

BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Implementasi adalah tahap penerapan sekaligus pengujian bagi sistem berdasarkan metode penelitian, analisa dan perancangan yang telah dilakukan pada BAB IV. Pada BAB V ini merupakan implementasi hasil rancangan menjadi sebuah aplikasi Sistem Informasi Manajemen Inventaris Berbasis Web dengan studi kasus di PT Railink.

4.1.1 Lingkungan Implementasi

Lingkungan implementasi aplikasi ini terbagi atas beberapa lingkungan yaitu lingkungan perangkat keras, lingkungan perangkat lunak dan lingkungan sumber daya manusia.

4.1.1.1 Lingkungan Perangkat Keras

Sistem Informasi Manajemen Inventaris dikembangkan dan dapat dijalankan dengan spesifikasi perangkat keras sebagai berikut :

- a. Processor : Core i5 Quad Core 2 Ghz
- b. Memory : 8 GB
- c. Storage : SSD 512 GB
- d. Perangkat Jaringan : Wireless Adapter b/g/n/ac 5Ghz
- e. Keyboard & Mouse

4.1.1.2 Lingkungan Perangkat Lunak

Sistem Informasi Manajemen Inventaris dikembangkan dan dapat dijalankan dengan spesifikasi perangkat lunak sebagai berikut :

- f. Sistem Operasi : MacOS 12.4
- g. IDE : PHPStorm dengan lisensi Educational
- h. DBMS : MySQL Versi 8
- i. Bahasa Pemrograman : PHP Versi 8
- j. Framework PHP : Laravel 8
- k. Web Browser : Google Chrome

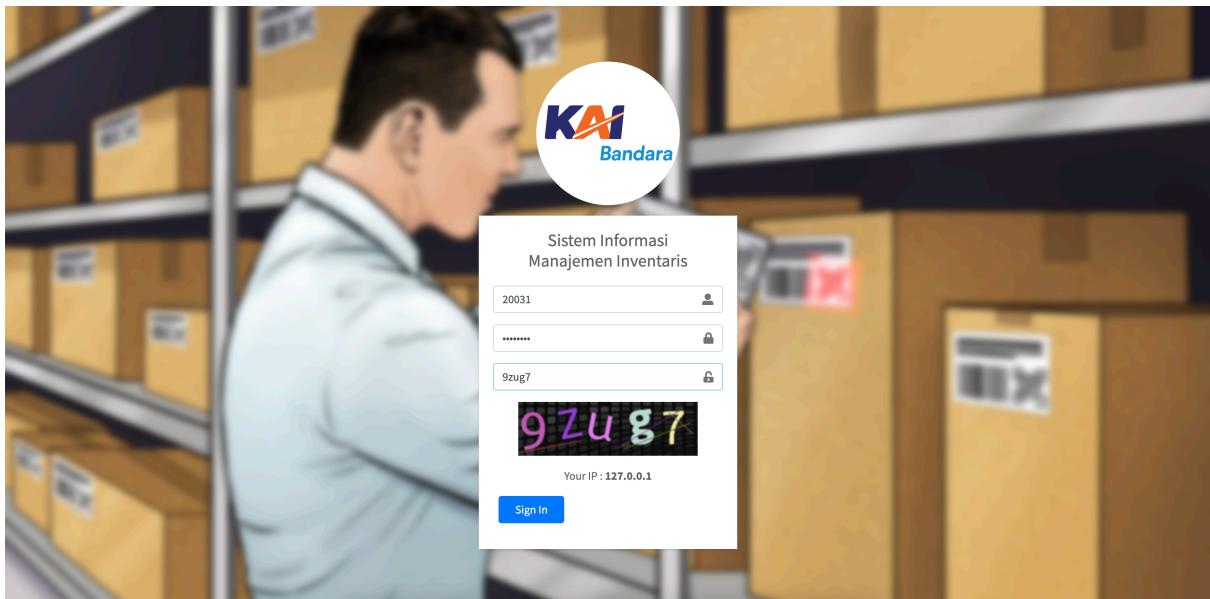
4.1.1.3 Lingkungan Perangkat Brainware

Implementasi Sistem Informasi Manajemen Inventaris dibutuhkan seorang pengguna yang mampu menggunakan dan mengoperasikan komputer dan menjalankan sistem yang telah dibuat sehingga penerapan implementasi sistem dapat berjalan dengan lancar.

4.1.2 Implementasi Antarmuka

4.1.2.1 Halaman Login

Pada halaman login terdapat logo perusahaan, field username, field password, field captcha dan gambar captcha, informasi IP client kemudian tombol Sign In.



Gambar 39 Implementasi Antarmuka – Halaman Login

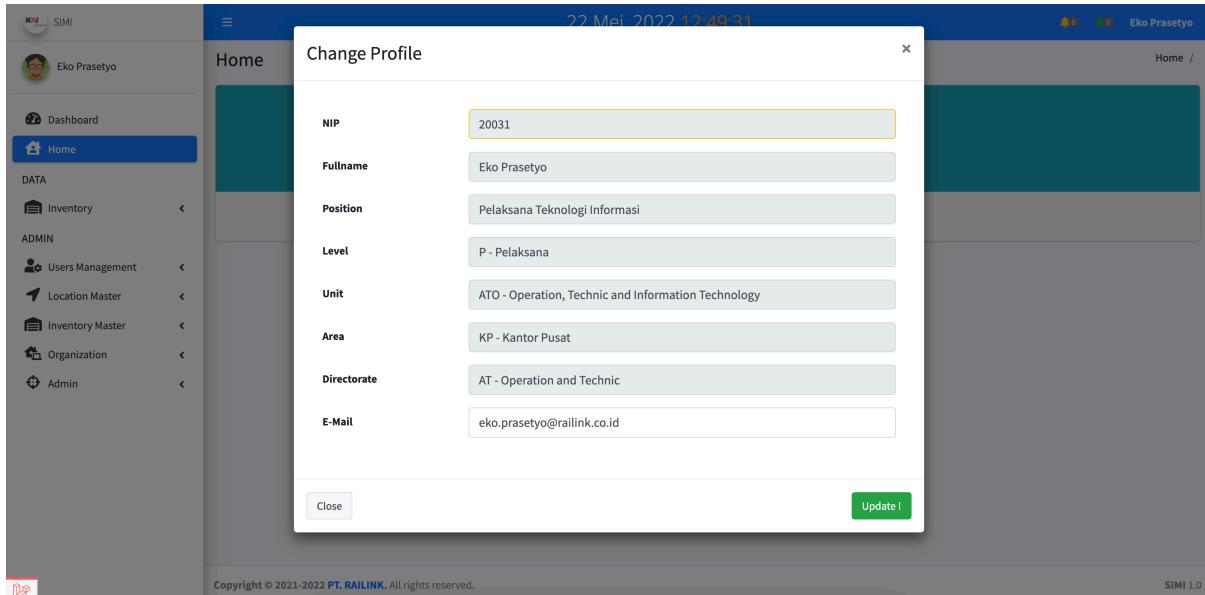
4.1.2.2 Halaman Home

Pada halaman home terdapat menu-menu Sistem Informasi Manajemen Inventaris di samping kiri, kemudian profile pengguna di area tengah dan sudut kanan atas terdapat tombol untuk ganti profile, ganti password dan logout kemudian tombol notifikasi untuk penerimaan inventaris.

