan make:controller AuthController

Paste ke AuthController

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\User;
use Illuminate\Support\Facades\Hash;

class AuthController extends Controller
{
    public function loginView()
    {
        return view('auth.login');
    }

    public function registerView()
    {
        return view('auth.register');
    }

    public function login(Request $request)
    {
        $request->validate([
            'nim' => 'required|numeric|digits_between:10,10',
            'password' => 'required',
        ]);
        $credentials = $request->only('nim', 'password');

        if (!auth()->attempt($credentials)) {
            return redirect()->back()->withErrors(['nim' => 'Invalid credentials.']);
        }

        return redirect()->route('dashboard');
    }

    public function register(Request $request)
    {
        $request->validate([
            'nim' => 'required|numeric|digits_between:10,10',
            'nama' => 'required|string|max:255',
            'password' => 'required|string|confirmed|min:5|max:10',
        ]);

        $nim = $request->input('nim');
        $nama = $request->input('nama');
        $password = $request->input('password');

        $user = User::create([
            'nim' => $nim,
            'nama' => $nama,
```

```

        'password' => Hash::make($password),
    ]]);

    return redirect()->route('login')->with('success', 'Registrasi Berhasil.');
```

```

    }

    public function logout(Request $request)
    {
        $request->user()->tokens()->delete();
        $request->session()->invalidate();
        $request->session()->regenerateToken();
        return redirect()->route('login');
    }
}

```

Buat php artisan make:middleware CheckUserType

```

<?php

namespace App\Http\Middleware;

use Closure;
use Illuminate\Http\Request;
use Symfony\Component\HttpFoundation\Response;
use Illuminate\Support\Facades\Auth;

class CheckUserType
{
    /**
     * Handle an incoming request.
     *
     * @param  \Closure(\Illuminate\Http\Request): (\Symfony\Component\HttpFoundation\Response)  $next
     */
    public function handle(Request $request, Closure $next): Response
    {
        if (!Auth::check()) {
            return redirect()->route('login');
        }

        $user = Auth::user();

        if ($user->role == 'mahasiswa') {
            // jika level admin, maka hanya boleh akses route cekdatapasien dan postdatapasien
            if (in_array($request->route()->getName(), ['judul', 'judul-tambah', 'judulstore', 'juduledit', 'judulupdate', 'juduldelete', 'judulbukti', 'judulbukti-post', 'judul-surat-pengantar'])) {
                // if ($request->route()->getName() == 'judulall') {
                return $next($request);
            } else {
                return redirect()->back()->with('error', 'Anda tidak memiliki akses ke halaman ini');
            }
        } elseif ($user->role == 'superadmin') {
            return $next($request);
        } elseif ($user->role == 'prodi') {
            // jika level kasir, maka hanya boleh akses route yang diizinkan

```

```

        if (in_array($request->route()->getName(), ['prodi', 'prodi.juduledit', 'prodi.judulupdate'])) {
            return $next($request);
        } else {
            return redirect()->back()->with('error', 'Anda tidak memiliki akses ke halaman ini');
        }
    } elseif ($user->role == 'staffprodi') {
        // jika level dokter, maka hanya boleh akses route yang diizinkan
        if (in_array($request->route()->getName(), ['staffprodi', 'staffprodi.suratadd', 'staffprodi.suratupdate',
'staffprodi.rekapitulasi'])) {
            return $next($request);
        } else {
            return redirect()->back()->with('error', 'Anda tidak memiliki akses ke halaman ini');
        }
    } else {
        return redirect()->back()->with('error', 'Level tidak valid');
    }
}
}
}

```

Tambahkan pada kernel.php

```

protected $middlewareAliases = [
    .....
    'checkUserType' => \App\Http\Middleware\CheckUserType::class,
];

```

Tambahkan Pada Config Auth

```

'guards' => [
    'web' => [
        'driver' => 'session',
        'provider' => 'users',
    ],
    'api' => [
        'driver' => 'token',
        'provider' => 'users',
        'hash' => false,
    ]
],

```

Buat Routenya

<?php

```

<?php

use Illuminate\Support\Facades\Route;
use App\Http\Controllers\AuthController;
use App\Http\Controllers\DashboardController;
use App\Http\Controllers\ProdiController;
use App\Http\Controllers\StaffProdiController;

```

```

use App\Http\Controllers\MahasiswaController;

Route::get('/', function () {
    return view('auth.login');
});

Route::get('/login', [AuthController::class, 'loginView'])->name('login');
Route::get('/register', [AuthController::class, 'registerView'])->name('register');

Route::post('/login', [AuthController::class, 'login'])->name('login.post');
Route::post('/register', [AuthController::class, 'register'])->name('register.post');
Route::get('/logout', [AuthController::class, 'logout'])->middleware('auth')->name('logout');

```

Buat php artisan make:controller DashboardController

```

<?php

namespace App\Http\Controllers;

use App\Models\judul;
use Illuminate\Http\Request;

class DashboardController extends Controller
{
    public function index()
    {
        return view('dashboard.index');
    }
}

```

Buat Form Register

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Register Form</title>
    <!-- Link Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>
    <div class="container vh-100 d-flex justify-content-center align-items-center">
        <div class="card p-4 shadow-lg" style="width: 400px;">
            <h3 class="text-center mb-4">Register</h3>
            <form action="{{ route('register.post') }}" method="POST">
                @csrf

                <!-- NIM Input -->
                <div class="mb-3">
                    <label for="nim" class="form-label">NIM</label>

```

```

        <input type="number" class="form-control" name="nim" id="nim" placeholder="Masukkan NIM"
required minlength="10" maxlength="10">
    </div>
    @error('nim')
        <div class="alert alert-danger">{{ $message }}</div>
    @enderror

    <!-- Nama Input -->
    <div class="mb-3">
        <label for="nama" class="form-label">Nama</label>
        <input type="text" class="form-control" name="nama" id="nama" placeholder="Masukkan Nama
Lengkap" required>
    </div>
    @error('nama')
        <div class="alert alert-danger">{{ $message }}</div>
    @enderror

    <!-- Password Input -->
    <div class="mb-3">
        <label for="password" class="form-label">Password</label>
        <input type="password" class="form-control" name="password" id="password" placeholder="Masukkan
Password" required>
    </div>
    @error('password')
        <div class="alert alert-danger">{{ $message }}</div>
    @enderror

    <!-- Confirm Password Input -->
    <div class="mb-3">
        <label for="confirmPassword" class="form-label">Konfirmasi Password</label>
        <input type="password" class="form-control" name="password_confirmation" id="confirmPassword"
placeholder="Konfirmasi Password" required>
    </div>
    @error('password_confirmation')
        <div class="alert alert-danger">{{ $message }}</div>
    @enderror

    <!-- Submit Button -->
    <button type="submit" class="btn btn-primary w-100">Register</button>

    <!-- Additional Links -->
    <div class="text-center mt-3">
        <span>Sudah punya akun?</span>
        <a href="{{ route('login') }}" class="text-decoration-none">Login</a>
    </div>
</form>
</div>
</div>

<!-- Link Bootstrap JS and Popper.js -->
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.8/dist/umd/popper.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.min.js"></script>
</body>

