

LAMPIRAN

Lampiran 1.1 File Login Controller

```
credentials = request.validate([
    'username' => 'required',
    'password' => 'required'
])
```

Lampiran 1.2 File Forgot Password

```
function submitResetPasswordForm(request)
    validate request parameters:
        - password: required, must be at least 6
characters long
        - password_confirmation: optional

    update the user's password in the "user" table with
the hashed value of request.password

    delete the record from the "password_resets" table
where email matches request.email
```

Lampiran 1.3 File Login Controller

```
if user attempts are greater than or equal to 3 {
    $blokir = User::where('username',
$credentials['username']->update(['flagblokir' =>
1]);

    // Redirect to the login page with a blocked account
alert
    return redirect('/login')->with('alert', 'Your
account has been blocked!');
}
```

Lampiran 1.4 File Forgot Password

```
validasi_request([
    'email' => 'wajib|email|terdapat:user',
    'password' => 'wajib|minimal:6',
    'password_confirmation' => 'boleh_kosong'
]);

pengguna = User::cari_berdasarkan('email', request-
>email)->perbarui(['password' => buat_hash(request-
>password)]);
```

Lampiran 1.5 File User Seeder

```
function run()
    admin2 = createUser([
        'nama' => 'adminppid',
        'username' => 'adminppid111',
        'email' => 'email5@gmail.com',
        'role' => '1',
        'password' => hashPassword('111111')
    ])
}
```

Lampiran 1.6 File Session

```
'lifetime' => env('SESSION_LIFETIME', 120),
'expire_on_close' => false,
```

Lampiran 1.7 File Login Controller

```
if (Auth::attempt($credentials)) {
    $request->session()->regenerate();
    sessionId = $request->session()->getId();
    $request->session()->regenerate();
}
```

Lampiran 1.8 File Check Session

```
batasWaktuSesi = 60; // 20 detik
```

```

if ($request->session()->memiliki('lastActivityTime')
&&
    waktu_sekarang() - $request->session()-
>dapatkan('lastActivityTime') > batasWaktuSesi) {
    $request->session()->mencabut();
    $request->session()->regenerasiToken();
    return redirect()->rute('login')-
>dengan('message', 'Sesi Anda telah berakhir. Silakan
masuk kembali.');
```

Lampiran 1.9 File Login Controller

```

sessionId = $request->session()->getId();

pengguna = User::cari_berdasarkan('username',
$credentials['username']->perbarui(['attempts' =>
0]);

Log::info('Pengguna berhasil login. Session ID: ' .
sessionId);
```

Lampiran 1.10 File Login Controller Api

```

login = validasi_request([
    'username' => 'wajib',
    'password' => 'wajib',
]);

if (! Auth::attempt(login)) {

    msg = 'invalid credential';
    kembalikan respons()->json(msg);
}

accesstoken = Auth::user()->buatToken('accessToken')-
>accessToken;

return respons()->json([
    'user' => Auth::user(),
```

```
        'access_token' => accesstoken
    ]);
```

Lampiran 1.11 File Layanan Informasi Controller

```
dataValidasi = validasi_request([
    'nama' => 'wajib|maksimal:50',
    'ktp' => 'wajib|maksimal:16|minimal:16',
    'alamat' => 'wajib',
    'pekerjaan' => 'wajib|maksimal:255',
    'telepon' => 'wajib',
    'email' => 'wajib|email',
    'rincian_informasi' => 'wajib',
    'tujuan_penggunaan' => 'wajib',
    'cara_memperoleh_informasi' => 'wajib',
    'cara_mendapat_salinan' => 'wajib',
    'image' =>
    'wajib|jenis_berkas:jpeg,png,jfif|maksimal:5000'
]);
```

Lampiran 1.12 File Admin Layanan Informasi Controller

```
dataPermohonan = PermohonanPerorangan::all();

untuk setiap permohonan dalam dataPermohonan {
    permohonan->ktp = Crypt::decrypt(permohonan->ktp);
}
```

Lampiran 1.13 File App

```
'timezone' => 'asia/jakarta',
```

Lampiran 1.14 File Route Web

```
Route::get('/dashboard', function(){
    return view('admin.dashboard');
})->middleware('auth');
```