Gambar 40 Implementasi Antarmuka – Halaman Home

4.1.2.3 Halaman Ganti Profile

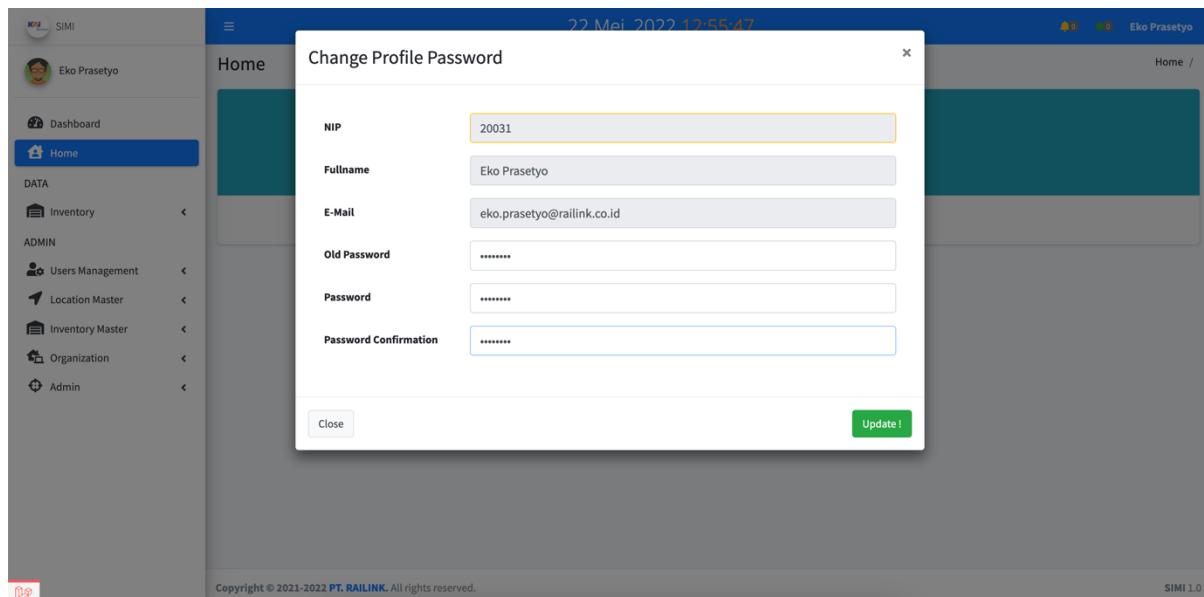
Pada halaman ganti profile, terdapat field-field yang menjelaskan profile dari user yang sedang login. Kemudian field yang dapat diubah hanya field E-Mail.



Gambar 41 Implementasi Antarmuka – Halaman Ganti Profile

4.1.2.4 Halaman Ganti Password

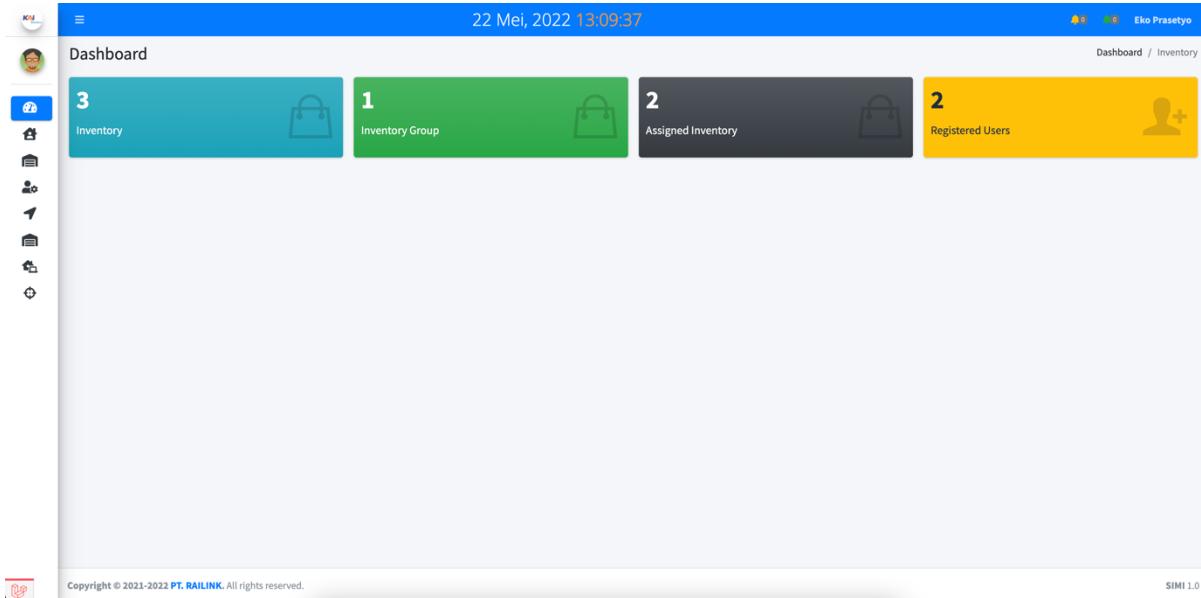
Pada halaman ganti password, terdapat field-field yang menjelaskan profile dari user yang sedang login. Kemudian field yang dapat diubah yaitu field Password Lama, field Password Baru dan field Konfirmasi Password Baru.



Gambar 42 Implementasi Antarmuka – Halaman Ganti Password

4.1.2.5 Halaman Dashboard

Pada halaman dashboard, terdapat balok-balok yang menjelaskan informasi yaitu total inventaris, total grup inventaris, total inventaris yang sudah diserahkan, dan total pengguna yang sudah terdaftar

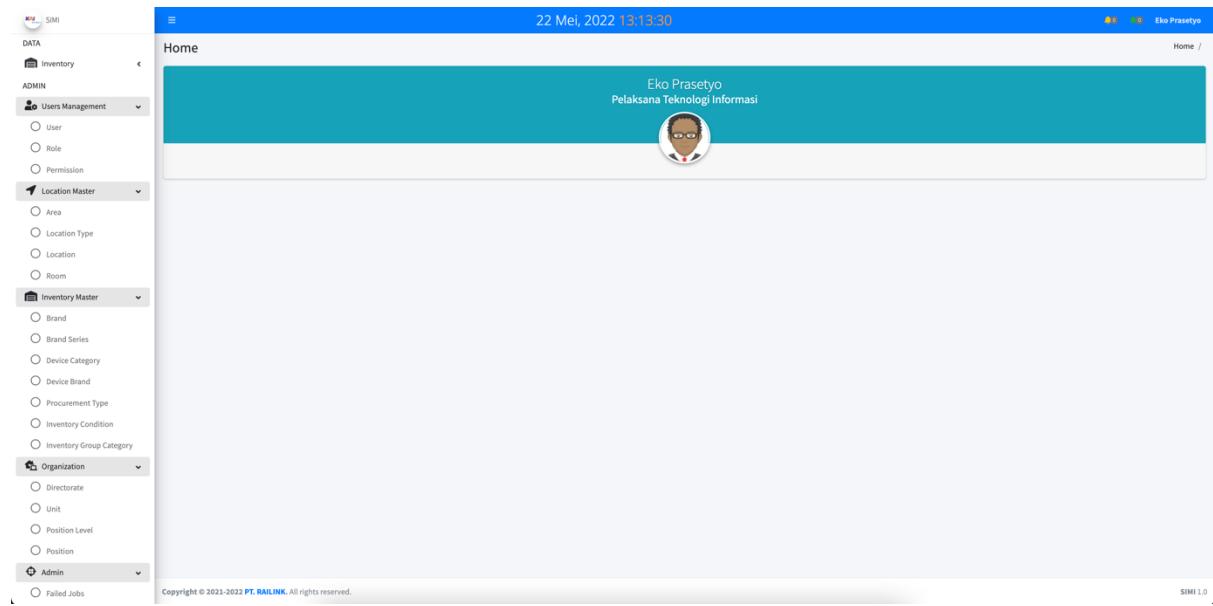


Gambar 43 Implementasi Antarmuka – Halaman Dashboard

4.1.2.6 Halaman Menu Admin

Pada halaman menu admin, terdapat menu-menu yang dibuat untuk mengelola master data sistem dan master data inventaris, diantaranya :

