

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem

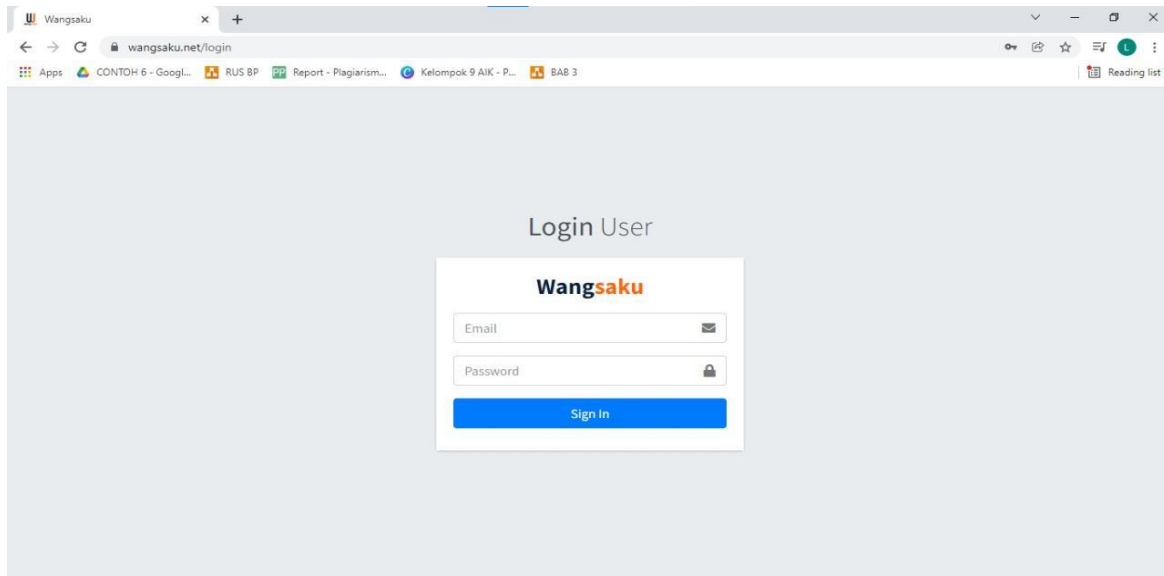
Penerapan dalam perancangan dan pengembangan sistem dilakukan dengan menggunakan laptop dengan sistem operasi *Windows 7*, kemudian menggunakan *web browser Google Chrome* dan pengembangan kode program menggunakan bahasa pemrograman PHP dan aplikasi *visual studio code* yang dapat memudahkan dalam pembangunan suatu web yang dinamis menjadi salah satu faktor penulis menggunakannya. Selain itu, penulis menggunakan *web server XAMPP Control Panel PHP* dengan basis MVC untuk mempermudah dalam pembangunan sistem karena terdapat fitur – fitur yang telah disediakan dan siap digunakan tanpa harus membangun kode program dari dasar.

4.2 Implementasi Aplikasi Web Wangsaku

Implementasi yang dilakukan antara lain adalah pengembangan sistem yang menerapkan perancangan diagram dan antarmuka atau *interface* pada bab sebelumnya, adapun beberapa implementasi yang akan dilakukan pada bab ini yaitu perancangan struktur data kedalam bentuk tabel dan *database* atau basis data, pembuatan kode program dan lain sebagainya. Berikut menu yang tersedia pada aplikasi web Wangsaku, yaitu :

4.2.1 Tampilan Halaman Beranda Login

Halaman login yang mendasari adalah tampilan yang mendasari aplikasi web Wangsaku di halaman ini ada dua bidang teks untuk memasukkan nama pengguna dan frase rahasia kemudian ada tombol untuk menemukan cara login untuk masuk ke aplikasi web. Gambar 4.1 di bawah menunjukkan tampilan halaman login dan Tabel 4.1 adalah kode sumber dari tampilan halaman login.