```

```
</html>
```

Buat Form Login

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Login Form</title>
  <!-- Link Bootstrap CSS -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>
  <div class="container vh-100 d-flex justify-content-center align-items-center">
    <div class="card p-4 shadow-lg" style="width: 400px;">
      <h3 class="text-center mb-4">Login</h3>
      @if (session('success'))
        <div class="alert alert-success">
          {{ session('success') }}
        </div>
      @endif
      <form action="{{ route('login.post') }}" method="POST">
        @csrf
        <!-- NIM Input -->
        <div class="mb-3">
          <label for="nim" class="form-label">NIM</label>
          <input type="number" class="form-control" name="nim" id="nim" placeholder="Masukkan NIM"
required minlength="10" maxlength="10">
        </div>
        @error('nim')
          <div class="alert alert-danger">{{ $message }}</div>
        @enderror

        <!-- Password Input -->
        <div class="mb-3">
          <label for="password" class="form-label">Password</label>
          <input type="password" class="form-control" name="password" id="password" placeholder="Masukkan
Password" required>
        </div>
        @error('password')
          <div class="alert alert-danger">{{ $message }}</div>
        @enderror

        <!-- Submit Button -->
        <button type="submit" class="btn btn-primary w-100">Login</button>

        <!-- Additional Links -->
        <div class="text-center mt-3">
          <span>Belum punya akun?</span>
          <a href="{{ route('register') }}" class="text-decoration-none">Register</a>
        </div>
      </form>
    </div>
  </div>
</body>
</html>
```



```

    </div>
  </form>
</div>
</div>

<!-- Link Bootstrap JS and Popper.js -->
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.8/dist/umd/popper.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.min.js"></script>
</body>
</html>

```

Buat template View - main

```

<!DOCTYPE html>
<html lang="id">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>SIPENGJUDSI</title>

  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/admin-lte@3.1.0/dist/css/adminlte.min.css">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/5.1.3/css/bootstrap.min.css">
  {{-- data tables --}}

  {{-- --}}

  @stack('styles')
</head>
<body class="hold-transition sidebar-mini">

<div class="wrapper">

  <!-- Navbar -->
  <nav class="main-header navbar navbar-expand navbar-white navbar-light">
    <ul class="navbar-nav">
      <li class="nav-item">
        <a class="nav-link" data-widget="pushmenu" href="#" role="button"><i class="fas fa-bars"></i><span
class="d-none d-md-inline ml-2">Sistem Informasi Pengajuan Judul Skripsi</span></a>
      </li>
    </ul>
  </nav>

  <!-- Sidebar -->
  <aside class="main-sidebar sidebar-dark-primary elevation-4">
    <div class="brand-link">
      <span class="brand-text font-weight-light">SIPENGJUDSI</span>
      <p class="text-sm brand-text">Welcome, <strong>{{ Auth::user()->nama }}</strong> ! <br> Role :
<strong>{{ Auth::user()->role }}</strong></p>
    </div>
    <div class="sidebar">

```

```

<nav class="mt-2">
  <ul class="nav nav-pills nav-sidebar flex-column" data-widget="treeview" role="menu" data-
accordion="false">
    <li class="nav-item">
      <a href="{{ route('dashboard') }}" class="nav-link">
        <i class="nav-icon fas fa-home"></i>
        <p>Home</p>
      </a>
    </li>
    @if (Auth::user()->role == 'mahasiswa' || Auth::user()->role == 'superadmin')
    <li class="nav-item">
      <a href="{{ route('judul') }}" class="nav-link">
        <i class="nav-icon fas fa-paper-plane"></i>
        <p>Pengajuan Judul</p>
      </a>
    </li>
    @endif
    @if (Auth::user()->role == 'prodi' || Auth::user()->role == 'superadmin')
    <li class="nav-item">
      <a href="{{ route('prodi') }}" class="nav-link">
        <i class="nav-icon fas fa-list-alt"></i>
        <p>Seleksi Judul</p>
      </a>
    </li>
    @endif
    @if (Auth::user()->role == 'staffprodi' || Auth::user()->role == 'superadmin')
    <li class="nav-item">
      <a href="{{ route('staffprodi') }}" class="nav-link">
        <i class="nav-icon fas fa-envelope"></i>
        <p>Surat bimbingan</p>
      </a>
    </li>
    @endif
    @if (Auth::user()->role == 'staffprodi' || Auth::user()->role == 'superadmin')
    <li class="nav-item">
      <a href="{{ route('staffprodi.rekapitulasi') }}" class="nav-link">
        <i class="nav-icon fas fa-book"></i>
        <p>Laporan Rekapitulasi</p>
      </a>
    </li>
    @endif
    <hr>
    <li class="nav-item">
      <a href="{{ route('logout') }}" class="nav-link">
        <p>Logout</p>
      </a>
    </li>
  </ul>
</nav>
</div>
</aside>

<!-- Content Wrapper -->

```

```

<div class="content-wrapper">
  <div class="content-header">
    <div class="container-fluid">
      <div class="row mb-2">
        <div class="col-sm-6">
          <h4 class="m-0">@yield('title')</h4>
        </div>
      </div>
    </div>
  </div>
  <div class="content">
    <div class="container-fluid">
      @yield('content')
    </div>
  </div>
</div>

<!-- Footer -->
<footer class="main-footer">
  {{-- <div class="float-right d-none d-sm-inline">
    Anything you want
  </div> --}}
  <strong>&copy; 2024 <a href="#">SIPENGJUDSI</a>.</strong> All rights reserved.
</footer>
</div>

<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/admin-lte@3.1.0/dist/js/adminlte.min.js"></script>

@stack('scripts')

@stack('scripts')
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/admin-lte@3.1.0/dist/js/adminlte.min.js"></script>

```

Buat view dashboard – index

```

@extends('template.main')

@section('title')
<h3>Selamat datang di halaman dashboard</h3>
@endsection

@section('content')
<p>halo halo</p>
@endsection

```

2. Halaman Pengajuan Judul

Buat php artisan make:model judul -m

Isi migrationnya

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('juduls', function (Blueprint $table) {
            $table->id();
            $table->string('title');
            $table->text('abstrak');
            $table->enum('status', ['diterima', 'ditolak', 'belum diproses'])->default('belum diproses');
            $table->string('alasan')->nullable();
            $table->string('bukti_pembayaran')->nullable();
            $table->string('surat_bimbingan')->nullable();
            $table->foreignId('user_id')->constrained('users')->onDelete('cascade');
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('juduls');
    }
};
```

Isi modelnya

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class judul extends Model
{
    use HasFactory;
}
```

```

protected $fillable = [
    'judul',
    'abstrak',
    'status',
    'alasan',
    'bukti_pembayaran',
    'surat_bimbingan',
    'user_id',
];

public function user()
{
    return $this->belongsTo(User::class);
}
}

```

Jalankan php artisan migrate

Jalankan php artisan make:controller MahasiswaController

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\judul;
use Illuminate\Support\Facades\Response;
use Illuminate\Support\Facades\Storage;

class MahasiswaController extends Controller
{
    public function judul()
    {
        $data = judul::where('user_id', auth()->user()->id)->get();
        return view('dashboard.mahasiswa.judul', compact('data'));
    }

    public function judultambah()
    {
        return view('dashboard.mahasiswa.judultambah');
    }

    public function judulStore(Request $request)
    {
        $request->validate([
            'title' => 'required|string',
            'abstrak' => 'required|string',
        ]);

        $data = new judul();
        $data->title = $request->title;
        $data->abstrak = $request->abstrak;
    }
}