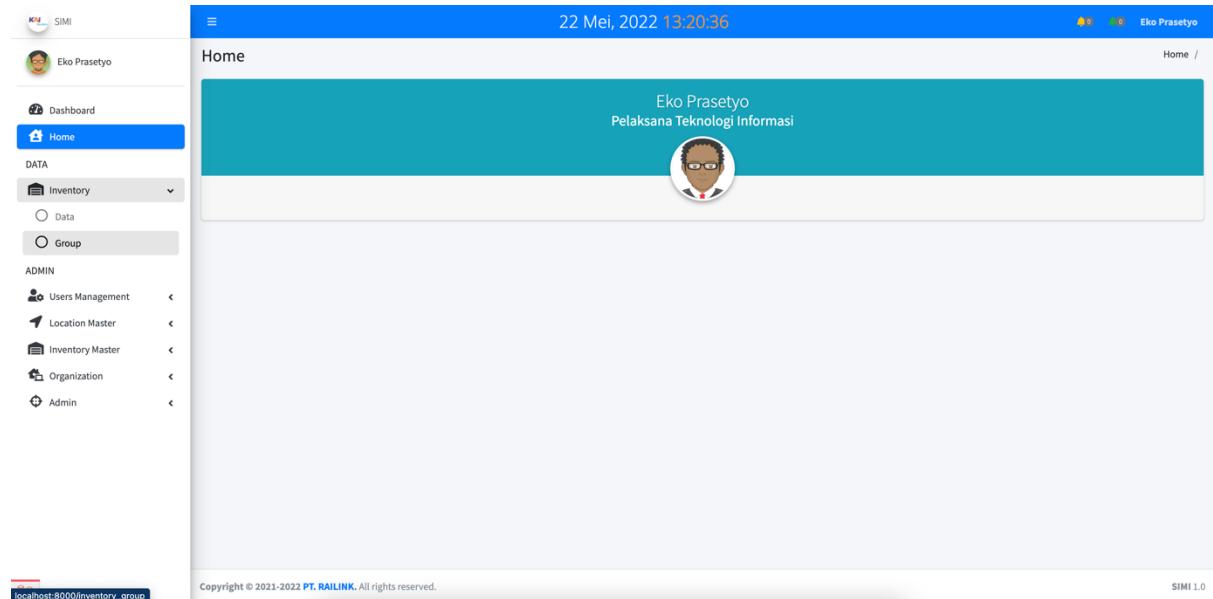
- a. Manajemen User (User, Role dan Permission)
- b. Master Lokasi (Area, Lokasi, Tipe Lokasi dan Ruangan)
- c. Master Inventaris (Merk, Seri Merk, Tipe Inventaris, Merk Inventaris, Tipe Pengadaan, Kondisi Inventaris dan Kategori Group Inventaris)
- d. Master Organisasi (Direktorat, Unit, Level Jabatan, dan Jabatan)
- e. Admin (Failed Jobs)



Gambar 44 Implementasi Antarmuka – Halaman Menu Admin

4.1.2.7 Halaman Menu Inventaris

Pada halaman menu inventaris, terdapat menu-menu yang dibuat untuk mengelola data inventaris, diantaranya data inventaris dan data grup inventaris.



Gambar 45 Implementasi Antarmuka – Halaman Menu Inventaris

4.1.2.8 Halaman List Data Inventaris

Pada halaman list data inventaris, terdapat tombol untuk menambahkan data inventaris baru, kemudian ada dropdown untuk melakukan filter data berdasarkan ruangan. Dilanjutkan dengan tabel yang memiliki kolom No, Nomor, Tipe Inventaris,

Merk, Serial Number, Ruangan, Kondisi, Inventory Group, User, Assign Status dan Actions.

No	Number	Device	Brand	Serial Number	Room
1	RL-INV-LPT-0522-00001	Laptop / Notebook	Apple - MacBook Pro 2020	123	Ruang Server Sudirman Baru - RSVSDB
2	RL-INV-EPC-0522-00001	Embedded PC	Advantech - Uno 2272G	123	Ruang Server Sudirman Baru - RSVSDB
3	RL-INV-PRT-0522-00001	Printer	Bixolon - SRP-352plusII	123	Ruang Server Sudirman Baru - RSVSDB

Gambar 46 Implementasi Antarmuka – Halaman List Data Inventaris 1

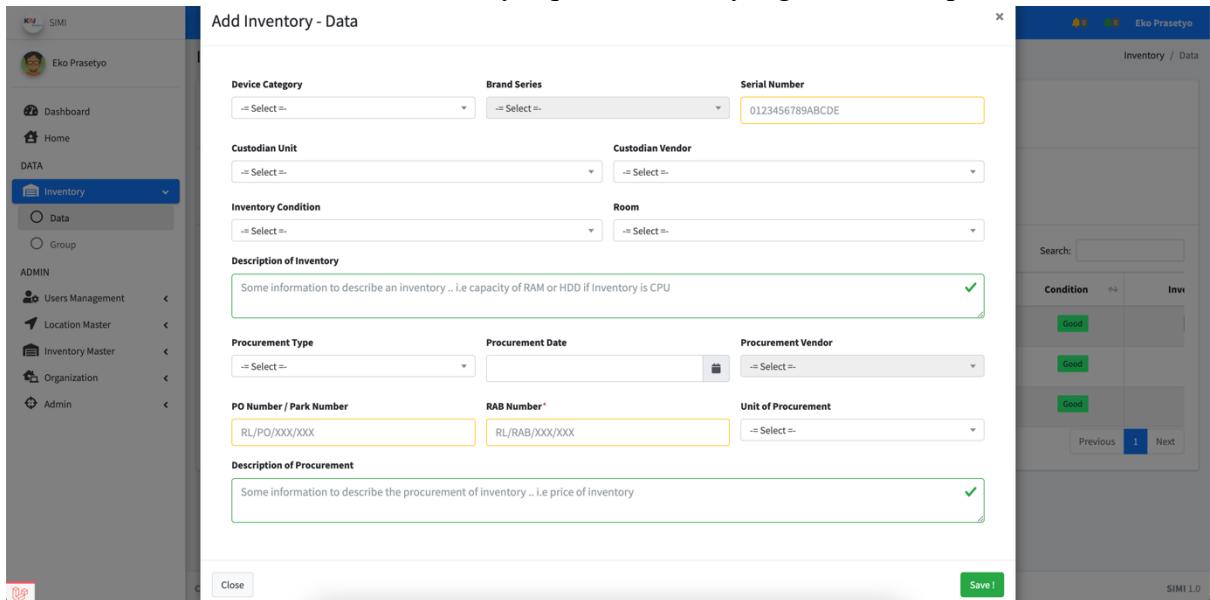
Room	Condition	Inventory Group	User	Assign Status	Action
man Baru - RSVSDB (Sudirman Baru - SDB)	Good	Unavailable	Not Yet Assigned	Unavailable	
man Baru - RSVSDB (Sudirman Baru - SDB)	Good	VM 1	Eko Prasetyo - 20031	Accepted	
man Baru - RSVSDB (Sudirman Baru - SDB)	Good	VM 1	Eko Prasetyo - 20031	Accepted	

Gambar 47 Implementasi Antarmuka – Halaman List Data Inventaris 2

4.1.2.9 Halaman Tambah Data Inventaris

Pada halaman tambah data inventaris, terdapat field-field yang dibutuhkan yaitu Device Category, Brand Series, Serial Number, Custodian Unit, Custodian Vendor, Inventory Condition, Room, Description of Inventory, Procurement Type, Procurement Date, Procurement Vendor, PO Number / Park Number, RAB Number, Unit of Procurement and Description of Procurement. Kemudian