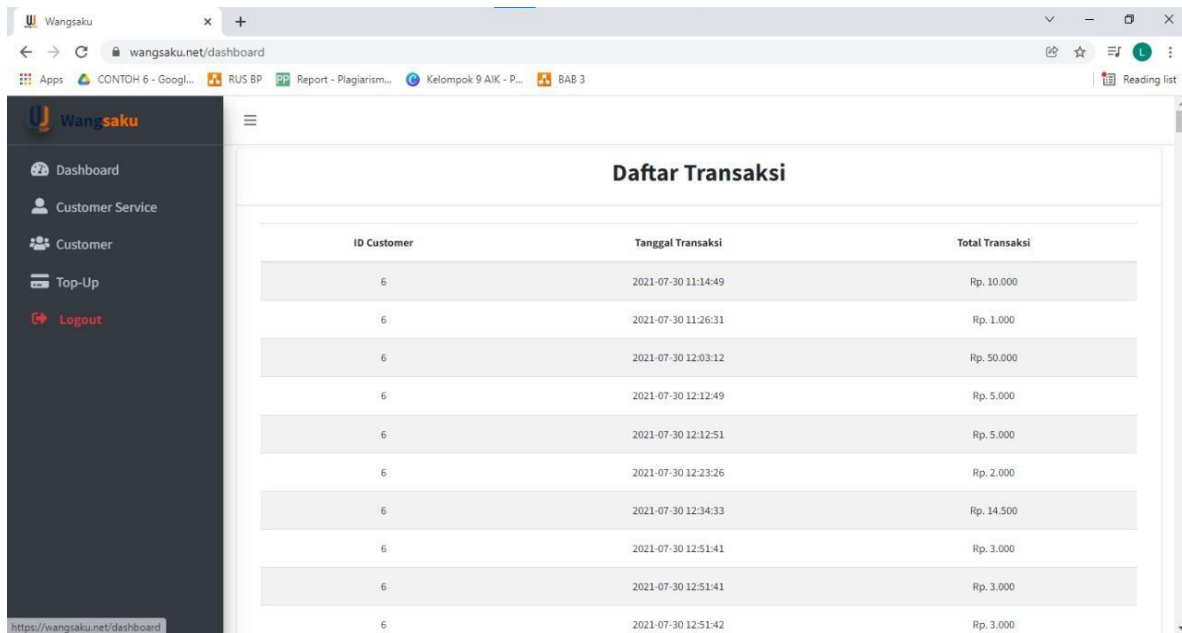
Gambar 4.1 Tampilan Halaman Beranda *Login*

Tabel 4.1 *Source Code* Tampilan Halaman Beranda *Login*

```
<?php
namespace App\Http\Controllers;
use Illuminate\Support\Facades\Auth;
use Illuminate\Http\Request;
use Illuminate\Support\Str;
use App\Models\User;
class LoginController extends Controller
{
    //
    public function login(){
        return view('layouts.login');
    }
    public function loginproses(Request $request){
        if(Auth::attempt($request->only('email','password'))){
            return redirect('/dashboard');
        }else{
            return redirect('/login');
        }
    }
    public function register(){
        return view('layouts.register');
    }
    public function registerproses(Request $request){
        User::create([
            'email' => $request->email,
            'password' => bcrypt($request->password),
            'remember_token' => Str::random(60),
        ]);
        return redirect('/login');
    }
    public function logout(){
        Auth::logout();
        return redirect('/login');
    }
}
```

4.2.2 Tampilan Halaman Beranda *Dashboard*

Halaman beranda *Dashboard* merupakan halaman utama dari website Wangsaku. Fitur ini meliputi kolom ID Customer, Tanggal Transaksi serta Total Transaksi. Berikut pada gambar 4.2 disajikan halaman *Dashboard* atau halaman Daftar Transaksi dan pada Tabel 4.2 merupakan *source code* tampilan halaman *Dashboard*.



The screenshot shows a web browser displaying the Wangsaku dashboard. The left sidebar contains navigation links: Dashboard, Customer Service, Customer, Top-Up, and Logout. The main content area is titled 'Daftar Transaksi' and displays a table with transaction data.

ID Customer	Tanggal Transaksi	Total Transaksi
6	2021-07-30 11:14:49	Rp. 10.000
6	2021-07-30 11:26:31	Rp. 1.000
6	2021-07-30 12:03:12	Rp. 50.000
6	2021-07-30 12:12:49	Rp. 5.000
6	2021-07-30 12:12:51	Rp. 5.000
6	2021-07-30 12:23:26	Rp. 2.000
6	2021-07-30 12:34:33	Rp. 14.500
6	2021-07-30 12:51:41	Rp. 3.000
6	2021-07-30 12:51:41	Rp. 3.000
6	2021-07-30 12:51:42	Rp. 3.000







