**LAPORAN**

**APLIKASI PEMBAYARAN SPP**

**BERBASIS DEKSTOP**

**Tanggal 13 Maret – 16 Maret**



**MEMBUAT APLIKASI PEMBAYARAN SPP**

**BERBASIS DEKSTOP**

**Disusun oleh :**

**Nama : RISKI RAHMAND**

**Kelas : XII RPL 2**

**NISN : 0051271864**

**REKAYASA PERANGKAT LUNAK**

**SMK NEGERI 1 SIMPANGKATIS**

**TAHUN AJARAN 2023/2024**

**TAMPILAN DAN DESIGN**

|  |  |
| --- | --- |
| Tampilan | Register Masyarakat |
|  | |

|  |  |
| --- | --- |
| Tampilan | Login Masyarakat |
|  | |

|  |  |
| --- | --- |
| Tampilan | Dashboard Masyarakat |
|  | |

|  |  |
| --- | --- |
| Tampilan | Detail Lelang |
|  | |

|  |  |
| --- | --- |
| Tampilan | Dashboard Staff / Admin / Petugas |
|  | |

|  |  |
| --- | --- |
| Tampilan | Pendataan Barang Staff / Admin / Petugas |
|  | |

|  |  |
| --- | --- |
| Tampilan | Aktivitas Lelang Staff / Admin / Petugas |
|  | |

|  |  |
| --- | --- |
| Tampilan | Pendataan Staff / Admin / Petugas |
|  | |

|  |  |
| --- | --- |
| Tampilan | Pendataan Masyarakat Staff / Admin / Petugas |
|  | |

|  |  |
| --- | --- |
| Tampilan | Generate laporan Staff / Admin / Petugas |
|  | |

**CLASS MODEL**

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| Barang.php | App\Models | index | Model barang |
| class Barang extends Model  {      use HasFactory, HasUUID;      protected $table = 'tb\_barang';      protected $primaryKey = 'id\_barang';      protected $keyType = 'string';      public $incrementing = false;      public $timestamps = false;      protected $fillable = [          'nama\_barang',          'tgl',          'harga\_awal',          'deskripsi\_barang',      ];      public function images() {          return $this->morphMany(Image::class, 'imageable');      }  } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| History.php | App\Models | index | Model History |
| class History extends Model  {      use HasFactory, HasUUID;      protected $table = 'histroy\_lelang';      protected $primaryKey = 'id\_history';      protected $keyType = 'string';      public $incrementing = false;      public $timestamps = false;      protected $fillable = [          'id\_lelang',          'id\_barang',          'id\_user',          'penawaran\_harga',      ];      public function user() {          return $this->belongsTo(Masyarakat::class, 'id\_user', 'id\_user');      }    } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| Image.php | App\Models | index | Model Model |
| class Image extends Model  {      use HasFactory;      protected $table = 'tb\_image';      public $timestamps = false;      protected $fillable = [          'url'      ];      public function imageable() {          return $this->morphTo();      }  } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| Lelang.php | App\Models | index | Model Lelang |