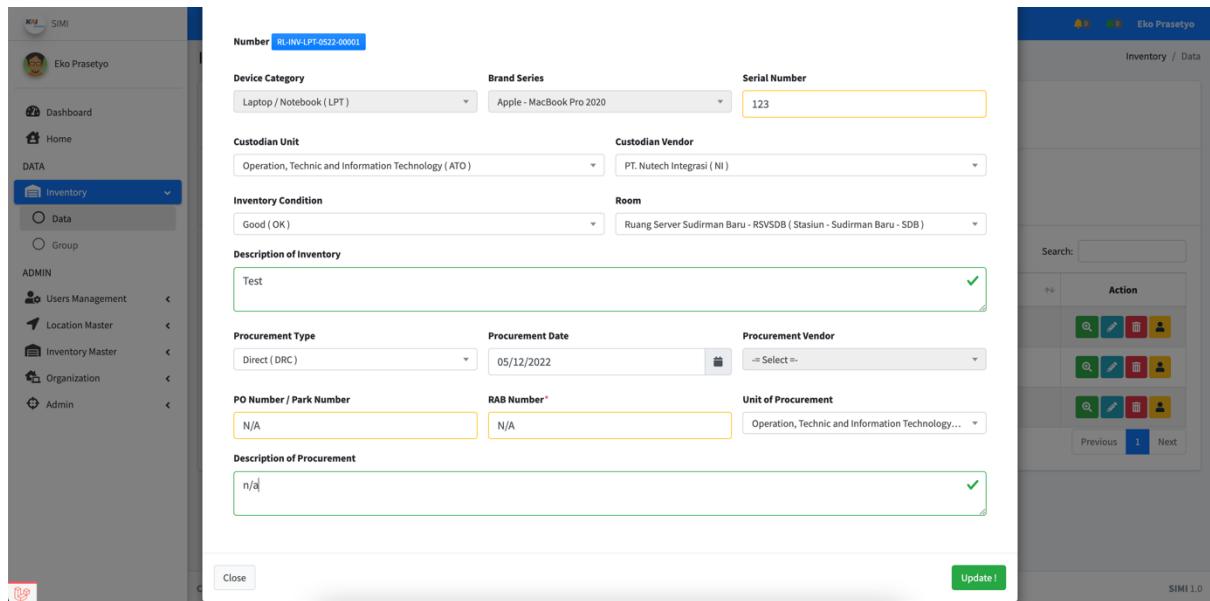
terdapat tombol Cancel untuk membatalkan dan tombol Save untuk menyimpan data-data yang sudah diisi pada field tersebut.



Gambar 48 Implementasi Antarmuka – Halaman Tambah Data Inventaris

4.1.2.10 Halaman Edit Data Inventaris

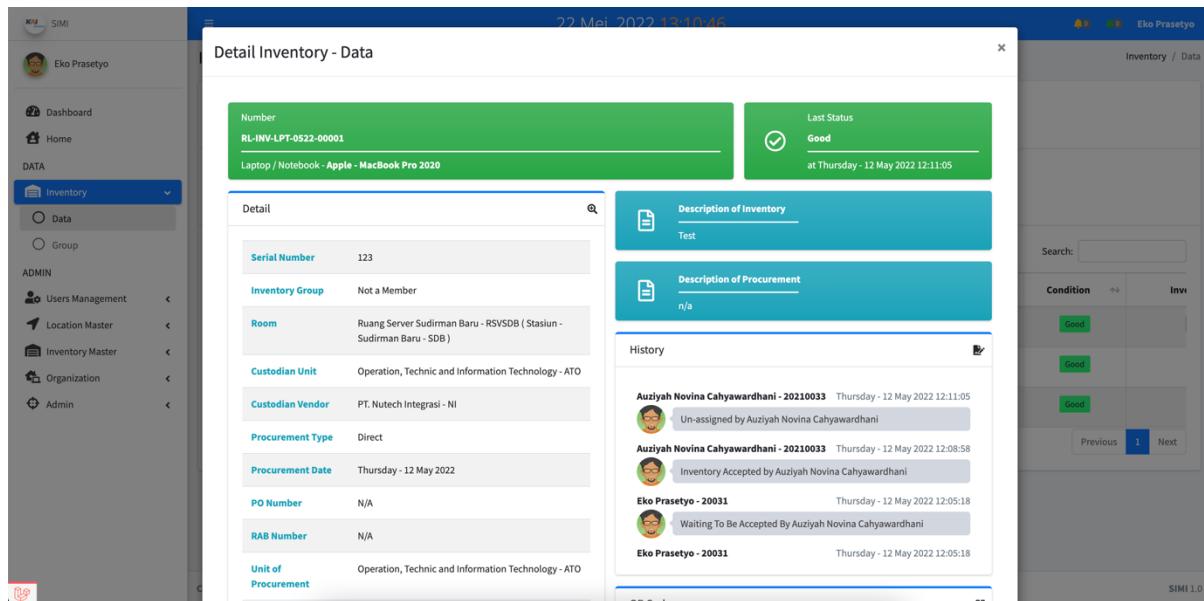
Pada halaman edit data inventaris, terdapat balok dengan nomor inventaris yang sudah terdaftar, kemudian field-field yang sudah terisi berdasarkan data yang sebelumnya tersimpan yaitu Device Category, Brand Series, Serial Number, Custodian Unit, Custodian Vendor, Inventory Condition, Room, Description of Inventory, Procurement Type, Procurement Date, Procurement Vendor, PO Number / Park Number, RAB Number, Unit of Procurement dan Description of Procurement. Kemudian terdapat tombol Cancel untuk membatalkan dan tombol Update untuk melakukan perubahan data-data yang sudah diisi pada field tersebut.



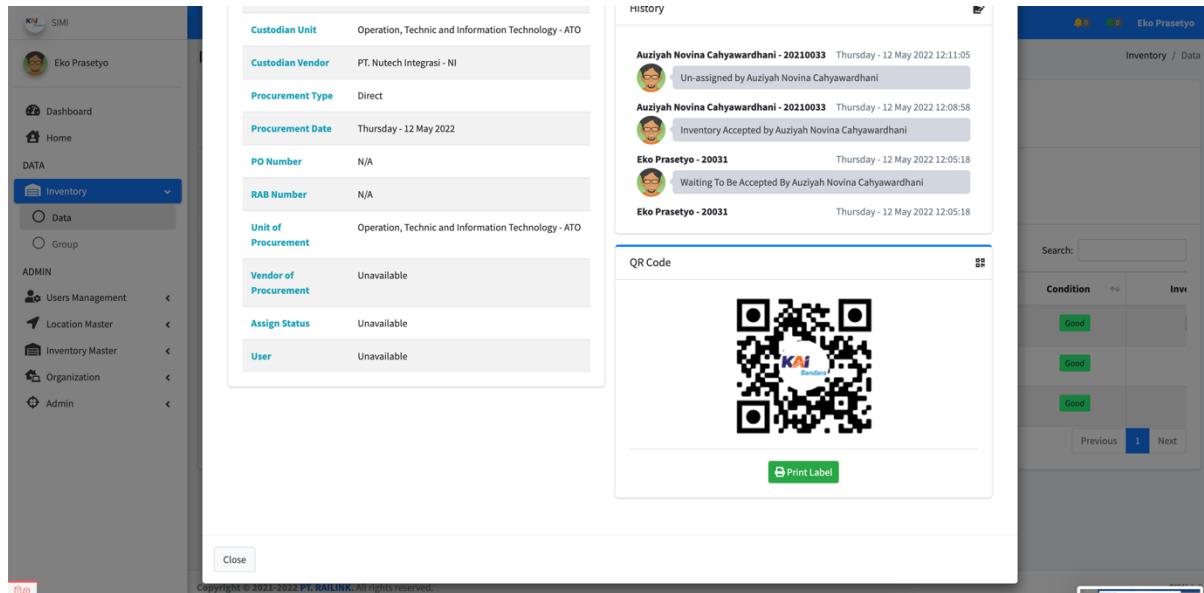
Gambar 49 Implementasi Antarmuka – Halaman Edit Data Inventaris

4.1.2.11 Halaman Detail Data Inventaris

Pada halaman detail data inventaris, terdapat balok di sebelah kiri dengan nomor inventaris yang sudah terdaftar, tipe inventaris dan merk inventaris, kemudian pada balok sebelah kanan menginformasikan kondisi terakhir inventaris beserta tanggalnya, kemudian pada tabel sebelah kiri terdapat baris-baris yang menjelaskan detail dari inventaris. Kemudian pada tabel History menjelaskan riwayat apa saja yang sudah dilakukan terhadap inventaris. Kemudian di bawah tabel History terdapat kolom QR Code yang berisi nomor inventaris yang disertai dengan tombol Print Label untuk mencetak label inventaris tersebut. Kemudian terdapat tombol Close untuk keluar dari jendela halaman detail data.