Gambar 4.2 Tampilan Halaman *Dashboard*

Tabel 4.2 *Source Code* Tampilan Halaman *Dashboard*

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use App\Models\Transaksi;
class DashboardController extends Controller
{
    //
    public function __construct()
    {
        $this->middleware('auth');
    }
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index(Request $request)
    {
        //
        $transaksi = Transaksi::get();
        return view('dashboard.index', compact('transaksi'));
    }
}
```

4.2.3 Tampilan Halaman Beranda *Customer service*

Pada halaman data *Customer Service* dari aplikasi web Wangsaku terdapat menu data *Customer Service* yang meliputi Nama, Jenis kelamin, Tempat Tgl.lahir, serta no Telp/Hp dan menu kelola yang berisi fitur look, edit dan delete data. Berikut pada gambar 4.3 disajikan tampilan halaman data *Customer Service* dan Tabel 4.3 merupakan *source code* halaman tampilan *Customer Service* data.

Nama	Jenis Kelamin	Tempat, Tgl. lahir	No. HP	Kelola
ega	Perempuan	malang, 2000-04-13	08289999999	  
Lale Wiega Arifah Chopsah	Perempuan	Lombok, 2000-04-16	085338520501	  

Gambar 4.3 Tampilan Halaman Beranda *Customer Service*

Tabel 4.3 *Source Code* Tampilan Halaman Beranda *Customer Service*

```

<?php

namespace App\Http\Controllers;

use App\Models\Pelayan;
use Illuminate\Http\Request;

class PelayanController extends Controller
{
    public function __construct()
    {
        $this->middleware('auth');

    public function index(Request $request)
    {
        //
        $q = $request->get('q');
        $pelayan = Pelayan::where('nama_cs', 'LIKE', '%'.$q.'%')->orderBy('nama_cs')->paginate(5);
        return view('pelayan.index', compact('pelayan','q'));
    }
    * @return \Illuminate\Http\Response
    */
    public function create()
    {
        return view('pelayan.create');
    }

    public function store(Request $request)
    {
        $this->validate($request, [

            'nama_cs' => 'required|string|max:255',
            'jenis_kelamin_cs' => 'required|string|max:255',
            'tempat_lahir_cs' => 'required',
            'tanggal_lahir_cs' => 'required|date',
            'nope_cs' => 'required|string|max:255',
            'alamat_cs' => 'required|string|max:255'

        ]);
        Pelayan::create($request->all());
        return redirect()->route('pelayan.index')->with('success', 'Berhasil Menyimpan Data Customer
Service : '.$request->get('nama_cs'));
    }

    public function show($id)
    {
        //
        $pelayan = Pelayan::findOrFail($id);
        return view('pelayan.detail', compact('pelayan'));
    }

    $pelayan = Pelayan::findOrFail($id);
    return view('pelayan.edit', compact('pelayan'));
}

```

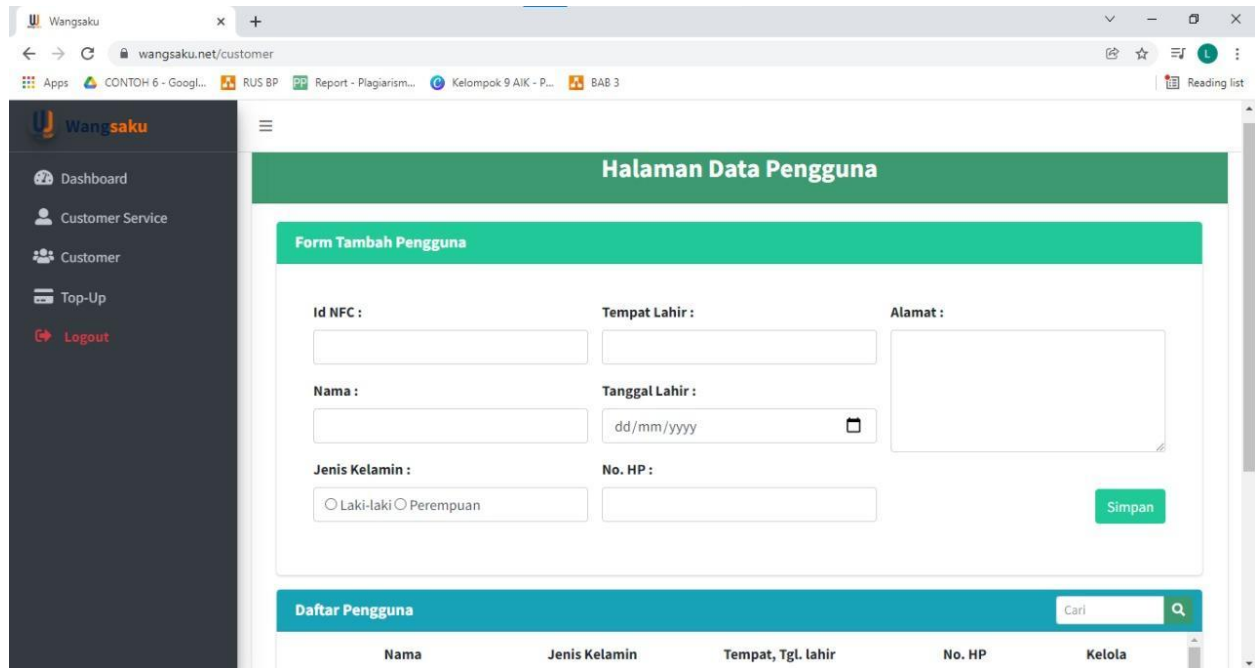
```

