

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem

Aplikasi mobile yang telah dikembangkan menggunakan pendekatan metode waterfall dan berbasis Android, didesain dengan menggunakan framework Flutter. Sistem ini menjadikan Cloud Firebase sebagai tempat penyimpanan database, memastikan aksesibilitas dari mana saja sehingga dapat meningkatkan efisiensi waktu dalam manajemen aset.

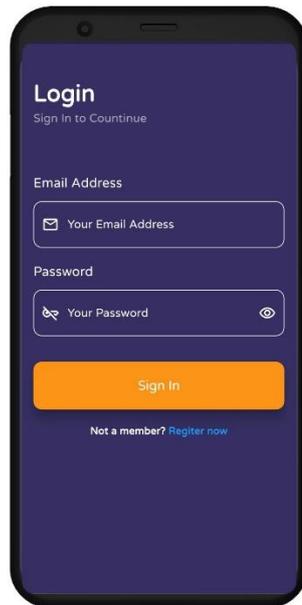
Keunikan dari aplikasi ini terletak pada penerapan teknologi QR Code untuk memberikan identifikasi yang unik pada setiap aset. Pendekatan ini tidak hanya mempermudah proses pelacakan aset, namun juga memberikan tingkat akurasi yang lebih tinggi. Dengan demikian, aplikasi ini menjadi solusi yang inovatif dalam mengoptimalkan manajemen aset, memberikan kemudahan akses, serta memastikan keakuratan dan keefisienan dalam pengelolaan aset secara keseluruhan.

1. Antarmuka Pengembangan Sistem

Berikut ini merupakan pembahasan dan hasil implementasi desain interface antarmuka dari sistem yang telah dibuat :

a. Tampilan *Login Screen*

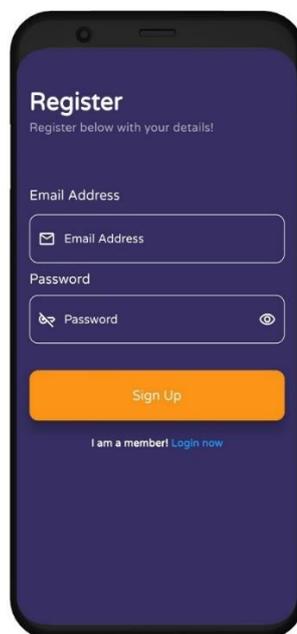
Halaman *Login* ditampilkan pada awal pengguna membuka aplikasi. Terdapat dua *textfield* diantaranya yaitu untuk inputan *email address* dan inputan *password*. *Button sign in* memiliki validasi dimana jika *password* dan *email address* salah maka sistem akan menampilkan peringatan kesalahan *error*. Berikut tampilan yang dapat dilihat pada gambar di bawah ini :



Gambar 4. 1 *Login Screen*

b. Tampilan *Register*

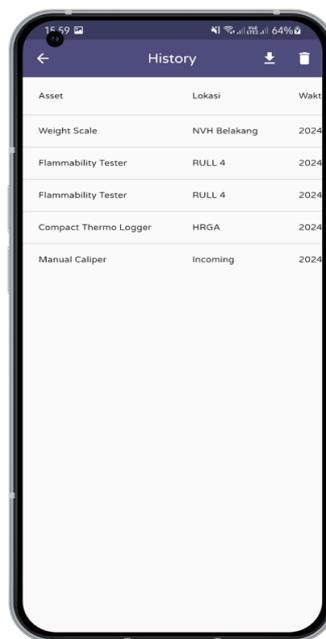
Halaman *Register* merupakan halaman untuk pengguna mendaftarkan email agar pengguna bisa masuk ke dalam sistem. Berikut tampilan yang dapat dilihat pada gambar di bawah ini :



Gambar 4. 2 *Register Screen*

c. Tampilan *History*

Tampilan *History* menampilkan informasi dengan detail sebagai berikut: alamat email pengguna yang terkait, nama aset, lokasi terakhir aset berada, dan timestamp waktu terjadinya pemindahan atau perubahan status. Dengan tampilan ini, pengguna dapat dengan mudah melacak sejarah pemindahan aset, mengetahui siapa yang bertanggung jawab, lokasi terakhir aset, serta waktu kejadian. Berikut tampilan yang dapat dilihat pada gambar dibawah ini :