```

```

    $data->user_id = auth()->user()->id;
    $data->save();

    return redirect()->route('judul');
}

public function judulEdit($id)
{
    $data = judul::find($id);
    return view('dashboard.mahasiswa.juduledit',compact('data'));
}

public function judulUpdate(Request $request, $id)
{
    $request->validate([
        'title' => 'required|string',
        'abstrak' => 'required|string',
    ]);

    $data = judul::find($id);
    $data->title = $request->title;
    $data->abstrak = $request->abstrak;
    $data->save();

    return redirect()->route('judul');
}

public function judulDelete($id)
{
    $data = judul::find($id);
    $data->delete();
    return redirect()->route('judul');
}

public function judulbukti($id)
{
    $data = judul::find($id);
    return view('dashboard.mahasiswa.judulbukti',compact('data'));
}

public function judulbukupost(Request $request, $id)
{
    $request->validate([
        'bukti' => 'required|image|mimes:jpeg,png,jpg,gif,svg|max:10240',
    ]);

    $file = $request->file('bukti');
    $ext = $file->getClientOriginalExtension();
    $newName = rand(100000,1001238912)." ".$ext;
    $file->move(public_path('bukti/'),$newName);

    $data = judul::find($id);
    $data->bukti_pembayaran = $newName;
    $data->save();
    return redirect()->route('judul');
}

```

```

public function judulsurpeng($id)
{
    $data = judul::find($id);
    $filePath = public_path('surat/' . $data->surat_bimbingan);

    if (file_exists($filePath)) {
        return response()->download($filePath);
    } else {
        return redirect()->route('dashboard');
        // return redirect()->back()->with('error', 'Surat bimbingan tidak ditemukan');
    }
}
}

```

Tambahkan route

```

<?php

use Illuminate\Support\Facades\Route;
use App\Http\Controllers\AuthController;
use App\Http\Controllers\DashboardController;
use App\Http\Controllers\ProdiController;
use App\Http\Controllers\StaffProdiController;
use App\Http\Controllers\MahasiswaController;

Route::get('/', function () {
    return view('auth.login');
});

Route::get('/login', [AuthController::class, 'loginView'])->name('login');
Route::get('/register', [AuthController::class, 'registerView'])->name('register');

Route::post('/login', [AuthController::class, 'login'])->name('login.post');
Route::post('/register', [AuthController::class, 'register'])->name('register.post');
Route::get('/logout', [AuthController::class, 'logout'])->middleware('auth')->name('logout');

Route::get('/dashboard', [DashboardController::class, 'index'])->middleware('auth')->name('dashboard');

Route::get('/judul', [MahasiswaController::class, 'judul'])->middleware('checkUserType')->name('judul');
Route::get('/judul-tambah', [MahasiswaController::class, 'judultambah'])->middleware('checkUserType')->
>name('judul-tambah');
Route::post('/judul-post', [MahasiswaController::class, 'judulStore'])->middleware('checkUserType')->
>name('judulstore');
Route::get('/judul-edit/{id}', [MahasiswaController::class, 'judulEdit'])->middleware('checkUserType')->
>name('juduledit');
Route::put('/judul-update/{id}', [MahasiswaController::class, 'judulUpdate'])->middleware('checkUserType')->
>name('judulupdate');
Route::delete('/judul-delete/{id}', [MahasiswaController::class, 'judulDelete'])->middleware('checkUserType')->
>name('juduldelete');
Route::get('/judul-bukti/{id}', [MahasiswaController::class, 'judulbukti'])->middleware('checkUserType')->
>name('judulbukti');

```

```

Route::put('/judul-bukti-post/{id}', [MahasiswaController::class, 'judulbukti-post'])->middleware('checkUserType')->name('judulbukti-post');
Route::get('/judul-suratpengantar/{id}', [MahasiswaController::class, 'judulsurpeng'])->middleware('checkUserType')->name('judulsuratpengantar');

Route::get('/prodi', [ProdiController::class, 'index'])->middleware('checkUserType')->name('prodi');
Route::get('/prodi/judul-edit/{id}', [ProdiController::class, 'edit'])->middleware('checkUserType')->name('prodi.judul-edit');
Route::put('/prodi/judul-update/{id}', [ProdiController::class, 'update'])->middleware('checkUserType')->name('prodi.judul-update');

Route::get('/staffprodi', [StaffProdiController::class, 'index'])->middleware('checkUserType')->name('staffprodi');
Route::get('/staffprodi/upload/{id}', [StaffProdiController::class, 'edit'])->middleware('checkUserType')->name('staffprodi.suratadd');
Route::put('/staffprodi/upload-post/{id}', [StaffProdiController::class, 'update'])->middleware('checkUserType')->name('staffprodi.suratupdate');
Route::get('/staffprodi/rekapitulasi', [StaffProdiController::class, 'rekapitulasi'])->middleware('checkUserType')->name('staffprodi.rekapitulasi');

```

Tambahkan view judul

```

@extends('template.main')

@section('title')
    Selamat datang di halaman judul
@endsection

@section('content')
    <div class="card">
        <div class="card-header">
            <h5 class="card-title">Daftar Judul Anda</h5>
        </div>
        <div class="card-body">
            <a href="{{ route('judul-tambah') }}" class="btn btn-primary my-1">TAMBAH</a>
            <table class="table table-bordered">
                <thead>
                    <tr>
                        <th>No</th>
                        <th>Title</th>
                        <th style="width: 50%;">Abstrak</th>
                        <th>Status</th>
                        <th>Alasan</th>
                        <th style="width: 10%;">Aksi</th>
                    </tr>
                </thead>
                <tbody>
                    @forelse ($data as $p)
                        <tr>

```



```

<td>{{ $p->title }}</td>
<td>{{ $p->abstrak }}</td>
<td>
    @if ($p->status == 'diterima')
        <span class="badge badge-success bg-green-500">Diterima</span>
    @elseif ($p->status == 'ditolak')
        <span class="badge badge-danger bg-red-500">Ditolak</span>
    @else
        <span class="badge badge-secondary bg-gray-500">Belum di Proses</span>
    @endif
</td>
<td>
    @if ($p->alasan != null)
        {{ $p->alasan }}
    @else
        -
    @endif
</td>
<td>
    <a href="{{ route('juduledit', $p->id) }}" class="btn btn-primary">Edit</a>
    <form action="{{ route('juduldelete', $p->id) }}" method="POST" class="d-inline">
        @csrf
        @method('DELETE')
        <button class="btn btn-danger">Delete</button>
    </form>
    @if ($p->status == 'diterima')
        <a href="{{ route('judulbukti', $p->id) }}" class="btn btn-warning m-1 text-sm">Upload
        Bukti Pembayaran</a>
    @endif
    @if ($p->surat_bimbingan != null)
        <a href="{{ route('judulsuratpengantar', $p->id) }}"
        class="btn btn-success m-1 text-sm">Download Surat bimbingan</a>
    @endif
</td>
</tr>
@empty
<tr>
    <td colspan="4">Data Kosong</td>
</tr>
@endforelse
</tbody>
</table>
</div>
</div>
@endsection