*/
public function update(Request $request, $id){
    $pelayan = Pelayan::findOrFail($id);
    $this->validate($request, [
        'nama_cs' => 'required|string|max:255',
        'jenis_kelamin_cs' => 'required|string|max:255',
        'tempat_lahir_cs' => 'required',
        'tanggal_lahir_cs' => 'required|date',
        'nope_cs' => 'required|string|max:255',
        'alamat_cs' => 'required|string|max:255'
    ]);
    $pelayan->update($request->all());
    return redirect()->route('pelayan.index')->with('success',
        'Berhasil Update Data Customer Service : '.$request->get('nama_cs'));
}
* Remove the specified resource from storage.
* @param int $id
* @return \Illuminate\Http\Response
*/
public function destroy($id)
{
    Pelayan::find($id)->delete();
    return redirect()->route('customer.index')->with('error',
        'Berhasil Hapus Data Customer Service !!');
    return redirect()->route('pelayan.index');
}
}
}

```

4.2.4 Tampilan Halaman Beranda *Customer*

Pada halaman data *Customer* dari aplikasi web Wangsaku terdapat menu data *Customer* yang meliputi Nama, Jenis kelamin, Tempat Tanggal Lahir, nomor Telp/Hp dan menu kelola yang berisi fitur look, edit dan delete data. Berikut pada gambar 4.4 disajikan tampilan halaman data *Customer* dan Tabel 4.4 merupakan *source code* untuk halaman data *Customer*.



Gambar 4.4 Tampilan Halaman Beranda *Customer*

Tabel 4.4 *Source Code* Tampilan Halaman Beranda *Customer*

```

<?php
namespace App\Http\Controllers;
use App\Models\Customer;
use App\Models\Tsaldo;
use Illuminate\Http\Request;
use Validator;
class CustomerController extends Controller
{
    public function __construct()
    {
        $this->middleware('auth');
    }
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index(Request $request)
    {
        //
        $q = $request->get('q');
        $customer = Customer::where('nama_customer', 'LIKE', '%'.$q.'%')->orderBy('nama_customer')-
        >paginate(100);
        return view('customer.indexbaru', compact('customer','q'));
    }
    /**