| class Lelang extends Model  {      use HasFactory, HasUUID;      protected $table = 'tb\_lelang';      protected $primaryKey = 'id\_lelang';      protected $keyType = 'string';      public $incrementing = false;      public $timestamps = false;      protected $fillable = [          'id\_barang',          'tgl\_lelang',          'harga\_akhir',          'id\_user',          'id\_petugas',          'status',      ];      public function barang() {          return $this->belongsTo(Barang::class, 'id\_barang', 'id\_barang');      }      public function petugas() {          return $this->belongsTo(Petugas::class, 'id\_petugas', 'id\_petugas');      }      public function winner() {          return $this->belongsTo(Masyarakat::class, 'id\_user', 'id\_user');      }      public function histories() {          return $this->hasMany(History::class, 'id\_lelang', 'id\_lelang')->orderBy('penawaran\_harga', 'desc');      }      public function scopeIsSubmited($query, String $id\_user) {          return $this->histories()->where('id\_user', $id\_user);      }  } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| Level.php | App\Models | index | Model Level |
| class Level extends Model  {      use HasFactory;      protected $table = 'tb\_level';      protected $primaryKey = 'id\_level';      public $timestamps = false;      protected $fillable = [          'level'      ];      public function users() {          return $this->hasMany(Petugas::class, 'id\_level', 'id\_level');      }  } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| Masyarakat.php | App\Models | index | Model Masyarakat |
| class Masyarakat extends Authenticatable  {      use HasApiTokens, HasFactory, Notifiable, HasUUID;      protected $table = 'tb\_masyarakat';      protected $primaryKey = 'id\_user';      protected $keyType = 'string';      public $incrementing = false;      public $timestamps = false;      protected $fillable = [          'nama\_lengkap',          'username',          'password',          'telp',      ];      public function avatar() {          return $this->morphOne(Image::class, 'imageable');      }      public function histories() {          return $this->hasMany(History::class, 'id\_user', 'id\_user');      }      public function notifications() {          return $this->morphMany(Notification::class, 'notificationable');      }    } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| Notification.php | App\Models | index | Model Notification |
| class Notification extends Model  {      use HasFactory;      protected $table = 'tb\_notification';      public $timestamps = false;      protected $fillable = [          'message',          'status'      ];      public function notificationable() {          return $this->morphTo();      }  } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| Petugas.php | App\Models | index | Model Petugas |
| class Petugas extends Authenticatable  {      use HasApiTokens, HasFactory, Notifiable, HasUUID;      protected $table = 'tb\_petugas';      protected $primaryKey = 'id\_petugas';      protected $keyType = 'string';      public $incrementing = false;      public $timestamps = false;      protected $fillable = [          'nama\_petugas',          'username',          'password',          'id\_level',      ];      protected $hidden = [          'password',      ];      public function level() {          return $this->belongsTo(Level::class, 'id\_level', 'id\_level');      }      public function avatar() {          return $this->morphOne(Image::class, 'imageable');      }      public function lelangs() {          return $this->hasMany(Lelang::class, 'id\_petugas', 'id\_petugas');      }  } | | | |