```

Tambahkan view judultambah

```
@extends('template.main')
```

```
@section('content')
```

```

<div class="card shadow-sm">
  <div class="card-header bg-primary text-white">
    <h5 class="card-title">Form Tambah Data Judul</h5>
  </div>
  <div class="card-body">
    <form action="{{ route('judulstore') }}" method="POST" id="dataForm">
      @csrf
      <div class="mb-3">
        <label for="title" class="form-label">Title</label>
        <input type="text" class="form-control" name="title" id="title" placeholder="Masukkan judul"
required>
      </div>
      <div class="mb-3">
        <label for="editor" class="form-label">Abstrak</label>
        <div id="editor" style="height: 200px;"></div>
        <input type="hidden" name="abstrak" id="abstract">
        <div class="word-count" id="wordCount">Jumlah kata: 0 / 150</div>
        <div class="error text-danger" id="errorMessage" style="display: none;">Abstrak harus diisi dan
tidak boleh lebih dari 150 kata.</div>
      </div>
      <button type="submit" class="btn btn-primary">Tambah Data</button>
      <a href="{{ route('judul') }}" class="btn btn-secondary">Kembali</a>
    </form>
  </div>
</div>
@endsection

@push('styles')
<link href="https://cdn.quilljs.com/1.3.6/quill.snow.css" rel="stylesheet">
@endpush

@push('scripts')
<script src="https://cdn.quilljs.com/1.3.6/quill.js"></script>
{{-- <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script> --}}
<script>
  // Inisialisasi Quill
  var quill = new Quill('#editor', {
    theme: 'snow'
  });

  // Update word count
  function updateWordCount() {
    const text = quill.getText().trim();
    const words = text ? text.split(/\s+/) : [];
    const wordCount = words.length;
    $('#wordCount').text(`Jumlah kata: ${wordCount} / 150`);

    // Tampilkan pesan kesalahan jika jumlah kata lebih dari 150
    if (wordCount > 150) {
      $('#errorMessage').show();
      return false;
    } else if (wordCount < 1) {
      return false;
    }
  }

```

```

    } else {
      $('#errorMessage').hide();
      return true;
    }
  }

  // Listen for text change in Quill editor
  quill.on('text-change', function() {
    updateWordCount();
  });

  $('#dataForm').on('submit', function(event) {
    event.preventDefault(); // Mencegah pengiriman form default
    if (!updateWordCount()) {
      $('#errorMessage').show();
      return;
    }
    $('#errorMessage').hide();
    $('#abstract').val(quill.root.innerText); // Ambil konten dari Quill
    this.submit(); // Kirim form
  });
</script>
@endpush

```

Buat view juduledit

```

@extends('template.main')

@section('content')
  <div class="card shadow-sm">
    <div class="card-header bg-primary text-white">
      <h5 class="card-title">Form Tambah Data Judul</h5>
    </div>
    <div class="card-body">
      <form action="{{ route('judulupdate', $data->id) }}" method="POST" id="dataForm" >
        @csrf
        @method('PUT')
        <div class="mb-3">
          <label for="title" class="form-label">Title</label>
          <input type="text" class="form-control" name="title" id="title" placeholder="Masukkan judul"
required value="{{ $data->title }}">
        </div>
        <div class="mb-3">
          <label for="editor" class="form-label">Abstrak</label>
          <div id="editor" style="height: 200px;">{{ $data->abstrak }}</div>
          <input type="hidden" name="abstrak" id="abstract">
          <div class="word-count" id="wordCount">Jumlah kata: 0 / 150</div>
          <div class="error text-danger" id="errorMessage" style="display: none;">Abstrak harus diisi dan
tidak boleh lebih dari 150 kata.</div>
        </div>
        <button type="submit" class="btn btn-primary">edit Data</button>
        <a href="{{ route('judul') }}" class="btn btn-secondary">Kembali</a>
      </form>
    </div>
  </div>

```

```

    </form>
  </div>
</div>
@endsection

@push('styles')
<link href="https://cdn.quilljs.com/1.3.6/quill.snow.css" rel="stylesheet">
@endpush

@push('scripts')
<script src="https://cdn.quilljs.com/1.3.6/quill.js"></script>
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script>
  // Inisialisasi Quill
  var quill = new Quill('#editor', {
    theme: 'snow'
  });

  // Update word count
  function updateWordCount() {
    const text = quill.getText().trim();
    const words = text ? text.split(/\s+/) : [];
    const wordCount = words.length;
    $('#wordCount').text(`Jumlah kata: ${wordCount} / 150`);

    // Tampilkan pesan kesalahan jika jumlah kata lebih dari 150
    if (wordCount > 150) {
      $('#errorMessage').show();
      return false;
    } else if (wordCount < 1) {
      return false;
    } else {
      $('#errorMessage').hide();
      return true;
    }
  }

  // Listen for text change in Quill editor
  quill.on('text-change', function() {
    updateWordCount();
  });

  $('#dataForm').on('submit', function(event) {
    event.preventDefault(); // Mencegah pengiriman form default
    if (!updateWordCount()) {
      $('#errorMessage').show();
      return;
    }
    $('#errorMessage').hide();
    $('#abstract').val(quill.root.innerHTML); // Ambil konten dari Quill
    this.submit(); // Kirim form
  });
</script>

```

@endpush

3. Halaman Seleksi Judul

Jalankan php artisan make:controller ProdiController

```
<?php

namespace App\Http\Controllers;

use App\Models\judul;
use Illuminate\Http\Request;

class ProdiController extends Controller
{
    public function index()
    {
        $data = judul::all();
        return view('dashboard.prodi.index',compact('data'));
    }

    public function edit($id)
    {
        $data = judul::find($id);
        return view('dashboard.prodi.edit',compact('data'));
    }

    public function update(Request $request, $id)
    {
        $data = judul::find($id);
        $data->status = $request->status;
        $data->alasan = $request->alasan;
        $data->save();

        return redirect()->route('prodi');
    }
}
```

Route

```
use App\Http\Controllers\ProdiController;

Route::get('/prodi', [ProdiController::class, 'index'])->middleware('checkUserType')->name('prodi');
Route::get('/prodi/judul-edit/{id}', [ProdiController::class, 'edit'])->middleware('checkUserType')->name('prodi.juduledit');
```

```
Route::put('/prodi/judul-update/{id}', [ProdiController::class, 'update'])->middleware('checkUserType')->name('prodi.judulupdate');
```

View Prodi.index

```
@extends('template.main')

@section('title')
    Halaman Prodi
@endsection

@section('content')
    <div class="card">
        <div class="card-header">
            <h5 class="card-title">Daftar Judul Mahasiswa</h5>
        </div>
        <div class="card-body">
            <table class="table table-bordered" id="myTable">
                <thead>
                    <tr>
                        <th>No</th>
                        <th>NIM</th>
                        <th>Nama</th>
                        <th>Judul</th>
                        <th style="width: 40%;">Abstrak</th>
                        <th>Status</th>
                        <th style="width: 10%;">Alasan</th>
                        <th>Aksi</th>
                    </tr>
                </thead>
                <tbody>
                    @forelse ($data as $p)
                        <tr>
                            <td>{{ $loop->iteration }}</td>
                            <td>{{ $p->user->nim }}</td>
                            <td>{{ $p->user->nama }}</td>
                            <td>{{ $p->title }}</td>
                            <td>{{ $p->abstrak }}</td>
                            <td>{{ $p->status }}</td>
                            <td>
                                @if ($p->alasan != null)
                                    {{ $p->alasan }}
                                @else
                                    -
                                @endif
                            </td>
                            <td>
                                <a href="{{ route('prodi.juduledit', $p->id) }}" class="btn btn-primary">Lihat</a>
                            </td>
                        </tr>
                    @empty
                        <tr>
                            <td colspan="8" class="text-center">Data Kosong</td>
                        </tr>
                    @endforelse
                </tbody>
            </table>
        </div>
    </div>
@endsection
```

```

    </table>
  </div>
</div>
@endsection

@push('styles')
<link rel="stylesheet" href="https://cdn.datatables.net/1.11.3/css/jquery.dataTables.min.css">
@endpush
@push('scripts')
<script src="https://cdn.datatables.net/1.11.3/js/jquery.dataTables.min.js"></script>
<script>
  $(document).ready(function() {
    $('#myTable').DataTable({
      paging: true,
      lengthChange: true,
      searching: true,
      ordering: true,
      info: true,
      autoWidth: false,
      responsive: true,
    });
  });
</script>
@endpush