    public function create()
    {
        //
        return view('customer.create');
    }
    /**
     * Store a newly created resource in storage.
     *
     * @param \Illuminate\Http\Request $request
     * @return \Illuminate\Http\Response
     */
    public function store(Request $request)
    {
        //
        $this->validate($request, [
            'nama_customer' => 'required|string|max:255',
            'jenis_kelamin' => 'required|string|max:255',
            'tempat_lahir' => 'required',
            'tanggal_lahir' => 'required|date',
            'nope' => 'required|string|max:255',
            'alamat_customer' => 'required|string|max:255',
            'id_nfc'=> 'required|string|max:255'
        ]);
        $customer=Customer::create($request->all());
    }
}

```



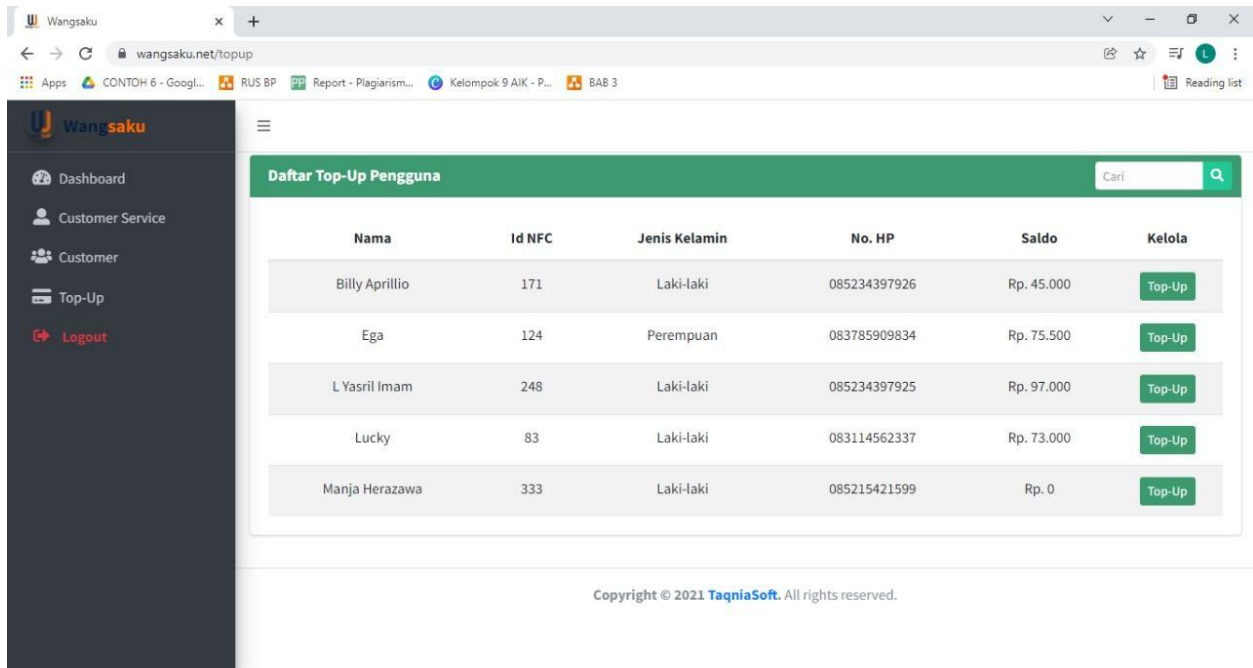
```