Gambar 4. 3 *History Screen*

d. Tampilan *Dashboard*

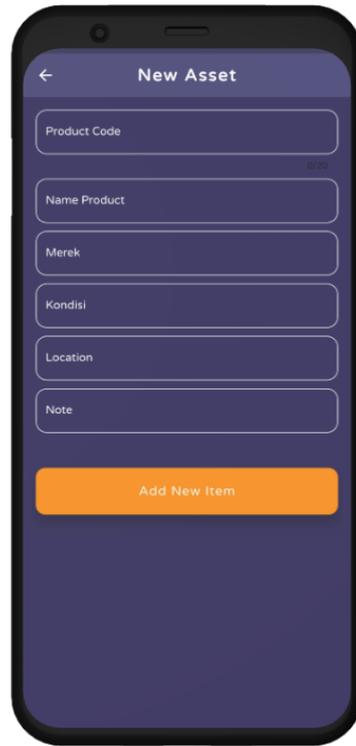
Halaman *Dashboard* memiliki data aset dengan informasi tentang barcode untuk identifikasi unik, address untuk lokasi fisik, dan name untuk nama aset. Berikut tampilan yang dapat dilihat pada gambar dibawah ini :



Gambar 4. 4 *Dashboard Screen*

e. Tampilan *New Asset*

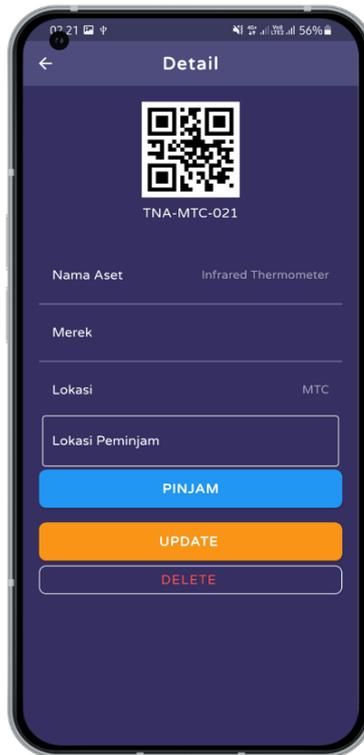
Halaman *New Asset* menyediakan kolom input untuk memasukkan informasi aset baru, termasuk kode aset, nama produk, merek, kondisi, lokasi, dan catatan tambahan. Admin dapat mengisi data yang diperlukan untuk menambahkan aset ke dalam sistem manajemen aset. Berikut tampilan yang dapat dilihat pada gambar dibawah ini :



Gambar 4. 5 Halaman *New Asset*

f. Tampilan *Detail Update*

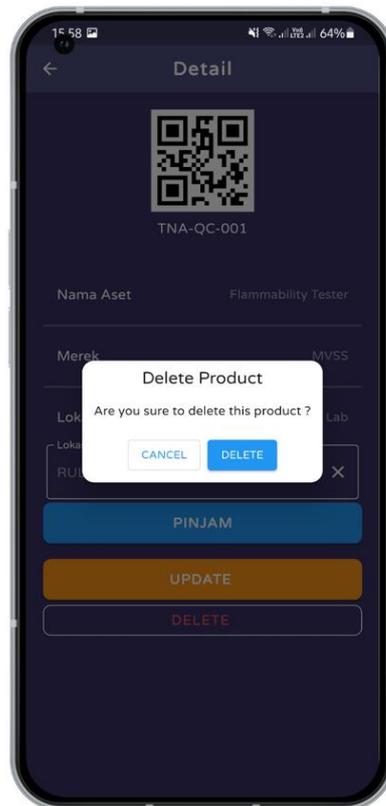
Halaman detail menyediakan kolom untuk mengubah beberapa informasi, dan terdapat alert jika tombol update di tekan untuk memberikan informasi jika perubahan berhasil. Berikut tampilan yang dapat dilihat pada gambar dibawah ini :