```

View Prodi.edit

```

@extends('template.main')

@section('content')
  <div class="card shadow-sm">
    <div class="card-header bg-primary text-white">
      <h5 class="card-title">Form Tambah Data Judul</h5>
    </div>
    <div class="card-body">
      <form action="{{ route('prodi.judulupdate', $data->id) }}" method="POST" id="dataForm" >
        @csrf
        @method('PUT')
        <div class="mb-3">
          <label for="title" class="form-label">Title</label>
          <p>{{ $data->title }}</p>
        </div>
        <div class="mb-3">
          <label for="editor" class="form-label">Abstrak</label><br>
          <p>{{ $data->abstrak }} </p>
        </div>
        <div class="mb-3">
          <label for="status" class="form-label">Status</label><br>
          <select class="form-select" id="status" name="status" aria-label="Default select example">
            <option {{ $data->status == 'belum diproses' ? 'selected' : '' }} value="belum diproses">belum
diproses</option>
            <option {{ $data->status == 'diterima' ? 'selected' : '' }} value="diterima">Diterima</option>
            <option {{ $data->status == 'ditolak' ? 'selected' : '' }} value="ditolak">Ditolak</option>
          </select>
        </div>
      </div>
    </div>
  </div>

```

```

        <div class="mb-3">
            <label for="alasan" class="form-label">Alasan</label>
            <input type="text" class="form-control" name="alasan" id="alasan" placeholder="Masukkan judul"
value="{{ $data->alasan }}">
        </div>
        <button type="submit" class="btn btn-primary">edit Data</button>
    </form>
</div>
</div>
@endsection

```

Ubah Halaman Dashboard.index

```

@extends('template.main')

@section('title')
<h3>Halaman Home</h3>
@endsection

@section('content')
<div class="card">
    <div class="card-header">
        <h5 class="card-title">Daftar Judul Mahasiswa yang Diterima</h5>
    </div>
    <div class="card-body">
        <table class="table table-bordered text-center">
            <thead>
                <tr>
                    <th>No</th>
                    <th>NIM</th>
                    <th>Nama</th>
                    <th>Judul</th>
                    <th style="width: 40%;">Abstrak</th>
                    <th>Status</th>
                </tr>
            </thead>
            <tbody>
                @forelse ($data as $p)
                    <tr>
                        <td>{{ $loop->iteration }}</td>
                        <td>{{ $p->user->nim }}</td>
                        <td>{{ $p->user->nama }}</td>
                        <td>{{ $p->title }}</td>
                        <td>{{ $p->abstrak }}</td>
                        <td>
                            @if ($p->status == 'diterima')
                                <span class="badge badge-success bg-green-500">Diterima</span>
                            @elseif ($p->status == 'ditolak')
                                <span class="badge badge-danger bg-red-500">Ditolak</span>
                            @else
                                <span class="badge badge-secondary bg-gray-500">Belum di Proses</span>
                            @endif
                        </td>
                    </tr>
                @endforelse
            </tbody>
        </table>
    </div>
</div>

```



```

        </tr>
        @empty
        <tr>
            <td colspan="6" class="text-center">Data Kosong</td>
        </tr>
        @endforelse
    </tbody>
</table>
</div>
</div>
@endsection

```

Tambahkan ke DashboardController

```

<?php

namespace App\Http\Controllers;

use App\Models\judul;
use Illuminate\Http\Request;

class DashboardController extends Controller
{
    public function index()
    {
        $data = judul::where('status', 'diterima')->get();
        return view('dashboard.index', compact('data'));
    }
}

```

4. HALAMAN MELAKUKAN PEMBAYARAN

Tambahkan Pada MahasiswaController

```

public function judulbukti($id)
{
    $data = judul::find($id);
    return view('dashboard.mahasiswa.judulbukti', compact('data'));
}

public function judulbuktiupload(Request $request, $id)
{
    $request->validate([
        'bukti' => 'required|image|mimes:jpeg,png,jpg,gif,svg|max:10240',
    ]);

    $file = $request->file('bukti');
    $ext = $file->getClientOriginalExtension();
}

```

```

$newName = rand(100000,1001238912).".$ext;
$file->move(public_path('bukti/'),$newName);

$data = judul::find($id);
$data->bukti_pembayaran = $newName;
$data->save();
return redirect()->route('judul');
}

```

Tambahkan pada Route

```

Route::get('/judul-bukti/{id}', [MahasiswaController::class, 'judulbukti'])->middleware('checkUserType')->name('judulbukti');

```

```

Route::put('/judul-bukti-post/{id}', [MahasiswaController::class, 'judulbukupost'])->middleware('checkUserType')->name('judulbukupost');

```

Tambahkan Pada view aksi – judul

```

@extends('template.main')

@section('title')
    Selamat datang di halaman judul
@endsection

@section('content')
    <div class="card">
        <div class="card-header">
            <h5 class="card-title">Daftar Judul Anda</h5>
        </div>
        <div class="card-body">
            <a href="{{ route('judul-tambah') }}" class="btn btn-primary my-1">TAMBAH</a>
            <table class="table table-bordered">
                <thead>
                    <tr>
                        <th>No</th>
                        <th>Title</th>
                        <th style="width: 50%;">Abstrak</th>
                        <th>Status</th>
                        <th>Alasan</th>
                        <th style="width: 10%;">Aksi</th>
                    </tr>
                </thead>
                <tbody>
                    @foreach ($data as $p)
                        <tr>
                            <td>{{ $loop->iteration }}</td>
                            <td>{{ $p->title }}</td>
                            <td>{{ $p->abstrak }}</td>
                            <td>
                                @if ($p->status == 'diterima')
                                    <span class="badge badge-success bg-green-500">Diterima</span>
                                @elseif ($p->status == 'ditolak')
                                    <span class="badge badge-danger bg-red-500">Ditolak</span>
                                @else
                                    <span class="badge badge-secondary bg-gray-500">Belum di Proses</span>
                                @endif
                            </td>
                        </tr>
                    @endforeach
                </tbody>
            </table>
        </div>
    </div>

```

```

        @endif
    </td>
    <td>
        @if ($p->alasan != null)
            {{ $p->alasan }}
        @else
            -
        @endif
    </td>
    <td>
        <a href="{{ route('juduledit', $p->id) }}" class="btn btn-primary">Edit</a>
        <form action="{{ route('juduldelete', $p->id) }}" method="POST" class="d-inline">
            @csrf
            @method('DELETE')
            <button class="btn btn-danger">Delete</button>
        </form>
        @if ($p->status == 'diterima')
            <a href="{{ route('judulbukti', $p->id) }}" class="btn btn-warning m-1 text-sm">Upload
            Bukti Pembayaran</a>
        @endif
        @if ($p->surat_bimbingan != null)
            <a href="{{ route('judulsuratpengantar', $p->id) }}"
            class="btn btn-success m-1 text-sm">Download Surat bimbingan</a>
        @endif
    </td>
</tr>
@empty
<tr>
    <td colspan="4">Data Kosong</td>
</tr>
@endforelse
</tbody>
</table>
</div>
</div>
@endsection

```

5. Upload Surat

Jalankan php artisan make:controller StaffProdiController

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\judul;

class StaffProdiController extends Controller
{
    public function index()

```

```

{
    $data = judul::where('status', 'diterima')->whereNotNull('bukti_pembayaran')->get();
    return view('dashboard.staffprodi.index', compact('data'));
}

public function edit($id)
{
    $data = judul::find($id);
    return view('dashboard.staffprodi.edit', compact('data'));
}

public function update(Request $request, $id)
{
    $request->validate([
        'surat' => 'required | max:10240',
    ]);