        $tsaldo = new Tsaldo();
        $tsaldo->id_nfc= $request->id_nfc;
        $tsaldo->jml_saldo= 0;
        $customer->tsaldo()->save($tsaldo);
        return redirect()->route('customer.index')->with('success', 'Berhasil Menyimpan Data Pengguna');
    : '$request->get('nama_customer')');
    }
    /**
     * Display the specified resource.
     *
     * @param \App\Models\Customer $customer
     * @return \Illuminate\Http\Response
     */
    public function show($id)
    {
        //
        $customer = Customer::findOrFail($id);
        return view('customer.detail', compact('customer'));
    }
    /**
     * Show the form for editing the specified resource.
     *
     * @param \App\Models\Customer $customer
     * @return \Illuminate\Http\Response
     */
    public function edit($id)
    {
        //
        $customer = Customer::findOrFail($id);
        $customer->id_nfc = $customer->tsaldo->id_nfc;
        return view('customer.edit', compact('customer'));
    }
    /**
     * Update the specified resource in storage.
     *
     * @param \Illuminate\Http\Request $request
     * @param \App\Models\Customer $customer
     * @return \Illuminate\Http\Response
     */
    public function update(Request $request, $id){
        $customer = Customer::findOrFail($id);
        $this->validate($request, [
            'nama_customer' => 'required|string|max:255',
            'jenis_kelamin' => 'required|string|max:255',
            'tempat_lahir' => 'required',
            'tanggal_lahir' => 'required|date',
            'nope' => 'required|string|max:255',
            'alamat_customer' => 'required|string|max:255',
            'id_nfc' => 'required|string|max:255',
        ]);
        $customer->update($request->all());
    }

```

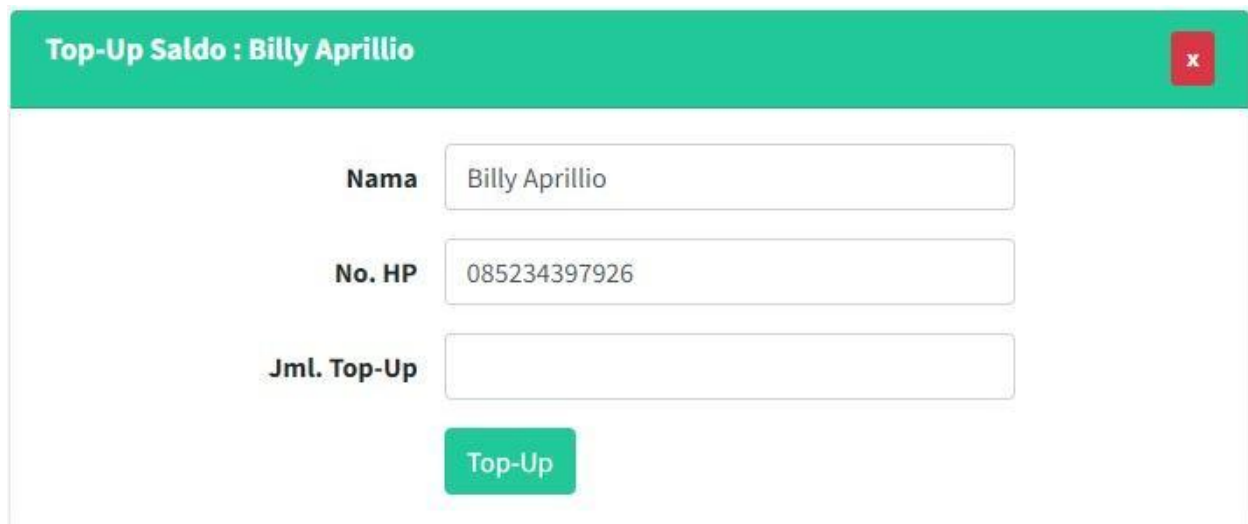
4.2.5 Tampilan Halaman Beranda Top Up

Pada halaman data Top Up dari aplikasi web Wangsaku terdapat menu data Pengguna yang meliputi Nama, Id NFC, Jenis kelamin, no Telp/Hp, Saldo dan menu kelola yang berisi *text field* Top Up untuk menampilkan halaman *form* pengisian saldo. Berikut pada gambar 4.5 disajikan tampilan halaman data Top Up, gambar 4.6 disajikan tampilan halaman *form* Top Up saldo dan Tabel 4.5 merupakan *source code* tampilan halaman data Top Up .



Nama	Id NFC	Jenis Kelamin	No. HP	Saldo	Kelola
Billy Aprillio	171	Laki-laki	085234397926	Rp. 45.000	<button>Top-Up</button>
Ega	124	Perempuan	083785909834	Rp. 75.500	<button>Top-Up</button>
L Yasril Imam	248	Laki-laki	085234397925	Rp. 97.000	<button>Top-Up</button>
Lucky	83	Laki-laki	083114562337	Rp. 73.000	<button>Top-Up</button>
Manja Herazawa	333	Laki-laki	085215421599	Rp. 0	<button>Top-Up</button>

Gambar 4.5 Tampilan Halaman Beranda Top Up



Top-Up Saldo : Billy Aprillio

Nama Billy Aprillio

No. HP 085234397926

Jml. Top-Up

Top-Up

Gambar 4.6 Tampilan Halaman *form* pengisian saldo

Tabel 4.5 *Source Code* Tampilan Halaman Beranda Top Up

```

<?php
namespace App\Http\Controllers;
use App\Models\Customer;
use App\Models\Tsaldo;
use App\Models\Transaksi;
use App\Models\Topup;
use Illuminate\Http\Request;
class TopupController extends Controller
{
    public function __construct()
    {
        $this->middleware('auth');
    }
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index(Request $request)
    {
        //
        $q = $request->get('q');
        $customer = Customer::where('nama_customer', 'LIKE', '%'.$q.'%')->orderBy('nama_customer')-
>paginate(5);
        return view('topup.index', compact('customer','q'));
    }
    public function topup($id)
    {
        //
        $customer = Customer::findOrFail($id);
        return view('topup.create', compact('customer'));
    }

    public function dotopup(Request $request, $id) {
        $tsaldo = Tsaldo::findOrFail($id);
        $this->validate($request, [
            'nama_customer' => 'required|string|max:255',
            'nope' => 'required|string|max:255',
            'jml_topup' => 'required'
        ]);

        $jml= $request->input('jml_topup');
        $nope= $request->input('nope');

