

2.4.1.5 Create handleKeycloakCallback Function


In the **LoginController.php** file, create a `handleKeycloakCallback` function. This function processes the Keycloak callback by exchanging the authorization code for an access token, enabling secure user authentication and access to protected resources.

```
public function handleKeycloakCallback(Request $request)
{
    try {
        $tokenResponse = Http::asForm()->post(env('KEYCLOAK_BASE_URL') . "/realms/" . env
('KEYCLOAK_REALM') . "/protocol/openid-connect/token", [
            'client_id' => env('KEYCLOAK_CLIENT_ID'),
            'client_secret' => env('KEYCLOAK_CLIENT_SECRET'),
            'redirect_uri' => env('KEYCLOAK_REDIRECT_URI'),
            'grant_type' => 'authorization_code',
            'code' => $request->code,
        ]->json();

        if (!isset($tokenResponse['access_token'])) return redirect('/login')->withErrors('Login failed.');
```

Figure 9: *handleKeycloakCallback* Function

Retrieve the user information using the access token by querying the Keycloak User Info endpoint. Extract the IC number and name from the response to securely access and utilize these details within the application for user identification and processing.

	Document Title: SSO Integration Guideline Document			
	Document Security Level: Public Document	No. Documents: PP24176	Version.: 2.0	Page: 30 of 75

```

$userInfo = Http::withToken($tokenResponse['access_token'])->get(env('KEYCLOAK_BASE_URL') . "/" .
realms/" . env('KEYCLOAK_REALM') . "/protocol/openid-connect/userinfo")->json();

if (!$userInfo || !isset($userInfo['nric'])) return redirect('/login')->withErrors('User data
missing.');
```

Figure 10: Retrieve User Information

Creates or updates the user in the database. Finds a user based on IC number and if the IC number is found, proceed to log the user in. If the IC number is not found or does not match, handle redirect to the login page with login error message.

```

$user = \App\Models\User::updateOrCreate(['ic_number' => $userInfo['nric'], ['name' => $userInfo
['nama']]);

Auth::login($user);

return redirect('/dashboard')->with('success', 'Login successful');

} catch (\Exception $e) {
    return redirect('/login')->withErrors('Login error: ' . $e->getMessage());
}
}
```

Figure 11: Finds IC Number and Stored in The Database