    $file = $request->file('surat');
    $ext = $file->getClientOriginalExtension();
    $newName = rand(100000, 1001238912) . "." . $ext;
    $file->move(public_path('surat/'), $newName);

    $data = judul::find($id);
    $data->surat_bimbingan = $newName;
    $data->save();

    return redirect()->route('staffprodi');
}

public function rekapitulasi(Request $request)
{
    // dd($request->all());
    if ($request->has('start') && $request->has('end')) {
        $start = date('Y-m-d', strtotime(request('start')));
        $end = date('Y-m-d', strtotime(request('end')));
        $data = judul::where('status', 'diterima')
            ->whereBetween('created_at', [$start, $end])
            ->get();
        $data2 = judul::where('status', 'diterima')
            ->whereNotNull('bukti_pembayaran')
            ->whereBetween('created_at', [$start, $end])
            ->get();
        $data3 = judul::where('status', 'diterima')
            ->whereNull('bukti_pembayaran')
            ->whereBetween('created_at', [$start, $end])
            ->get();
    } else {
        $data = judul::where('status', 'diterima')->get();
        $data2 = judul::where('status', 'diterima')->whereNotNull('bukti_pembayaran')->get();
        $data3 = judul::where('status', 'diterima')->whereNull('bukti_pembayaran')->get();
    }
    return view('dashboard.staffprodi.rekapitulasi', compact('data', 'data2', 'data3'));
}
}

```

Tambahkan Route

```
use App\Http\Controllers\StaffProdiController;

Route::get('/staffprodi', [StaffProdiController::class, 'index'])->middleware('checkUserType')-
>name('staffprodi');
Route::get('/staffprodi/upload/{id}', [StaffProdiController::class, 'edit'])->middleware('checkUserType')-
>name('staffprodi.suratadd');
Route::put('/staffprodi/upload-post/{id}', [StaffProdiController::class, 'update'])->
>middleware('checkUserType')->name('staffprodi.suratupdate');
Route::get('/staffprodi/rekapitulasi', [StaffProdiController::class, 'rekapitulasi'])->middleware('checkUserType')-
>name('staffprodi.rekapitulasi');
```

Tambahkan View staffprodi.index

```
@extends('template.main')

@section('title')
Selamat datang di halaman StaffProdi
@endsection

@section('content')
<div class="card">
    <div class="card-header">
        <h5 class="card-title">Daftar Judul Mahasiswa Yang Diterima </h5>
    </div>
    <div class="card-body">
        <table class="table table-bordered text-center">
            <thead>
                <tr>
                    <th>No</th>
                    <th>NIM</th>
                    <th>Nama</th>
                    <th>Judul</th>
                    <th style="width: 40%;">Abstrak</th>
                    <th>Status</th>
                    <th>Alasan</th>
                    <th>Bukti Pembayaran</th>
                    <th style="width: 5%;">Aksi</th>
                </tr>
            </thead>
            <tbody>
                @forelse ($data as $p)
                <tr>
                    <td>{{ $loop->iteration }}</td>
                    <td>{{ $p->user->nim }}</td>
                    <td>{{ $p->user->nama }}</td>
                    <td>{{ $p->title }}</td>
                    <td>{{ $p->abstrak }}</td>
                    <td>{{ $p->status }}</td>
                    <td>
                        @if ($p->alasan != null)
                            {{ $p->alasan }}
                        @else
                            <p class="text-center"></p>

```

```

        @endif
      </td>
    <td>
      @if ($p->bukti_pembayaran != null)
        <p class="badge badge-success"><i class="fa fa-check-circle" aria-hidden="true"></i></p>
      @else
        <p class="badge badge-danger"><i class="fa fa-times-circle" aria-hidden="true"></i></p>
      @endif
    </td>
    <td>
      <a href="{{ route('staffprodi.suratadd', $p->id) }}" class="btn btn-sm btn-primary"><i class="fa fa-
upload" aria-hidden="true"></i></a>
    </td>
  </tr>
  @empty
  <tr>
    <td colspan="4">Data Kosong</td>
  </tr>
  @endforelse
</tbody>
</table>
</div>
</div>
@endsection

```

Tambahkan View staffprodi.edit

```

@extends('template.main')

@section('content')
  <div class="card shadow-sm">
    <div class="card-header bg-primary text-white">
      <h5 class="card-title">Upload Bukti Pembayaran</h5>
    </div>
    <div class="card-body">
      <form action="{{ route('staffprodi.suratupdate', $data->id) }}" method="POST" id="dataForm"
enctype="multipart/form-data" >
        @csrf
        @method('PUT')
        <div class="mb-3">
          <label for="title" class="form-label">Title</label>
          <p>{{ $data->title }}</p>
        </div>
        <div class="mb-3">
          <label for="editor" class="form-label">Abstrak</label><br>
          <p>{{ $data->abstrak }} </p>
        </div>
        <div class="mb-3">
          <label for="formFile" class="form-label">Bukti Pembayaran</label>
          <p>Nama File : {{ $data->bukti_pembayaran }}</p>
          
        </div>
        <div class="mb-3">
          <label for="formFile1" class="form-label">Upload Surat Bimbingan</label>
          @if($data->surat_bimbingan)
            <p>{{ $data->surat_bimbingan }}</p>
          @endif
        </div>
      </form>
    </div>
  </div>

```

```

        <h6>Atau Ganti</h6>
    @else
        <p>Belum diupload</p>
    @endif
    <input class="form-control" type="file" id="formFile1" name="surat">
</div>
<button type="submit" class="btn btn-primary">Kirim</button>
<a href="{{ route('staffprodi') }}" class="btn btn-secondary ">Kembali</a>
</form>
</div>
</div>
@endsection

```

6.Laporan rekapitulasi

Tambahkan ke StaffProdiController

```

public function rekapitulasi(Request $request)
{
    // dd($request->all());
    if ($request->has('start') && $request->has('end')) {
        $start = date('Y-m-d', strtotime(request('start')));
        $end = date('Y-m-d', strtotime(request('end')));
        $data = judul::where('status', 'diterima')
            ->whereBetween('created_at', [$start, $end])
            ->get();
        $data2 = judul::where('status', 'diterima')
            ->whereNotNull('bukti_pembayaran')
            ->whereBetween('created_at', [$start, $end])
            ->get();
        $data3 = judul::where('status', 'diterima')
            ->whereNull('bukti_pembayaran')
            ->whereBetween('created_at', [$start, $end])
            ->get();
    } else {
        $data = judul::where('status', 'diterima')->get();
        $data2 = judul::where('status', 'diterima')->whereNotNull('bukti_pembayaran')->get();
        $data3 = judul::where('status', 'diterima')->whereNull('bukti_pembayaran')->get();
    }
    return view('dashboard.staffprodi.rekapitulasi', compact('data', 'data2', 'data3'));
}

```

Tambahkan Route

```

Route::get('/staffprodi/rekapitulasi', [StaffProdiController::class, 'rekapitulasi'])->middleware('checkUserType')->name('staffprodi.rekapitulasi');

```

Tambahkan View staffprodi – rekapitulasi