        //Save Topup
        $data = $request->all();
        $topup = new Topup;
        $topup->id_topup = $id;
        $topup->jml_topup = $request->input('jml_topup');
        $topup -> save();
    }
}

```

4.3 Pengujian Sistem

Sistem yang telah selesai dirancang kemudian akan melalui tahapan pengujian sistem. Pengujian sistem dilakukan dengan menilai dari segi fungsionalitas sistem dan dari sudut pandang pengguna menggunakan dua metode yaitu metode pengujian *Blakcbox Testing* dan pengujian UAT (*User Acceptance Testing*). Tujuan dari pengujian sistem pada penelitian ini adalah untuk mengetahui sistem yang telah didesain apakah sesuai kebutuhan atau belum memenuhi. Dan juga untuk menghindari terjadinya kesalahan saat sistem digunakan.

4.3.1 Pengujian Blackbox Testing

Sebelum melakukan pengujian pada *Blackbox Testing*, adapun skenario pengujian yang dirancang untuk menentukan langkah – langkah dalam melakukan pengujian sistem aplikasi web Wangsaku, diantaranya adalah sebagai berikut :

1. load halaman login
2. button "sign in"
3. load halaman dashboard
4. button "dashboard"
5. load halaman customer service
6. button “customer service”
7. button "simpan"
8. button mata"look"
9. button "back"
10. button pensil "edit"
11. button “update”
12. button "x"
13. button bak sampah "delete"
14. load halaman customer
15. button "customer "
16. button "simpan"
17. button mata"look"
18. button "back"
19. pensil "edit"

20. button “update”
21. button "x"
22. button bak sampah "delete"
23. load halaman Top Up
24. button "Top Up"
25. text link "Top Up"
26. button "Top Up"
27. button "x"
28. button log out

Setelah menemukan skenario pengujian, langkah berikutnya yaitu melakukan pengujian pada sistem menggunakan tabel *Blackbox Testing*. Tujuan dari pengujian menggunakan metode ini yaitu memeriksa secara menyeluruh terhadap fungsi pada sistem yang telah dikembangkan untuk mengkonfirmasi apakah sistem ini beroperasi dengan semestinya sesuai harapan oleh pengguna. Berikut pada Tabel 4.6 menjelaskan terkait pembagian serta hasil dari pengujian *Blackbox Testing*.

Tabel 4.6 Tabel Pengujian *Blackbox Testing*.

No	Fitur	Hasil yang diharapkan	Hasil
1	load halaman <i>login</i>	Dapat menampilkan halaman eranda <i>login</i>	Sukses <input checked="" type="checkbox"/> Gagal <input type="checkbox"/>
2	button "sign in"	Dapat memvalidasi data berdasrkan inputon <i>Button “Sign in”</i>	Sukses <input checked="" type="checkbox"/> Gagal <input type="checkbox"/>
3	load halaman dashboard	Dapat menampilkan halaman beranda Daftar Transaksi	Sukses <input checked="" type="checkbox"/> Gagal <input type="checkbox"/>
4	button "dashboard"	Dapat mengalihkan ke halaman Daftar Transaksi	Sukses <input checked="" type="checkbox"/> Gagal <input type="checkbox"/>
5	load halaman customer service	Dapat menampilkan halaman customer service	Sukses <input checked="" type="checkbox"/> Gagal <input type="checkbox"/>
6	button “customer service”	Dapat mengalihkan ke halaman customer service	Sukses <input checked="" type="checkbox"/> Gagal <input type="checkbox"/>

7	button "simpan"	Dapat menyimpan data inputan dari <i>form</i> menu “Tambah CS”	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
8	button mata"look"	Dapat mengalihkan ke halaman detail informasi customer service	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
9	button "back"	Dapat menutup halaman informasi detail customer service	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
10	button pensil "edit"	Dapat mengalihkan ke <i>form</i> edit customer service	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
11	button “update”	Dapat menyimpan dan memperbaharui informasi yang telah di edit	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
12	button "x"	Dapat menutup <i>form</i> “edit data customer service”	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
13	button bak sampah "delete"	Dapat menghapus data customer service yang terdaftar	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
14	load halaman customer	Dapat menampilkan halaman pengguna	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
15	button "customer "	Dapat mengalihkan ke halaman pengguna	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
16	button "simpan"	Dapat menyimpan data inputan dari <i>form</i> menu “Tambah daftar pengguna”	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
17	button mata"look"	Dapat mengalihkan ke halaman detail informasi pengguna	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
18	button "back"	Dapat menutup halaman informasi detail pengguna	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