Gambar 4. 6 Halaman *Update*

g. Tampilan *Detail Delete*

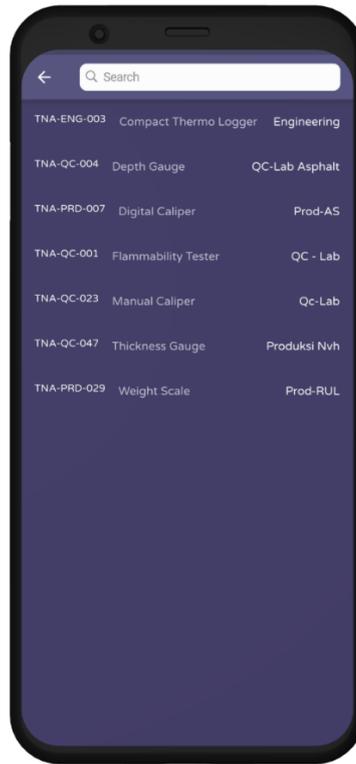
Halaman Detail juga mempunyai tombol delete yang berfungsi menghapus data yang ada dalam sistem. Berikut tampilan yang dapat dilihat pada gambar dibawah ini :



Gambar 4. 7 Tampilan *Delete*

h. Tampilan Search

Halaman search atau pencarian termasuk fitur pada sistem yang penulis kembangkan. Pada halaman search terdapat kolom untuk mencari data untuk memperoleh informasi dengan cepat. Fitur ini akan sangat efisien jika data dalam sistem sudah terlalu banyak. Berikut tampilan yang dapat dilihat pada gambar dibawah ini :



Gambar 4. 8 Tampilan *Search*

i. Tampilan *QR Code Scanner*

Halaman Scan terdapat fungsi utama yaitu mendeteksi *QR Code* yang tertera pada aset yang telah terdaftar pada sistem untuk memperoleh informasi yang diinginkan. Ada tiga fungsi pada halaman scan yaitu kamera yang bisa di ganti depan maupun belakang, flash light untuk menyalakan lampu kamera untuk memberi cahaya pada tempat yang gelap, dan cancel untuk membatalkan. Berikut tampilan yang dapat dilihat pada gambar dibawah ini :



Gambar 4.9 Tampilan Scan QR Code

2. Pengkodean Sistem

Pada tahap ini penulis melakukan aktivitas penulisan kode program untuk aplikasi. Contoh kode program tersebut sebagai berikut :

```

1
2 void main() async {
3   // proses inialisasi
4   WidgetsFlutterBinding.ensureInitialized();
5   await Firebase.initializeApp(
6     options: DefaultFirebaseOptions.currentPlatform,
7   );
8
9   // injeksi data dari auth controller berupa user id dan set permanen untuk semua page
10  Get.put(AuthController(), permanent: true);
11
12  runApp(MyApp());
13 }
14
15 class MyApp extends StatelessWidget {
16   MyApp({super.key});
17
18   // meng inialisasi firebase auth
19   final FirebaseAuth auth = FirebaseAuth.instance;
20
21   /* firebase auth dengan streambuilder ( import package firebase_auth ) ->
22   auto login dengan firebase authentication */
23   @override
24   Widget build(BuildContext context) {
25     return StreamBuilder<User?>({
26       // memantau setiap perubahan dari otentikasi
27       stream: auth.authStateChanges(),
28       builder: (context, snapAuth) {
29         // handle status if connection waiting -> LoginScreen(berupa Loading progres indikator)
30         if (snapAuth.connectionState == ConnectionState.waiting) {
31           return const LoadingView();
32         }
33
34         return GetMaterialApp(
35           debugShowCheckedModeBanner: false,
36           title: "QR Code",
37           theme: ThemeData(
38             textTheme: GoogleFonts.varelaRoundTextTheme(
39               Theme.of(context).textTheme,
40             ),
41           useMaterial3: false,
42         );
43         // cek status snapAuth yang berupa user jika memiliki data -> home ! jika tidak -> login
44         initialRoute: snapAuth.hasData ? Routes.home : Routes.login,
45         getPages: AppPages.routes,
46       });
47     },
48   );

```

Gambar 4.10 Code Main

4.2 Hasil Pengujian

Penelitian ini menggunakan metode Black Box Testing untuk menguji fungsionalitas sistem. Pengujian dilakukan dengan membuat tabel yang memuat deskripsi pengujian, fitur yang diuji, parameter kesuksesan, hasil pengujian, dan kesimpulan. Fokus pengujian mencakup berbagai fitur dalam sistem, seperti halaman Login, *Dashboard* admin dan pengguna, halaman pencarian, halaman detail, halaman aset baru dan *QR Code*. Tujuannya adalah memastikan sistem beroperasi dengan benar sesuai dengan kriteria keberhasilan yang telah ditetapkan. Berikut kerangka dasar tabel yang digunakan dalam pengujian :

Tabel 4. 1 Tabel Kerangka Dasar Pengujian Black Box

No	Test Case	Pengujian Kasus	Hasil Yang Diharapkan	Hasil Pengujian
..
..
..
..