```

@extends('template.main')

```

```

@section('title')
    Halaman Laporan Rekapitulasi
@endsection

@section('content')
    <div class="card">
        <div class="card-header">
            <div class="row border-bottom">
                <div class="col">
                    <h2 class="mb-0">Laporan Rekapitulasi</h2>
                </div>
                <div class="col d-flex justify-content-lg-between">
                    <form action="{{ route('staffprodi.rekapitulasi') }}" method="GET" class="d-flex" style="max-width:
300px">
                        <label for="start" class="me-2 m-2">Start</label>
                        <input type="date" name="start" value="{{ request()->query('start') ?? date('Y-m-d') }}"
class="form-control me-2" width="50">
                        <label for="end" class="me-2 m-2">End</label>
                        <input type="date" name="end" value="{{ request()->query('end') ?? date('Y-m-d') }}"
class="form-control me-2">
                        <button type="submit" class="btn btn-primary mx-2">Cari</button>
                    </form>
                    <a href="javascript:void(0);" class="btn btn-primary" onclick="printPage()">Cetak</a>
                </div>
            </div>
        </div>
        <div class="card-body">
            <div class="row mb-4">
                <div class="col-auto">
                    <div class="card text-center">
                        <div class="card-body">
                            <h5 class="card-title">Jumlah Judul Diterima</h5>
                            <h1 class="display-4 text-primary">{{ $data->count() }}</h1>
                        </div>
                    </div>
                </div>
                <div class="col-auto">
                    <div class="card text-center">
                        <div class="card-body">
                            <h5 class="card-title">Jumlah Judul Sudah Membayar</h5>
                            <h1 class="display-4 text-success">{{ $data2->count() }}</h1>
                        </div>
                    </div>
                </div>
                <div class="col-auto">
                    <div class="card text-center">
                        <div class="card-body">
                            <h5 class="card-title">Jumlah Judul Belum Membayar</h5>
                            <h1 class="display-4 text-danger">{{ $data3->count() }}</h1>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</hr>

```


<h5 class="mt-4">Judul yang Diterima dan Sudah Membayar</h5>

<table class="table table-striped table-bordered">

<thead>

<tr>

<th>No</th>

<th>NIM</th>

<th>Nama</th>

<th>Judul</th>

<th style="width: 40%;">Abstrak</th>

</tr>

</thead>

<tbody>

@forelse (\$data2 as \$p)

<tr>

<td>{{ \$loop->iteration }}</td>

<td>{{ \$p->user->nim }}</td>

<td>{{ \$p->user->nama }}</td>

<td>{{ \$p->title }}</td>

<td>{{ \$p->abstrak }}</td>

</tr>

@empty

<tr>

<td colspan="5" class="text-center">Data Kosong</td>

</tr>

@endforelse

</tbody>

</table>

<hr>

<h5 class="mt-4">Judul yang Diterima & Belum Membayar</h5>

<table class="table table-striped table-bordered">

<thead>

<tr>

<th>No</th>

<th>NIM</th>

<th>Nama</th>

<th>Judul</th>

<th style="width: 40%;">Abstrak</th>

</tr>

</thead>

<tbody>

@forelse (\$data3 as \$p)

<tr>

<td>{{ \$loop->iteration }}</td>

<td>{{ \$p->user->nim }}</td>

<td>{{ \$p->user->nama }}</td>

<td>{{ \$p->title }}</td>

<td>{{ \$p->abstrak }}</td>

</tr>

@empty

<tr>

<td colspan="5" class="text-center">Data Kosong</td>

</tr>

@endforelse

</tbody>

</table>

```
</div>
</div>
@endsection

@push('scripts')
<script>
    function printPage() {
        window.print();
    }
</script>
@endpush
```

Daftar Controller

AuthController

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\User;
use Illuminate\Support\Facades\Hash;

class AuthController extends Controller
{
    public function loginView()
    {
        return view('auth.login');
    }

    public function registerView()
    {
        return view('auth.register');
    }

    public function login(Request $request)
    {
        $request->validate([
            'nim' => 'required|numeric|digits_between:10,10',
            'password' => 'required',
        ]);
        $credentials = $request->only('nim', 'password');

        if (!auth()->attempt($credentials)) {
            return redirect()->back()->withErrors(['nim' => 'Invalid credentials.']);
        }

        return redirect()->route('dashboard');
    }

    public function register(Request $request)
    {
        $request->validate([
            'nim' => 'required|numeric|digits_between:10,10',
```

```

        'nama' => 'required|string|max:255',
        'password' => 'required|string|confirmed|min:5|max:10',
    ));

    $nim = $request->input('nim');
    $nama = $request->input('nama');
    $password = $request->input('password');

    $user = User::create([
        'nim' => $nim,
        'nama' => $nama,
        'password' => Hash::make($password),
    ]);

    return redirect()->route('login')->with('success', 'Registrasi Berhasil.');
```

```

    }

    public function logout(Request $request)
    {
        $request->user()->tokens()->delete();
        $request->session()->invalidate();
        $request->session()->regenerateToken();
        return redirect()->route('login');
    }
}

```

DashboardController

```

<?php

namespace App\Http\Controllers;

use App\Models\judul;
use Illuminate\Http\Request;

class DashboardController extends Controller
{
    public function index()
    {
        $data = judul::where('status', 'diterima')->get();
        return view('dashboard.index', compact('data'));
    }
}

```

MahasiswaController

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\judul;
use Illuminate\Support\Facades\Response;
use Illuminate\Support\Facades\Storage;

```

```

class MahasiswaController extends Controller
{
    public function judul()
    {
        $data = judul::where('user_id', auth()->user()->id)->get();
        return view('dashboard.mahasiswa.judul',compact('data'));
    }

    public function judultambah()
    {
        return view('dashboard.mahasiswa.judultambah');
    }

    public function judulStore(Request $request)
    {
        $request->validate([
            'title' => 'required|string',
            'abstrak' => 'required|string',
        ]);

        $data = new judul();
        $data->title = $request->title;
        $data->abstrak = $request->abstrak;
        $data->user_id = auth()->user()->id;
        $data->save();

        return redirect()->route('judul');
    }

    public function judulEdit($id)
    {
        $data = judul::find($id);
        return view('dashboard.mahasiswa.juduledit',compact('data'));
    }

    public function judulUpdate(Request $request, $id)
    {
        $request->validate([
            'title' => 'required|string',
            'abstrak' => 'required|string',
        ]);

        $data = judul::find($id);
        $data->title = $request->title;
        $data->abstrak = $request->abstrak;
        $data->save();

        return redirect()->route('judul');
    }

    public function judulDelete($id)
    {
        $data = judul::find($id);
        $data->delete();
        return redirect()->route('judul');
    }
}

```

```

public function judulbukti($id)
{
    $data = judul::find($id);
    return view('dashboard.mahasiswa.judulbukti',compact('data'));
}

public function judulbukupost(Request $request, $id)
{
    $request->validate([
        'bukti' => 'required|image|mimes:jpeg,png,jpg,gif,svg|max:10240',
    ]);

    $file = $request->file('bukti');
    $ext = $file->getClientOriginalExtension();
    $newName = rand(100000,1001238912)." ".$ext;
    $file->move(public_path('bukti/'),$newName);

    $data = judul::find($id);
    $data->bukti_pembayaran = $newName;
    $data->save();
    return redirect()->route('judul');
}

public function judulsurpeng($id)
{
    $data = judul::find($id);
    $filePath = public_path('surat/' . $data->surat_bimbingan);

    if (file_exists($filePath)) {
        return response()->download($filePath);
    } else {
        return redirect()->route('dashboard');
        // return redirect()->back()->with('error', 'Surat bimbingan tidak ditemukan');
    }
}
}

```

ProdiController