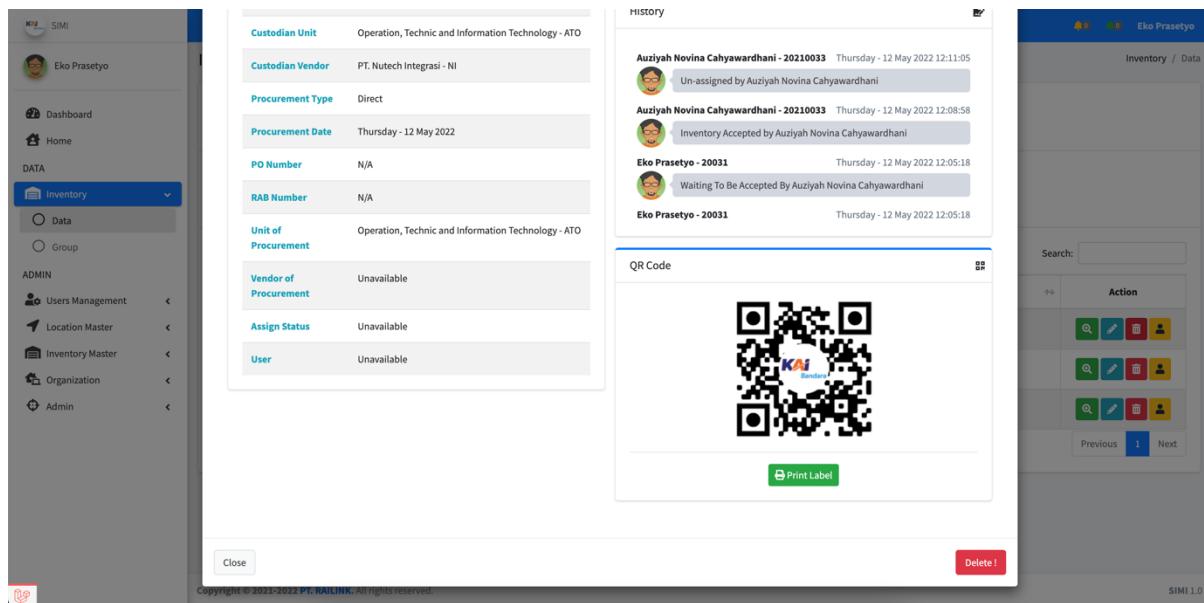
Gambar 50 Implementasi Antarmuka – Halaman Detail Data Inventaris 1



Gambar 51 Implementasi Antarmuka – Halaman Detail Data Inventaris 2

4.1.2.12 Halaman Hapus Data Inventaris

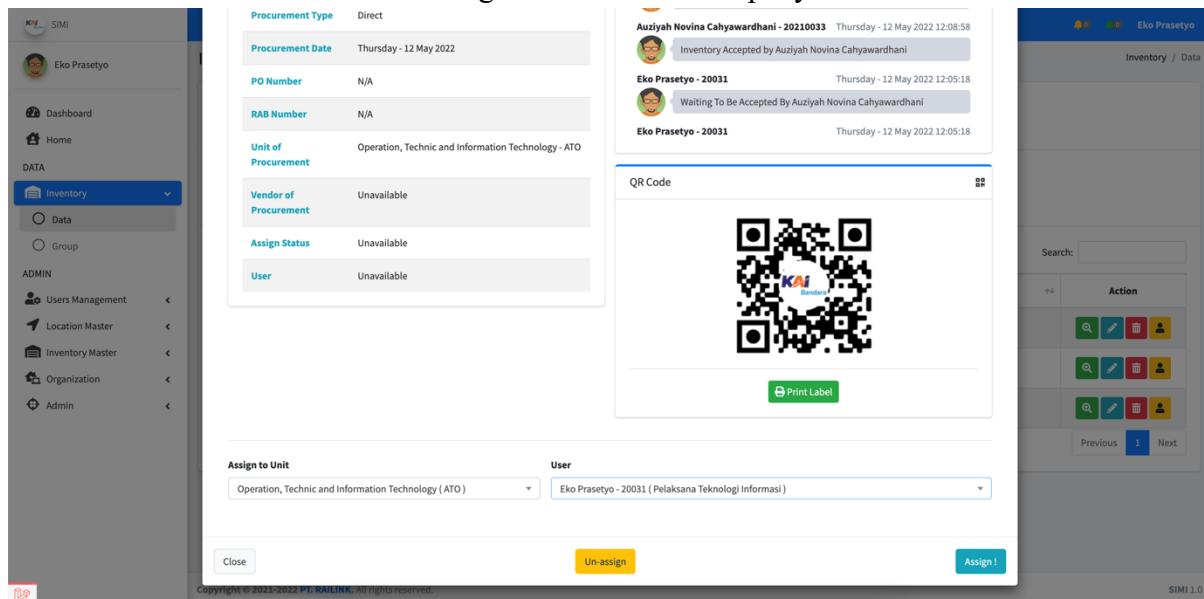
Pada halaman hapus data inventaris, tampilannya serupa dengan detail data inventaris, kemudian pada bagian bawah jendela terdapat 2 tombol yaitu Close untuk menutup jendela dan Delete untuk melakukan hapus data inventaris.



Gambar 52 Implementasi Antarmuka – Halaman Detail Data Inventaris 2

4.1.2.13 Halaman Penyerahan Inventaris

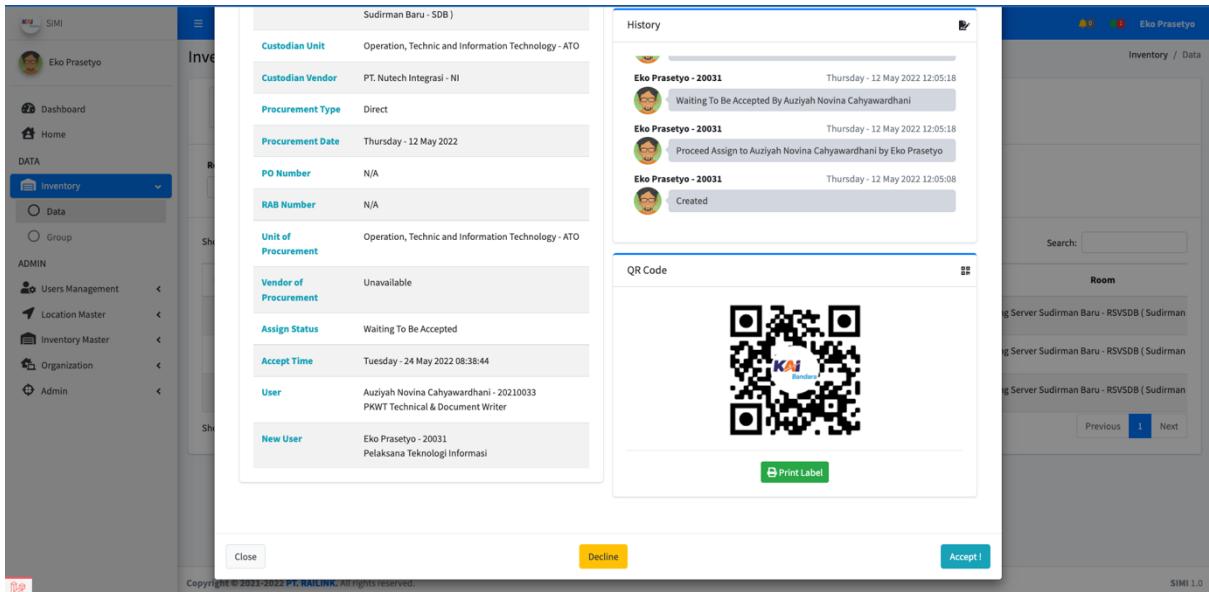
Pada halaman penyerahan inventaris, tampilannya serupa dengan detail data inventaris, kemudian pada bagian bawah 2 dropdown yaitu Unit dan User yang selanjutnya user tersebut akan ditunjuk sebagai pemegang inventaris yang akan diserahkan, kemudian di bagian bawah jendela terdapat 3 tombol yaitu Close untuk menutup jendela, Un-assign untuk pengembalian inventaris dan Assign untuk melakukan penyerahan inventaris.