CLASS CONTROLLER

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| AuthController.php | App\Http\Controllers\CMS | index | Menampilkan view Login |
| public function index() {          return view('cms.auth.login');      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| AuthController.php | App\Http\Controllers\CMS | \_login(LoginRequest $request) | Memproses login Admin |
| public function \_login(LoginRequest $request)      {          $credential = $request->validated();          if (Auth::guard('cms')->attempt($credential)) {              $request->session()->regenerate();              return redirect()->intended(route('view.home.cms'));          }          return back()->withErrors([              'username' => 'The provided credentials do not match our records.',          ])->onlyInput('username');      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| AuthController.php | App\Http\Controllers\CMS | \_logout(Request $request) | Memproses logout Admin |
| public function \_logout(Request $request)      {          Auth::guard('cms')->logout();          return redirect('/');      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| BarangController.php | App\Http\Controllers\CMS | index(Barang $barang = null) | Menampilkan view Pendataan Barang |
| Public function index(Barang $barang = null)      {          $barang\_list = Barang::with('images')->get();          $barang\_detail = $barang;          return view('cms.barang.index', compact('barang\_list', 'barang\_detail'));      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| BarangController.php | App\Http\Controllers\CMS | \_store(BarangRequest $request) | Menambahkan Barang |
| public function \_store(BarangRequest $request)      {          $credential = Arr::except($request->validated(), ['images']);          $images = $request->validated('images');          try {              DB::transaction(function () use ($credential, $images) {                  $barang = Barang::create($credential);                  foreach ($images as $image) {                      $barang->images()->create([                          'url' => $image                      ]);                  }              });              return redirect(route('view.barang'))->with('success', 'Berhasil menambahkan barang');          } catch (Exception $e) {              return redirect(route('view.barang'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| BarangController.php | App\Http\Controllers\CMS | \_update(Barang $barang, BarangRequest $request) | Memper baruhi Barang |
| public function \_update(Barang $barang, BarangRequest $request) {          $credential = Arr::except($request->validated(), ['images']);          $images = $request->images == null ? [] : $request->validated('images');          try {              DB::transaction(function () use ($barang, $credential, $images) {                  $barang->update($credential);                  $barang->images()->whereNotIn('url', $images)->delete();                  foreach ($images as $image) {                      $barang->images()->create([                          'url' => $image                      ]);                  }              });              return redirect(route('view.barang'))->with('success', 'Berhasil memperbaruhi barang');          } catch (Exception $e) {              return redirect(route('view.barang'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| BarangController.php | App\Http\Controllers\CMS | \_destroy(Barang $barang) | Menghapus Barang |
| public function \_destroy(Barang $barang) {          try {              $barang->delete();              return redirect(route('view.barang'))->with('success', 'Berhasil menghapus barang');          } catch (Exception $e) {              return redirect(route('view.barang'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| HomeController.php | App\Http\Controllers\CMS | index() | Menampilkan view Home Admin |
| public function index() {          $barangCount = Barang::count();          $lelangCount = Lelang::count();          $petugasCount = Petugas::where('id\_level', 2)->count();          $adminCount =  Petugas::where('id\_level', 1)->count();          return view('cms.home.index', compact('barangCount', 'lelangCount', 'petugasCount', 'adminCount'));      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| LaporanController.php | App\Http\Controllers\CMS | index() | Menampilkan view laporan |
| public function index() {          $lelang\_list = null;          if(Auth::guard('cms')->user()->level->level == 'petugas') {              $lelang\_list = Auth::guard('cms')->user()->lelangs()->with(['barang.images', 'petugas', 'winner'])->where('status', 'ditutup')->get();          } else {              $lelang\_list = Lelang::with(['barang.images', 'petugas', 'winner'])->where('status', 'ditutup')->get();          }          return view('cms.laporan.index', compact('lelang\_list'));      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| LelangController.php | App\Http\Controllers\CMS | index(Lelang $lelang = null) | Menampilkan view Lelang Admin |