Penjelasan :

Test Case : Fungsionalitas fitur sistem apapun yang dapat diuji.

Pengujian Kasus : Tindakan Penguji terhadap suatu fungsi dari sistem.

Hasil yang Diharapkan : Hasil yang diharapkan dari pengujian.

Hasil Pengujian : Apakah pengujian *Valid* ataupun Tidak

1. Halaman *Register*

Tabel 4. 2 Pengujian Halaman *Register*

No	Test Case	Pengujian Kasus	Hasil Yang diharapkan	Hasil Pengujian
1	<i>Email</i> dan <i>password</i> tidak diisi kemudian klik tombol <i>Sign Up</i>	<i>Email</i> : (kosong) <i>Password</i> : (Kosong)	Sistem akan menampilkan pesan “Email dan Password wajib di isi”	<i>Valid</i>

2	Mendaftarkan <i>Email</i> yang sudah ada kemudian klik tombol <i>Sign Up</i>	<i>Email</i> : admin@gmail.com <i>Password</i> : 12345678	Sistem akan menampilkan pesan “ Error. The email address is already in use ”	<i>Valid</i>
4	<i>Email</i> diisi dan <i>password</i> diisi dengan benar kemudian klik tombol Sign Up	<i>Email</i> : admin@gmail.com <i>Password</i> : admin123	Sistem menyimpan informasi <i>Register</i> dan kemudian menampilkan <i>Dashboard</i> user	<i>Valid</i>
5	Pengujian Icon Button Eye pada kolom password	Klik Icon Button Eye	Menampilkan password yang dapat terbaca	<i>Valid</i>

2. Halaman Login

Tabel 4. 3 Pengujian Halaman *Login*

No	Test Case	Pengujian Kasus	Hasil Yang diharapkan	Hasil Pengujian
1	<i>Email</i> dan <i>password</i> tidak diisi kemudian klik tombol Sign In	<i>Email</i> : (kosong) <i>Password</i> : (Kosong)	Sistem akan menampilkan pesan “Email dan Password wajib di isi”	<i>Valid</i>
2	<i>Email</i> benar dan <i>password</i> salah kemudian klik tombol Sign In	<i>Email</i> : admin@gmail.com <i>Password</i> : 12345678	Sistem akan menampilkan pesan “the password is invalid or the user does not have a password”	<i>Valid</i>
3	<i>Email</i> benar dan <i>password</i> salah kemudian klik tombol Sign In	<i>Email</i> : user@gmail.com <i>Password</i> : 12345678	Sistem akan menampilkan pesan “there is no user record corresponding to this identifier. the user may	<i>Valid</i>

			have been <i>deleted</i> ”	
4	<i>Email</i> diisi dan <i>password</i> diisi dengan benar kemudian klik tombol Sign In	<i>Email</i> : admin@gmail.com <i>Password</i> : admin123	Sistem Menerima akses <i>Login</i> dan kemudian menampilkan <i>Dashboard</i> user	<i>Valid</i>
5	Pengujian Icon Button Eye pada kolom password	Klik Icon Button Eye	Menampilkan password yang dapat terbaca	<i>Valid</i>