Gambar 53 Implementasi Antarmuka – Halaman Penyerahan Inventaris

4.1.2.14 Halaman Penerimaan Inventaris

Pada halaman penerimaan inventaris, tampilannya serupa dengan detail data inventaris, kemudian di bagian bawah jendela terdapat 3 tombol yaitu Close untuk menutup jendela, Decline untuk menolak penyerahan inventaris dan Accept untuk menerima penyerahan inventaris.



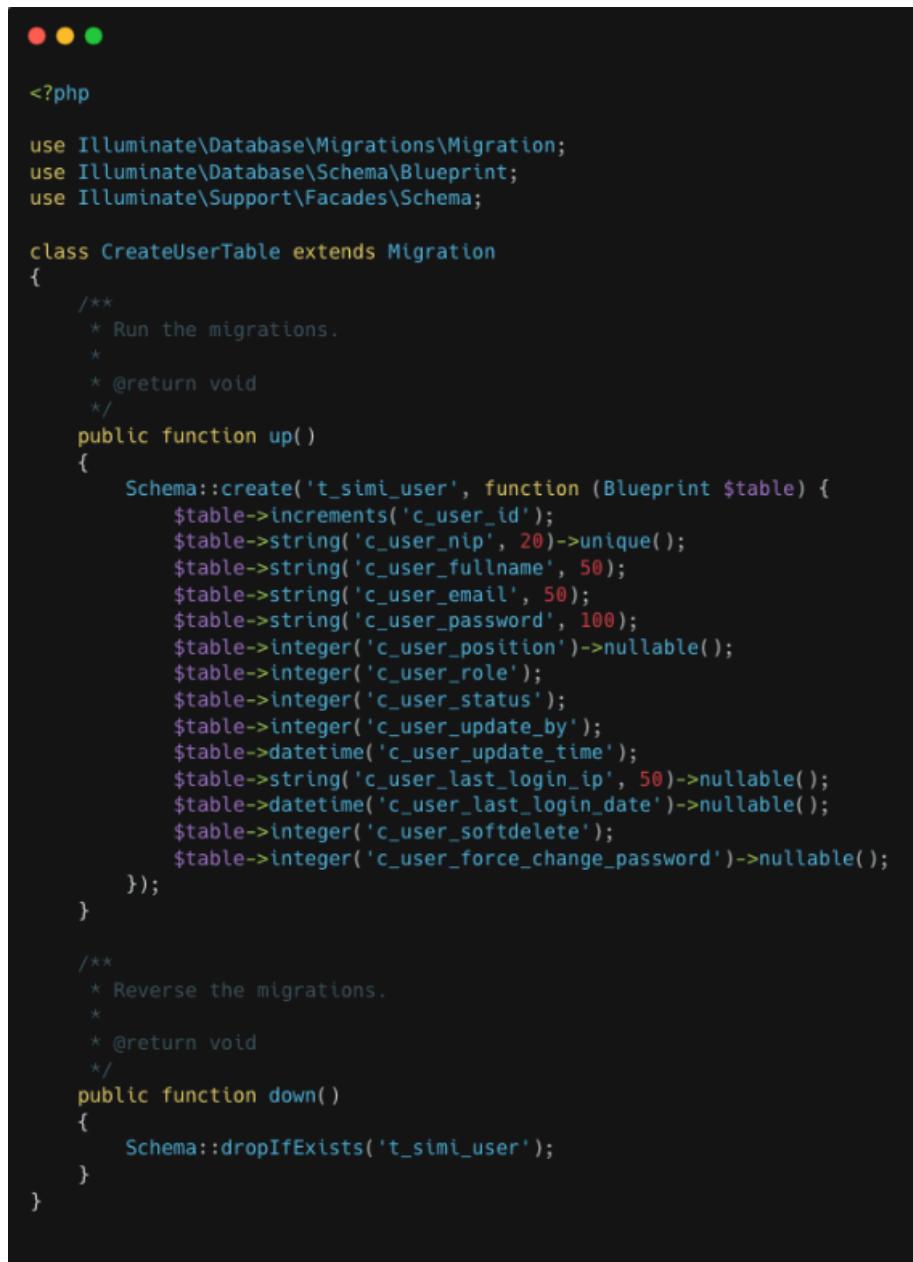
Gambar 54 Implementasi Antarmuka – Halaman Penerimaan Inventaris

4.1.3 Implementasi Database

Untuk implementasi database, Laravel memiliki fitur yang memudahkan kita sebagai developer yang bernama Artisan. Artisan sendiri adalah sebuah perintah Command Line Interface (CLI) berbasis PHP yang salah satunya dapat menangani semua hal tentang database yang dalam konteks ini memiliki nama Migration. Migration dalam Laravel berfungsi seperti version control untuk database, selain itu juga dapat mengelola dan memodifikasi skema database yang dikembangkan.

Dalam implementasi database Sistem Informasi Manajemen Inventaris, fitur dari Laravel yang digunakan untuk mengelola database yaitu, Migration dan Seeder. Migration digunakan untuk mengelola skema database, sedangkan Seeder digunakan untuk memasukkan data-data awal ke database saat sistem digunakan pertama kali.

Berikut adalah penggalan kode untuk Migration dari Class CreateUsersTable untuk membuat tabel User ke database.



```

<?php

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

class CreateUserTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('t_simi_user', function (Blueprint $table) {
            $table->increments('c_user_id');
            $table->string('c_user_nip', 20)->unique();
            $table->string('c_user_fullname', 50);
            $table->string('c_user_email', 50);
            $table->string('c_user_password', 100);
            $table->integer('c_user_position')->nullable();
            $table->integer('c_user_role');
            $table->integer('c_user_status');
            $table->integer('c_user_update_by');
            $table->datetime('c_user_update_time');
            $table->string('c_user_last_login_ip', 50)->nullable();
            $table->datetime('c_user_last_login_date')->nullable();
            $table->integer('c_user_softdelete');
            $table->integer('c_user_force_change_password')->nullable();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('t_simi_user');
    }
}

```

Gambar 55 Implementasi Database – Migration Membuat Tabel User

Setelah semua file Migration siap, maka kita gunakan perintah Artisan untuk menjalankan Migration tersebut dengan ketik perintah pada CLI :