| public function index() {          $lelang\_list = null;          if(Auth::guard('cms')->user()->level->level == 'petugas') {              $lelang\_list = Auth::guard('cms')->user()->lelangs()->with(['barang.images', 'petugas', 'winner'])->where('status', 'ditutup')->get();          } else {              $lelang\_list = Lelang::with(['barang.images', 'petugas', 'winner'])->where('status', 'ditutup')->get();          }          return view('cms.laporan.index', compact('lelang\_list'));      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| LelangController.php | App\Http\Controllers\CMS | \_store(LelangRequest $request) | Menambahkan Lelang |
| public function \_store(LelangRequest $request)      {          $credential = $request->validated();          try {              Auth::guard('cms')->user()->lelangs()->create($credential);              return redirect(route('view.lelang.cms'))->with('success', 'Berhasil menambahkan lelang');          } catch (Exception $e) {              return redirect(route('view.lelang.cms'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| LelangController.php | App\Http\Controllers\CMS | \_update(Lelang $lelang, LelangRequest $request) | Memperbaruhi lelang Lelang |
| public function \_update(Lelang $lelang, LelangRequest $request)      {          try {              $lelang->update(array\_filter($request->validated()));              return redirect(route('view.lelang.cms'))->with('success', 'Berhasil memperbaruhi lelang');          } catch (Exception $e) {              return redirect(route('view.lelang.cms'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| LelangController.php | App\Http\Controllers\CMS | \_open(Lelang $lelang) | Mebuka aktivitas Lelang |
| public function \_open(Lelang $lelang)      {          try {              $data = [                  'harga\_akhir' => null,                  'id\_user' => null,                  'status' => 'dibuka',              ];              $lelang->update($data);              return redirect(route('view.lelang.cms'))->with('success', 'Berhasil membuka lelang');          } catch (Exception $e) {              return redirect(route('view.lelang.cms'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| LelangController.php | App\Http\Controllers\CMS | \_close(Lelang $lelang) | Menutup aktivitas Lelang |
| public function \_open(Lelang $lelang)      {          try {              $data = [                  'harga\_akhir' => null,                  'id\_user' => null,                  'status' => 'dibuka',              ];              $lelang->update($data);              return redirect(route('view.lelang.cms'))->with('success', 'Berhasil membuka lelang');          } catch (Exception $e) {              return redirect(route('view.lelang.cms'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| LelangController.php | App\Http\Controllers\CMS | \_destroy(Lelang $lelang) | Menghapus aktivitas Lelang |
| public function \_destroy(Lelang $lelang)      {          try {              $lelang->delete();              return redirect(route('view.lelang.cms'))->with('success', 'Berhasil menghapus lelang');          } catch (Exception $e) {              return redirect(route('view.lelang.cms'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| MasyarakatController.php | App\Http\Controllers\CMS | index() | Menampilkan View Masyarakat |
| public function index() {          $user\_list = Masyarakat::with('avatar')->get();          return view('cms.masyarakat.index', compact('user\_list'));      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| PetugasController.php | App\Http\Controllers\CMS | index(Petugas $petugas = null) | Menampilkan View Petugas |
| public function index() {          $user\_list = Masyarakat::with('avatar')->get();          return view('cms.masyarakat.index', compact('user\_list'));      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| PetugasController.php | App\Http\Controllers\CMS | \_store(PetugasRequest $request) | Menambahkan Petugas |
| public function \_store(PetugasRequest $request)      {          $credential = Arr::except($request->validated(), ['image']);          $image = $request->validated('image');          try {              DB::transaction(function () use ($credential, $image) {                  $petugas = Petugas::create($credential);                  $petugas->avatar()->create([                      'url' => $image                  ]);              });              return redirect(route('view.petugas.cms'))->with('success', 'Berhasil menambah petugas baru');          } catch (Exception $e) {              return redirect(route('view.petugas.cms'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| PetugasController.php | App\Http\Controllers\CMS | \_update(Petugas $petugas, PetugasRequest $request) | Meperbaruhi Petugas |
| public function \_update(Petugas $petugas, PetugasRequest $request)      {          $credential = Arr::except($request->validated(), ['image']);          $image = $request->validated('image');          try {              DB::transaction(function () use ($petugas, $credential, $image) {                  $petugas->update(array\_filter($credential));                  if($petugas->avatar() !== null && $image != null) {                      $petugas->avatar()->update([                          'url' => $image                      ]);                  }                    if ($petugas->avatar() == null && $image != null) {                      $petugas->avatar()->create([                          'url' => $image                      ]);                  }              });              return redirect(route('view.petugas.cms'))->with('success', 'Berhasil meperbaruhi petugas');          } catch (Exception $e) {              return redirect(route('view.petugas.cms'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| PetugasController.php | App\Http\Controllers\CMS | \_destroy(Petugas $petugas) | Menghapus Petugas |