Lampiran 1.15 File Layanan Informasi Controller

```
data = [
```

```

        'nama' => request->nama,
        'ktp' => Crypt::encrypt(request->ktp),
        'alamat' => request->alamat,
        'pekerjaan' => request->pekerjaan,
        'email' => request->email,
        'rincian_informasi' => request->rincian_informasi,
        'tujuan_penggunaan' => request->tujuan_penggunaan,
        'cara_memperoleh_informasi' => request->
>cara_memperoleh_informasi,
        'cara_mendapat_salinan' => request->
>cara_mendapat_salinan,
        'image' => request->image
    ]
}

```

Lampiran 1.16 File Berandaimage Controller

```

BerandaImage::destroy(berandaimage->id)

```

```

redirect('/beranda-ppid')->with('success', 'Data
Berhasil di Hapus !')

```

Lampiran 1.17 File Force HTTPS

```

function handle(request, next) {
    if (tidak(request->secure()) dan env('APP_ENV')
sama dengan 'production') {
        return redirect()->secure(request->
>getRequestUri());
    }
    return next(request);
}

```

Lampiran 1.18 File Suspicious Login Notification

```

function __construct(username, ipAddress) {
    this->username = username
    this->ipAddress = ipAddress
}

function build() {

```

```

    kembalikan from('benedictusradyan@gmail.com')
    ->view('suspicious_login')
    ->subject('Aktivitas Mencurigakan Terdeteksi')
}

```

Lampiran 1.19 Templating Blade

```

<a href="/storage/{{ $item->data_file }}" class="btn
btn-primary"

target="blank"><i class="fa fa-eye"></i></a>

```

Lampiran 1.20 Proteksi CSRF

```

<form

action="/layanandinformasi/PermohonanInformasi/{{
$item->id }}/hapus"

method="POST" class="d-inline">

@csrf

<button class="btn btn-danger"><i class="fa fa-trash
fa-lg"></i></button>

</form>

```

Lampiran 1.21 File Data Profil PPID

```

public function up()
{
    Schema::create('data_profil_ppids', function
(Blueprint $table) {
        $table->id();
        $table->foreignId('profil_id');
        $table->string('judul');
        $table->string('body');
        $table->string('image')->nullable();
        $table->string('data_file')->nullable();
        $table->timestamps();
    });
}

```

```
    });  
}
```

Lampiran 1.22 File env

```
APP_URL=https://dinkesjatim.my.id
```

```
DB_CONNECTION=mysql
```

```
DB_HOST=localhost
```

```
DB_PORT=3306
```

```
DB_DATABASE=dind8689_dinkesjatim
```

```
DB_USERNAME=dind8689_dinkesjatim
```

```
DB_PASSWORD=dinkesjatim2022
```

```
MAIL_DRIVER=smtp
```

```
MAIL_HOST=smtp.gmail.com
```

```
MAIL_PORT=465
```

```
MAIL_USERNAME=benedictusradyan@gmail.com
```

```
MAIL_PASSWORD=tllazruhacyfdcmr
```

```
MAIL_ENCRYPTION=tls
```

```
MAIL_FROM_ADDRESS="hello@example.com"
```

```
MAIL_FROM_NAME="{APP_NAME}"
```

Lampiran 1.23 File Route Api

```
Route::middleware('auth:api')->group(function () {  
    Route::get('/api-perorangan-api',  
[LayananInformasiApiController::class, 'index']);  
});
```

Lampiran 1.24 File login controller api

```
public function login(Request $request)  
{  
    $login = $request->validate([  
        'username' => 'required',  
        'password' => 'required',  
    ]);  
  
    if (! Auth::attempt($login)) {  
        $msg = 'invalid credential';  
        return response()->json($msg);  
    }  
}
```

```

    }

    $accesstoken = Auth::user()-
>createToken('accessToken')->accessToken;
    return response()->json([
        'user' => Auth::user(),
        'access_token' => $accesstoken
    ]);
}

```

Lampiran 1.25 File permohonan perorangan

```

@if (session()->has('success'))
    <div class="alert alert-success" role="alert">
        {{ session('success') }}
    </div>
@endif

@if ($errors->any())
    <div class="alert alert-danger">
        <ul>
            @foreach ($errors->all() as $error)
                <li>{{ $error }}</li>
            @endforeach
        </ul>
    </div>
@endif

```