php artisan migrate

Berikut adalah gambar perintah tersebut ketika dijalankan.

```

Migrating: 2022_01_01_000000_create_inventory_condition_table
Migrated: 2022_01_01_000000_create_inventory_condition_table (25.71ms)
Migrating: 2022_01_01_000000_create_inventory_group_category_table
Migrated: 2022_01_01_000000_create_inventory_group_category_table (9.02ms)
Migrating: 2022_01_01_000000_create_inventory_group_table
Migrated: 2022_01_01_000000_create_inventory_group_table (12.44ms)
Migrating: 2022_01_01_000000_create_inventory_table
Migrated: 2022_01_01_000000_create_inventory_table (13.23ms)
Migrating: 2022_01_01_000000_create_jobs_table
Migrated: 2022_01_01_000000_create_jobs_table (25.94ms)
Migrating: 2022_01_01_000000_create_last_inventory_number_table
Migrated: 2022_01_01_000000_create_last_inventory_number_table (10.30ms)
Migrating: 2022_01_01_000000_create_location_table
Migrated: 2022_01_01_000000_create_location_table (31.77ms)
Migrating: 2022_01_01_000000_create_location_type_table
Migrated: 2022_01_01_000000_create_location_type_table (8.89ms)
Migrating: 2022_01_01_000000_create_permission_table
Migrated: 2022_01_01_000000_create_permission_table (36.23ms)
Migrating: 2022_01_01_000000_create_position_level_table
Migrated: 2022_01_01_000000_create_position_level_table (6.98ms)
Migrating: 2022_01_01_000000_create_position_table
Migrated: 2022_01_01_000000_create_position_table (24.98ms)
Migrating: 2022_01_01_000000_create_procurement_type_table
Migrated: 2022_01_01_000000_create_procurement_type_table (14.68ms)
Migrating: 2022_01_01_000000_create_role_permission_table
Migrated: 2022_01_01_000000_create_role_permission_table (12.00ms)
Migrating: 2022_01_01_000000_create_role_table
Migrated: 2022_01_01_000000_create_role_table (40.58ms)
Migrating: 2022_01_01_000000_create_room_table
Migrated: 2022_01_01_000000_create_room_table (27.40ms)
Migrating: 2022_01_01_000000_create_unit_table
Migrated: 2022_01_01_000000_create_unit_table (23.52ms)
Migrating: 2022_01_01_000000_create_user_table
Migrated: 2022_01_01_000000_create_user_table (29.02ms)

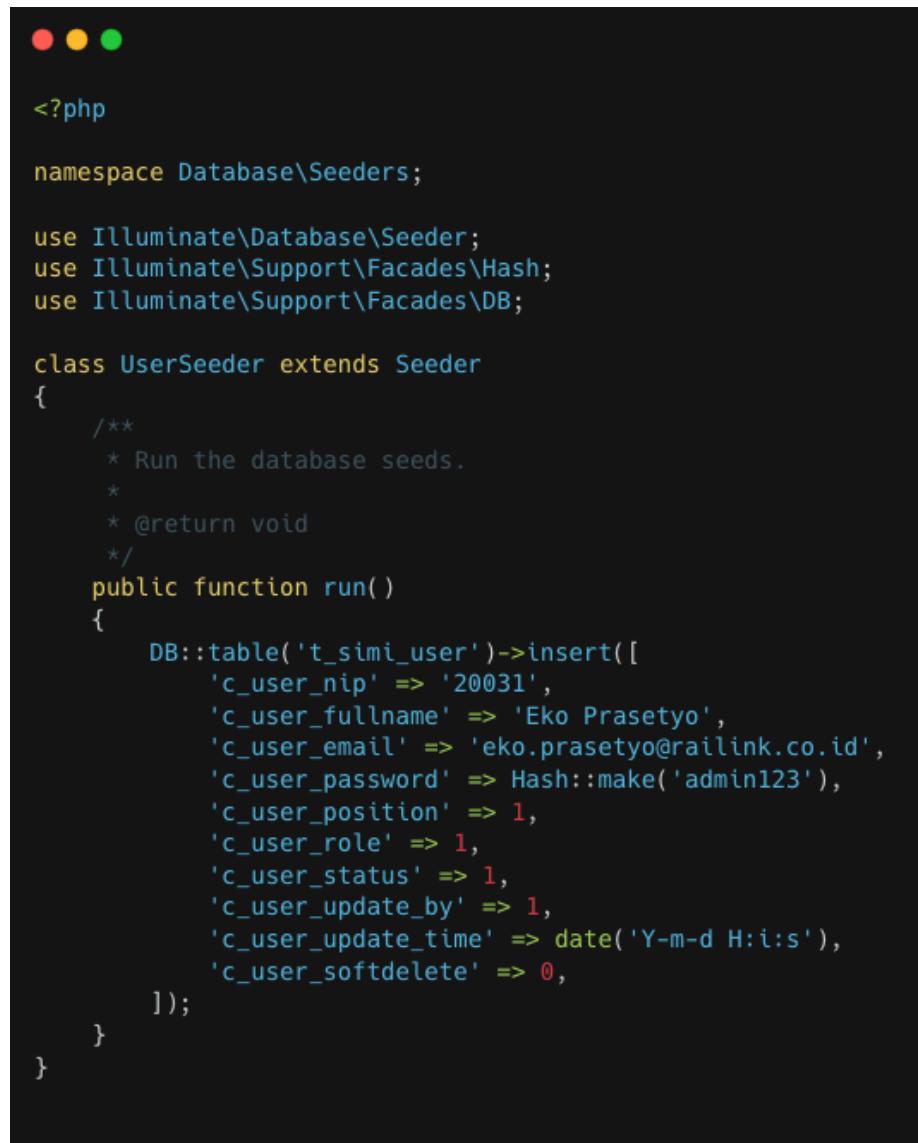
➔ ~/Projects/simi-web ➔ on ⬤ master !2 ➔

```

Gambar 56 Implementasi Database – Menjalankan Perintah Migration

Setelah Migration dijalankan, selanjutnya membuat file Seeder untuk memasukkan data-data awal ke database.

Berikut adalah penggalan kode untuk Seeder dari Class UserSeeder untuk memasukkan 1 data ke database.



```
<?php

namespace Database\Seeders;

use Illuminate\Database\Seeder;
use Illuminate\Support\Facades\Hash;
use Illuminate\Support\Facades\DB;

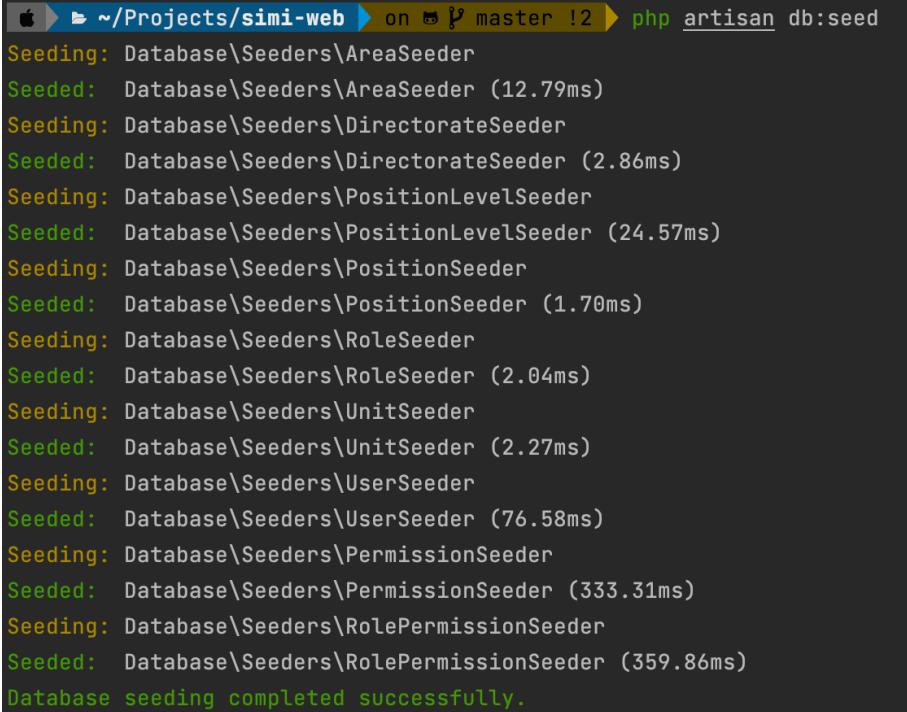
class UserSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        DB::table('t_simi_user')->insert([
            'c_user_nip' => '20031',
            'c_user_fullname' => 'Eko Prasetyo',
            'c_user_email' => 'eko.prasetyo@railink.co.id',
            'c_user_password' => Hash::make('admin123'),
            'c_user_position' => 1,
            'c_user_role' => 1,
            'c_user_status' => 1,
            'c_user_update_by' => 1,
            'c_user_update_time' => date('Y-m-d H:i:s'),
            'c_user_softdelete' => 0,
        ]);
    }
}
```