| public function \_destroy(Petugas $petugas)      {          try {              $petugas->delete();              return redirect(route('view.petugas.cms'))->with('success', 'Berhasil menghapus petugas');          } catch (Exception $e) {              return redirect(route('view.petugas.cms'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| AuthController.php | App\Http\Controllers\Frontend; | indexLogin() | Menampilkan view Login Masyarakat |
| ppublic function indexLogin() {          return view('frontend.auth.login');      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| AuthController.php | App\Http\Controllers\Frontend; | indexRegister () | Menampilkan view Daftar Masyarakat |
| public function indexRegister() {          return view('frontend.auth.register');      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| AuthController.php | App\Http\Controllers\Frontend; | login () | Memproses Login Masyarakat |
| public function \_login(LoginRequest $request) {          $credential = $request->validated();          if (Auth::guard('frontend')->attempt($credential)) {              $request->session()->regenerate();              return redirect()->intended('/');          }          return back()->withErrors([              'Username' => 'The provided credentials do not match our records.',          ])->onlyInput('email');      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| AuthController.php | App\Http\Controllers\Frontend; | \_register(RegisterRequest $request) | Memproses Daftar Masyarakat |
| public function \_register(RegisterRequest $request) {          $credential = Arr::except($request->validated(), ['image', 'user\_password']);          $image = $request->validated('image');          try {              DB::transaction(function () use ($credential, $image) {                  $masyarakat = Masyarakat::create($credential);                  $masyarakat->avatar()->create([                      'url' => $image                  ]);              });              return redirect(route('login'))->with('success', 'Berhasil mendaftarkan pengguna baru');          } catch (Exception $e) {              return back()->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| AuthController.php | App\Http\Controllers\Frontend; | \_logout(Request $request) | Memproses Logout Masyarakat |
| public function \_logout(Request $request) {          Auth::guard('frontend')->logout();          return redirect('/');      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| HomeController.php | App\Http\Controllers\Frontend; | index() | Menampilkan view Home Masyarakat |
| public function index() {          $search = request()->query('search');          $lelang\_list = Lelang::with(['barang.images', 'winner'])->when($search, function($query, $search) {              return $query->whereHas('barang', function($query) use ($search) {                  $query->where('nama\_barang', 'like', "%{$search}%");              });          });          $lelang\_count = $lelang\_list->count();          $lelang\_list = $lelang\_list->paginate(10);          return view('frontend.dashboard.index', compact('lelang\_list', 'lelang\_count'));      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| LelangController.php | App\Http\Controllers\Frontend; | index(Lelang $lelang) | Menampilkan view Detail Lelang Masyarakat |
| public function index(Lelang $lelang)      {          // if($lelang->status == 'ditutup') {          //     return redirect('/');          // }          $barang = $lelang->barang;          $participans = $lelang->histories()->with('user')->get();          $submited = null;          if(Auth::guard('frontend')->check()) {              $submited = $lelang->isSubmited(Auth::guard('frontend')->user()->id\_user)->first();          };          return view('frontend.lelang.index', compact('lelang', 'barang', 'participans', 'submited'));      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| LelangController.php | App\Http\Controllers\Frontend; | \_store(Lelang $lelang, LelangRequest $request) | Menambahkan tawaran pada lelang |
| public function \_store(Lelang $lelang, LelangRequest $request)      {          if($lelang->tgl\_lelang > now()->toDateString()) {              return back()->with('error', 'Lelang belum dimulai');          }          $submited = $lelang->isSubmited(Auth::guard('frontend')->user()->id\_user)->first();          if($submited) {              return abort(404);          }          $harga\_awal = $lelang->barang->harga\_awal;          if ($request->validated('penawaran\_harga') <= $harga\_awal) {              return back()->withErrors([                  'penawaran\_harga' => 'Harga yang anda berikan lebih rendah dari harga awal'              ])->onlyInput('penawaran\_harga');          }          try {              $lelang->histories()->create([                  'id\_barang' => $lelang->id\_barang,                  'id\_user' => Auth::guard('frontend')->user()->id\_user,                  'penawaran\_harga' => $request->validated('penawaran\_harga'),              ]);              return back()->with('success', 'Berhasil menambahkan tawaran');          } catch (Exception $e) {              return redirect(route('view.barang'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| NotificationController.php | App\Http\Controllers\Frontend; | \_all() | Mengambil semua notifikasi Masyarakat |