3. Halaman Dashboard

Tabel 4. 4 Pengujian Halaman *Dashboard*

No	Test Case	Pengujian Kasus	Hasil Yang Diharapkan	Hasil Pengujian
1	Halaman <i>Dashboard</i>	Masuk dengan Akun yang terdaftar	Menampilkan Halaman <i>Dashboard</i> dengan menu list data aset ,pencarian aset, tambah aset, <i>scan QR Code</i> , tombol log out	<i>Valid</i>
2	Tambah aset	Klik icon (+)	Menampilkan halaman <i>new asset</i>	<i>Valid</i>
3	<i>Logout</i> sistem Akun	Klik Tombol <i>Logout</i>	Pengguna Keluar dari halaman <i>Dashboard</i> dengan validasi dan Menampilkan halaman <i>Login</i>	<i>Valid</i>
4	Pencarian aset	Klik icon <i>search</i>	Menampilkan halaman pencarian	<i>Valid</i>
5	<i>Scan QR Code</i>	Klik <i>Floating action button</i> dengan icon <i>QR Code</i>	Menampilkan halaman kamera <i>scan QR Code</i>	<i>Valid</i>

4. Halaman Menu Detail

Tabel 4. 5 Pengujian Halaman *Detail*

No	Test case	Pengujian Kasus	Hasil Yang Diharapkan	Hasil Pengujian
1	Halaman <i>Detail</i>	Klik <i>card list data</i> pada <i>Dashboard</i>	Menampilkan halaman <i>Detail</i> dengan elemen <i>QR Code</i> , <i>textfield</i> , dan <i>button update delete</i>	<i>Valid</i>
2	Mengubah isi <i>textfield</i> atau <i>update data</i>	Mengubah data aset dan klik tombol <i>update</i>	Menampilkan <i>alert</i> “berhasil update aset” jika berhasil.	<i>Valid</i>
3	Pengujian hapus data aset	Klik tombol <i>delete</i>	Menampilkan alert dialog dan alert “berhasil <i>delete</i> aset”	<i>Valid</i>
4	Pengujian tombol kembali	Klik <i>icon arrow back</i>	Menampilkan halaman <i>Dashboard</i>	<i>Valid</i>

5. Halaman Menu New Asset

Tabel 4. 6 Pengujian Halaman *New Asset*

No	Test Case	Pengujian Kasus	Hasil Yang Diharapkan	Hasil Pengujian
1	Halaman <i>New Asset</i>	Klik icon (+) pada <i>Dashboard</i>	Menampilkan halaman <i>New Asset</i>	<i>Valid</i>
2	Menambah data aset baru	Mengisi semua <i>textfield</i> data aset	Menampilkan <i>alert</i> “berhasil menambahkan aset” jika berhasil dan <i>generate QR Code</i> di halaman <i>Dashboard</i> .	<i>Valid</i>
3	Menambah data aset baru	Salah satu <i>textfield</i> data aset tidak di isi	Menampilkan <i>alert</i> “ <i>Error</i> semua wajib di isi”	<i>Valid</i>

6. Halaman Menu Scan *QR Code*

Tabel 4. 7 Pengujian Halaman *camera scan*

No	Test Case	Pengujian Kasus	Hasil Yang Diharapkan	Hasil Pengujian
1	Halaman kamera Scan	Klik <i>floating action button</i>	Menampilkan Halaman kamera <i>scan</i>	Valid
2	Scan <i>QR Code</i>	Arahkan kamera ke <i>QR Code</i> yang terdaftar	Menampilkan halaman <i>Detail</i>	Valid
3	Scan di tempat minim cahaya	Menyalakan <i>flash camera</i>	Lampu kamera menyala	Valid
4	Membatalkan <i>scan</i>	Klik tombol <i>cancel</i>	Menampilkan <i>alert</i> “tidak ada aset ini di <i>database</i> ”	Valid

7. Halaman Menu Pencarian

Tabel 4. 8 Pengujian Halaman *Search*

No	Test Case	Pengujian Kasus	Hasil Yang Diharapkan	Hasil Pengujian
1	Halaman pencarian	Klik <i>icon (+)</i> pada <i>Dashboard</i>	Menampilkan Halaman pencarian	Valid
2	Mencari aset	Tulis kata kunci aset yang akan di cari	Menampilkan list data dengan sesuai kata kunci	Valid

4.3 Pemeliharaan Sistem

Pada tahapan kali ini, penulis menjalin komunikasi dengan pengguna apabila terdapat error pada aplikasi atau perlu adanya perubahan maupun penambahan fitur maka akan segera dilakukan perubahan sesuai dengan kebutuhan. Pemeliharaan sistem juga dilakukan oleh penulis dengan melakukan backup data demi mengantisipasi hal-hal yang kemungkinan menyerang keamanan sistem.