19	pensil "edit"	Dapat mengalihkan ke <i>form</i> edit pengguna	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
20	button “update”	Dapat menyimpan dan memperbaharui informasi yang telah di edit	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
21	button "x"	Dapat menutup <i>form</i> “edit data pengguna”	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>

22	button bak sampah "delete"	Dapat menghapus data pengguna yang terdaftar	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
23	load halaman Top Up	Dapat menampilkan halaman Top Up	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
24	button "Top Up"	Dapat mengalihkan ke halaman Top Up	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
25	text link "Top Up"	Dapat menampilkan form "Top Up saldo Pengguna"	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
26	button "Top Up"	Dapat memperbaharui informasi saldo pengguna	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
27	button "x"	Dapat menutup form "Top Up saldo pengguna"	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>
28	button log out	Dapat keluar dari sistem dan kembali ke halaman beranda login	Sukses <input checked="" type="checkbox"/>	Gagal <input type="checkbox"/>

Dari skenario uji yang terdapat 28 poin pengujian yang dimana hasil pengujian pada fungsionalitas aplikasi web tidak terdapat kesalahan dan semuanya dapat dioperasikan dengan baik dan sukses.

4.3.2 Pengujian UAT (User Acceptance Test)

Setelah dilakukannya pengujian *Blackbox Testing* maka tahapan selanjutnya dilakukan pengujian dari sisi pengguna menggunakan metode UAT *testing*. Pengguna akan diberikan pernyataan yang sesuai dengan format pengujian UAT. Berikut pada Tabel 4.7 menjelaskan terkait pembagian serta hasil dari pengujian UAT.

Tabel 4.7 Tabel Pengujian UAT

No	Pengujian	Langkah pengujian	Hasil yang diharapkan	Hasil		
				Diterima	Diterima dengan catatan	ditolak

1	<i>Login</i>	Memasukkan <i>username</i> dan <i>password</i> => <i>Sign in</i>	<i>Login</i> sukses, masuk ke halaman Dashboard	✓		
2	Lihat <i>History</i> Transaksi	Klik menu dashboard	Halaman <i>History</i> Transaksi dapat ditampilkan	✓		
3	<i>Customer service</i>	Klik menu <i>Customer service</i>	Dapat menampilkan halaman <i>Customer service</i>	✓		
4	Lihat data <i>customer service</i>	Klik menu <i>Customer service</i> => button mata	Dapat menampilkan halaman detail data <i>Customer service</i>	✓		
5	Tambah data <i>customer service</i>	Klik menu <i>Customer service</i> => isi identitas nama => tempat lahir => tanggal lahir => pilih jenis kelamin => alamat => simpan	Data <i>Customer service</i> berhasil ditambahkan	✓		
6	Edit data <i>customer service</i>	Klik menu <i>Customer service</i> => button pensil => edit nama,	Data <i>Customer service</i> berhasil diedit	✓		

		tempat lahir , tanggal lahir, pilih jenis kelamin, alamat => simpan				
7	Hapus data <i>customer service</i>	Klik menu <i>Customer service</i> => button bak sampah	Data <i>Customer service</i> berhasil dihapus	✓		
8	<i>customer</i>	Klik menu <i>Customer</i>	Dapat menampilkan halaman Pengguna	✓		
9	Lihat data pengguna	Klik menu <i>Customer</i> => button mata	Dapat menampilkan halaman detail data Pengguna	✓		
10	Tambah data pengguna	Klik menu <i>Customer</i> => isi identitas nama => Id NFC => tempat tanggal lahir=> pilih jenis kelamin=>alamat => simpan	Data Pengguna berhasil ditambahkan	✓		
11	Edit data pengguna	Klik menu <i>Customer</i> => button pensil => edit nama,Id NFC tempat tanggal lahir, pilih jenis kelamin, alamat => simpan	Data Pengguna berhasil diedit	✓		

12	Hapus data pengguna	Klik menu <i>Customer</i> => button bak sampah	Data Pengguna berhasil dihapus	✓		
13	Top Up	Klik menu Top Up	Dapat menampilkan halaman daftar data Top Up pengguna	✓		
14	Tambah Saldo	Klik menu Top Up => button Top Up => isi nominal saldo => Top Up	Dapat memperbaharui nominal saldo Pengguna	✓		
15	<i>Logout</i>	Klik menu <i>Logout</i>	<i>Logout</i> sukses, kembali ke halaman <i>Login</i> .	✓		

Berdasarkan pengujian hasil UAT *testing* yang sudah diselesaikan, setiap fitur pada aplikasi web Wangsaku telah sesuai dengan yang pengguna butuhkan serta dapat diterima dan dipahami dengan baik. Adapun beberapa catatan dari hasil pengujian UAT *testing* antara lain:

1. Perlu penambahan fitur cari dalam menu Dashboard demi memudahkan dalam pencarian data pengguna sewaktu – waktu jika di butuhkan dan hanya perlu memasukkan kata kunci seperti Id NFC dari pengguna tersebut.
2. Perlu ditambahkan menu edit pada fitur data Top Up pengguna, demi menghindari adanya kesalahan dalam input data Top