| public function \_all() {          $notifications = Auth::guard('frontend')->user()->notifications()->where('status', 'belum\_dibaca')->get(['id', 'message', 'status']);          return response()->json([              'data' => $notifications          ]);      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| NotificationController.php | App\Http\Controllers\Frontend; | \_readall() | Membaca semua notifikasi |
| public function  \_readall()      {          try {              $notifications = Auth::guard('frontend')->user()->notifications;              foreach ($notifications as $notif) {                  $notif->update([                      'status' => 'dibaca'                  ]);              }              return back()->with('success', 'Berhasil membaca semua notifikasi');          } catch (Exception $e) {              return redirect(route('view.barang'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| ImageUploadController.php | App\Http\Controllers; | \_upload(Request $request) | Mengunggah gambar |
| public function \_upload(Request $request) {          $request->validate([              'file' => ['required', 'file', 'mimes:png,jpg,jpeg']          ]);          $image = $request->file('file');          $store = $image->store('images', 'public');          $url = Storage::disk('public')->url($store);          return response()->json([              'url' => $url          ]);      }              return back()->with('success', 'Berhasil membaca semua notifikasi');          } catch (Exception $e) {              return redirect(route('view.barang'))->with('error', $e->getMessage());          }      } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nama File | Name Space | Nama Fungsi | Fungsi |
| web.php |  |  | Mendaftarkan semua url |
| Route::get('/', [HomeController::class, 'index'])->name('home');  Route::get('/login', [AuthController::class, 'indexLogin'])->name('login')->middleware('guest');  Route::get('/register', [AuthController::class, 'indexRegister'])->name('register')->middleware('guest');  Route::get('/tawar-aja/{lelang}', [LelangController::class, 'index'])->name('tawar');  Route::get('/notifications', [NotificationController::class, '\_all'])->name('api.notifications');  Route::post('/login', [AuthController::class, '\_login'])->name('post.login.frontend')->middleware('guest');  Route::post('/register', [AuthController::class, '\_register'])->name('post.register.frontend')->middleware('guest');  Route::post('/image-upload', [ImageUploadController::class, '\_upload'])->name('post.image-upload');  Route::post('/tawar-aja/{lelang}', [LelangController::class, '\_store'])->name('post.tawaran');  Route::patch('/tawar-aja/{lelang}/{history}', [LelangController::class, '\_update'])->name('patch.tawaran');  Route::patch('/readall-notification', [NotificationController::class, '\_readall'])->name('patch.notfication');  Route::delete('/logout', [AuthController::class, '\_logout'])->name('delete.logout.frontend');  Route::delete('/tawar-aja/{lelang}/{history}', [LelangController::class, '\_destroy'])->name('delete.tawaran');  Route::get('/cms/login', [CMSAuthController::class, 'index'])->name('login.cms')->middleware('guest:cms');  Route::post('/cms/login', [CMSAuthController::class, '\_login'])->name('post.login.cms');  Route::prefix('/cms')->middleware(['auth:cms'])->group(function() {      Route::get('/', [CMSHomeController::class, 'index'])->name('view.home.cms');      Route::get('/barang/{barang?}', [CMSBarangController::class, 'index'])->name('view.barang');      Route::get('/lelang/{lelang?}', [CMSLelangController::class, 'index'])->name('view.lelang.cms');      Route::get('/petugas/{petugas?}', [PetugasController::class, 'index'])->name('view.petugas.cms')->middleware('admin.verify');      Route::get('/masyarakat', [MasyarakatController::class, 'index'])->name('view.masyarakat.cms');      Route::get('/laporan', [LaporanController::class, 'index'])->name('view.laporan.cms');      Route::post('/barang', [CMSBarangController::class, '\_store'])->name('post.barang.cms');      Route::post('/lelang', [CMSLelangController::class, '\_store'])->name('post.lelang.cms');      Route::post('/petugas', [PetugasController::class, '\_store'])->name('post.petugas.cms')->middleware('admin.verify');      Route::patch('/lelang/{lelang}', [CMSLelangController::class, '\_update'])->name('patch.lelang');      Route::patch('/lelang/open/{lelang}', [CMSLelangController::class, '\_open'])->name('patch.lelang.open');      Route::patch('/lelang/close/{lelang}', [CMSLelangController::class, '\_close'])->name('patch.lelang.close');      Route::patch('/barang/{barang}', [CMSBarangController::class, '\_update'])->name('patch.barang.cms');      Route::patch('/petugas/{petugas}', [PetugasController::class, '\_update'])->name('patch.petugas.cms');        Route::delete('/logout', [CMSAuthController::class, '\_logout'])->name('delete.logout.cms');      Route::delete('/barang/{barang}', [CMSBarangController::class, '\_destroy'])->name('delete.barang.cms');      Route::delete('/lelang/{lelang}', [CMSLelangController::class, '\_destroy'])->name('delete.lelang.cms');      Route::delete('/petugas/{petugas}', [PetugasController::class, '\_destroy'])->name('delete.petugas.cms');  }); | | | |