```

<?php

namespace App\Http\Controllers;

use App\Models\judul;
use Illuminate\Http\Request;
class ProdiController extends Controller
{
    public function index()
    {
        $data = judul::all();
        return view('dashboard.prodi.index',compact('data'));
    }

    public function edit($id)

```

```

{
    $data = judul::find($id);
    return view('dashboard.prodi.edit',compact('data'));
}

public function update(Request $request, $id)
{
    $data = judul::find($id);
    $data->status = $request->status;
    $data->alasan = $request->alasan;
    $data->save();

    return redirect()->route('prodi');
}
}

```

StaffProdiController

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\judul;

class StaffProdiController extends Controller
{
    public function index()
    {
        $data = judul::where('status', 'diterima')->whereNotNull('bukti_pembayaran')->get();
        return view('dashboard.staffprodi.index', compact('data'));
    }

    public function edit($id)
    {
        $data = judul::find($id);
        return view('dashboard.staffprodi.edit', compact('data'));
    }

    public function update(Request $request, $id)
    {
        $request->validate([
            'surat' => 'required | max:10240',
        ]);

        $file = $request->file('surat');
        $ext = $file->getClientOriginalExtension();
        $newName = rand(100000, 1001238912) . "." . $ext;
        $file->move(public_path('surat/'), $newName);

        $data = judul::find($id);
        $data->surat_bimbingan = $newName;
        $data->save();

        return redirect()->route('staffprodi');
    }
}

```

```

}

public function rekapitulasi(Request $request)
{
    // dd($request->all());
    if ($request->has('start') && $request->has('end')) {
        $start = date('Y-m-d', strtotime(request('start')));
        $end = date('Y-m-d', strtotime(request('end')));
        $data = judul::where('status', 'diterima')
            ->whereBetween('created_at', [$start, $end])
            ->get();
        $data2 = judul::where('status', 'diterima')
            ->whereNotNull('bukti_pembayaran')
            ->whereBetween('created_at', [$start, $end])
            ->get();
        $data3 = judul::where('status', 'diterima')
            ->whereNull('bukti_pembayaran')
            ->whereBetween('created_at', [$start, $end])
            ->get();
    } else {
        $data = judul::where('status', 'diterima')->get();
        $data2 = judul::where('status', 'diterima')->whereNotNull('bukti_pembayaran')->get();
        $data3 = judul::where('status', 'diterima')->whereNull('bukti_pembayaran')->get();
    }
    return view('dashboard.staffprodi.rekapitulasi', compact('data', 'data2', 'data3'));
}
}

```

Route Web

```

<?php

use Illuminate\Support\Facades\Route;
use App\Http\Controllers\AuthController;
use App\Http\Controllers\DashboardController;
use App\Http\Controllers\ProdiController;
use App\Http\Controllers\StaffProdiController;
use App\Http\Controllers\MahasiswaController;

Route::get('/', function () {
    return view('auth.login');
});

Route::get('/login', [AuthController::class, 'loginView']->name('login'));
Route::get('/register', [AuthController::class, 'registerView']->name('register'));

Route::post('/login', [AuthController::class, 'login'])->name('login.post');
Route::post('/register', [AuthController::class, 'register'])->name('register.post');
Route::get('/logout', [AuthController::class, 'logout']->middleware('auth')->name('logout');

Route::get('/dashboard', [DashboardController::class, 'index']->middleware('auth')->name('dashboard');

Route::get('/judul', [MahasiswaController::class, 'judul']->middleware('checkUserType')->name('judul');
Route::get('/judul-tambah', [MahasiswaController::class, 'judultambah']->middleware('checkUserType')->name('judul-tambah');

```

```

Route::post('/judul-post', [MahasiswaController::class, 'judulStore'])->middleware('checkUserType')-
>name('judulstore');
Route::get('/judul-edit/{id}', [MahasiswaController::class, 'judulEdit'])->middleware('checkUserType')-
>name('juduledit');
Route::put('/judul-update/{id}', [MahasiswaController::class, 'judulUpdate'])->middleware('checkUserType')-
>name('judulupdate');
Route::delete('/judul-delete/{id}', [MahasiswaController::class, 'judulDelete'])->middleware('checkUserType')-
>name('juduldelete');
Route::get('/judul-bukti/{id}', [MahasiswaController::class, 'judulbukti'])->middleware('checkUserType')-
>name('judulbukti');
Route::put('/judul-bukti-post/{id}', [MahasiswaController::class, 'judulbukti-post'])->
>middleware('checkUserType')->name('judulbukti-post');
Route::get('/judul-suratpengantar/{id}', [MahasiswaController::class, 'judulsurpeng'])->
>middleware('checkUserType')->name('judulsuratpengantar');

Route::get('/prodi', [ProdiController::class, 'index'])->middleware('checkUserType')->name('prodi');
Route::get('/prodi/judul-edit/{id}', [ProdiController::class, 'edit'])->middleware('checkUserType')-
>name('prodi.juduledit');
Route::put('/prodi/judul-update/{id}', [ProdiController::class, 'update'])->middleware('checkUserType')-
>name('prodi.jdulupdate');

Route::get('/staffprodi', [StaffProdiController::class, 'index'])->middleware('checkUserType')-
>name('staffprodi');
Route::get('/staffprodi/upload/{id}', [StaffProdiController::class, 'edit'])->middleware('checkUserType')-
>name('staffprodi.suratadd');
Route::put('/staffprodi/upload-post/{id}', [StaffProdiController::class, 'update'])->
>middleware('checkUserType')->name('staffprodi.suratupdate');
Route::get('/staffprodi/rekapitulasi', [StaffProdiController::class, 'rekapitulasi'])->middleware('checkUserType')-
>name('staffprodi.rekapitulasi');

```

Middleware

```

<?php

namespace App\Http\Middleware;

use Closure;
use Illuminate\Http\Request;
use Symfony\Component\HttpFoundation\Response;
use Illuminate\Support\Facades\Auth;

class CheckUserType
{
    /**
     * Handle an incoming request.
     *
     * @param  \Closure(\Illuminate\Http\Request): (\Symfony\Component\HttpFoundation\Response)  $next
     */
    public function handle(Request $request, Closure $next): Response
    {
        if (!Auth::check()) {
            return redirect()->route('login');
        }

        $user = Auth::user();
    }
}

```



```

if ($user->role == 'mahasiswa') {
    // jika level admin, maka hanya boleh akses route cekdatapasien dan postdatapasien
    if (in_array($request->route()->getName(), ['judul', 'judul-tambah', 'judulstore', 'juduledit', 'judulupdate', 'juduldelete', 'judulbukti', 'judulbukti-post', 'judulsuratpengantar'])) {
        // if ($request->route()->getName() == 'judulall') {
        return $next($request);
        } else {
            return redirect()->back()->with('error', 'Anda tidak memiliki akses ke halaman ini');
        }
    } elseif ($user->role == 'superadmin') {
        return $next($request);
    } elseif ($user->role == 'prodi') {
        // jika level kasir, maka hanya boleh akses route yang diizinkan
        if (in_array($request->route()->getName(), ['prodi', 'prodi.juduledit', 'prodi.judulupdate'])) {
            return $next($request);
        } else {
            return redirect()->back()->with('error', 'Anda tidak memiliki akses ke halaman ini');
        }
    } elseif ($user->role == 'staffprodi') {
        // jika level dokter, maka hanya boleh akses route yang diizinkan
        if (in_array($request->route()->getName(), ['staffprodi', 'staffprodi.suratadd', 'staffprodi.suratupdate', 'staffprodi.rekapitulasi'])) {
            return $next($request);
        } else {
            return redirect()->back()->with('error', 'Anda tidak memiliki akses ke halaman ini');
        }
    } else {
        return redirect()->back()->with('error', 'Level tidak valid');
    }
}
}
}

```