Gambar 57 Implementasi Database – Seeder Memasukkan Data Ke Tabel User

Setelah semua file Seeder siap, maka kita gunakan perintah Artisan untuk menjalankan Seeder tersebut dengan ketik perintah pada CLI :

php artisan db:seed

Berikut adalah gambar perintah tersebut ketika dijalankan.



```

~/Projects/simi-web on master !2 php artisan db:seed
Seeding: Database\Seeders\AreaSeeder
Seeded: Database\Seeders\AreaSeeder (12.79ms)
Seeding: Database\Seeders\DirectorateSeeder
Seeded: Database\Seeders\DirectorateSeeder (2.86ms)
Seeding: Database\Seeders\PositionLevelSeeder
Seeded: Database\Seeders\PositionLevelSeeder (24.57ms)
Seeding: Database\Seeders\PositionSeeder
Seeded: Database\Seeders\PositionSeeder (1.70ms)
Seeding: Database\Seeders\RoleSeeder
Seeded: Database\Seeders\RoleSeeder (2.04ms)
Seeding: Database\Seeders\UnitSeeder
Seeded: Database\Seeders\UnitSeeder (2.27ms)
Seeding: Database\Seeders\UserSeeder
Seeded: Database\Seeders\UserSeeder (76.58ms)
Seeding: Database\Seeders\PermissionSeeder
Seeded: Database\Seeders\PermissionSeeder (333.31ms)
Seeding: Database\Seeders\RolePermissionSeeder
Seeded: Database\Seeders\RolePermissionSeeder (359.86ms)
Database seeding completed successfully.

```

Gambar 58 Implementasi Database – Menjalankan Perintah Seeder

4.2 Pengujian Sistem

Pengujian sistem merupakan proses pengeksekusian sistem perangkat lunak untuk menentukan apakah sistem tersebut sesuai dengan spesifikasi sistem dan berjalan di lingkungan yang diinginkan. Pengujian sistem sering diasosiasikan dengan pencarian bug, ketidak sempurnaan program, kesalahan pada program yang menyebabkan kegagalan pada eksekusi sistem perangkat lunak.

Dalam pengujian Sistem Informasi Manajemen Inventaris studi kasus di PT. Railink menggunakan teknik Pengujian Black Box karena memiliki kelebihan diantaranya sebagai berikut :

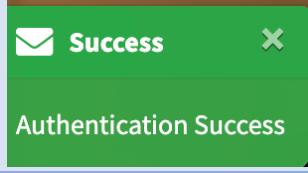
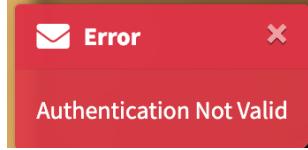
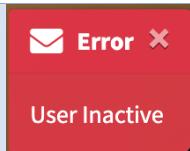
- Penguji tidak perlu memiliki pengetahuan tentang bahasa pemrograman tertentu.
- Pengujian yang dilakukan berdasarkan sudut pandang user agar dapat mengungkapkan inkonsistensi atau ambiguitas dalam spesifikasi.
- Programmer dan tester memiliki ketergantungan satu sama lain [12].

Pengujian Black Box akan digunakan untuk melakukan pengecekan detail pada program antara input dan output. Hal tersebut dilakukan untuk memastikan fungsi-fungsi sistem berjalan sesuai dengan yang dirancang. Batasan pengujian Black Box dilakukan langsung melalui halaman web dari Sistem Informasi Manajemen Inventaris yang mencakup fungsi-fungsi sebagai berikut :

- Proses Login User
- Proses Input Data Inventaris
- Proses Tampil Data Inventaris

- d. Proses Edit Data Inventaris
- e. Proses Penyerahan Inventaris
- f. Proses Penerimaan Inventaris
- g. Proses Logout User

4.2.1 Pengujian Black Box Proses Login User

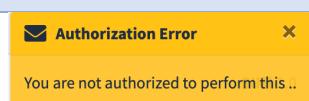
No	Kasus Uji	Hasil Yang Diharapkan	Gambar	Keterangan
1	Submit dengan data user yang valid	Muncul pesan “Authentication Success” dan diarahkan ke halaman Home		Sesuai
2	Submit dengan data user yang tidak valid	Muncul pesan “Authentication Not Valid”		Sesuai
3	Submit dengan data user yang sudah tidak aktif	Muncul pesan “User Inactive”		Sesuai

4.2.2 Pengujian Black Box Proses Input Data Inventaris

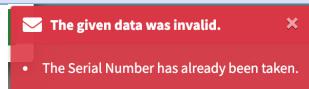
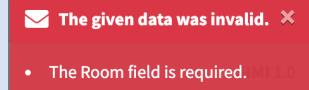
No	Kasus Uji	Hasil Yang Diharapkan	Gambar	Keterangan
1	Submit dengan data inventaris yang valid	Muncul pesan “Data Monitor / LED / LCD with Number RL-INV-MON-0522-0001 has been added ..”		Sesuai
2	Submit dengan serial number yang sudah terdaftar	Muncul pesan “The Serial Number has already been taken”		Sesuai
3	Submit dengan salah satu field kosong – contoh Condition	Muncul pesan “The Condition field is required”		Sesuai

4.2.3 Pengujian Black Box Proses Tampil Data Inventaris

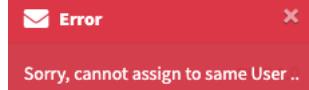
No	Kasus Uji	Hasil Yang Diharapkan	Gambar	Keterangan
----	-----------	-----------------------	--------	------------

1	Akses halaman Tampil Data Inventaris dengan permission yang valid	Muncul halaman List Data Inventaris		Sesuai
2	Akses halaman Tampil Data Inventaris dengan permission yang tidak valid	Muncul pesan “You are not authorized to perform this” dan diarahkan ke halaman Home		Sesuai

4.2.4 Pengujian Black Box Proses Edit Data Inventaris

No	Kasus Uji	Hasil Yang Diharapkan	Gambar	Keterangan
1	Submit dengan data inventaris yang valid	Muncul pesan “Data Monitor / LED / LCD with Number RL-INV-MON-0522-0001 has been updated ..”		Sesuai
2	Submit dengan serial number yang sudah terdaftar	Muncul pesan “The Serial Number has already been taken”		Sesuai
3	Submit dengan salah satu field kosong – contoh Room	Muncul pesan “The Room field is required”		Sesuai

4.2.5 Pengujian Black Box Proses Penyerahan Inventaris

No	Kasus Uji	Hasil Yang Diharapkan	Gambar	Keterangan
1	Submit dengan tujuan pengguna yang valid.	Muncul pesan “Data Inventory with Number RL-INV-LPT-0522-00001 proceed assigned to Eko Prasetyo..”		Sesuai
2	Submit dengan data pengguna yang sama	Muncul pesan “Sorry, cannot assign to same User”		Sesuai

dengan data
pengguna
sebelumnya

3	Submit dengan status Inventaris masih menunggu penerimaan	Muncul pesan “Sorry, This Inventory is Waiting For Acceptance”		Sesuai
4	Submit dengan status Inventaris adalah bagian dari Grup Inventaris	Muncul Pesan “Sorry, this inventory is a member of the Inventory Group”		Sesuai

4.2.6 Pengujian Black Box Proses Penerimaan Inventaris

No	Kasus Uji	Hasil Yang Diharapkan	Gambar	Keterangan
1	Tekan tombol Accept.	Muncul pesan “Data Inventory with Number RL-INV-LPT-0522-00001 has been assigned to Auziyah Novina Cahyawarhani..”		Sesuai
2	Tekan tombol Decline	Muncul pesan “Data Inventory with Number RL-INV-LPT-0522-00001 has been declined”		Sesuai

4.2.7 Pengujian Black Box Proses Logout User

No	Kasus Uji	Hasil Yang Diharapkan	Gambar	Keterangan
1	Tekan tombol Logout	Kembali ke halaman Login		Sesuai