

# LAMPIRAN

## Lampiran 1. Surat Kesepakatan Bimbingan

### SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama : Sugeng Sulistianto  
NIM : 20090091  
Program Studi : Sarjana Terapan Teknik Informatika

Pihak Kedua

Nama : Muhammad Fikri Hidayattullah, S.T., M.Kom.  
Status : Dosen Tetap  
NIDN : 0623108801  
Jabatan Fungsional : Lektor  
Pangkat/Golongan : Penata Tk. I/III-d

Pada hari ini Kamis tanggal 01 April 2024 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing I Skripsi Pihak Pertama dengan syarat Pihak Pertama berjanji akan menyelesaikan Skripsi dalam waktu maksimal 3 bulan (awal bulan Juni 2024). Jika syarat tersebut tidak terpenuhi, maka Pihak Kedua berhak untuk tidak melanjutkan proses bimbingan. Adapun waktu dan tempat pelaksanaan disepakati antar pihak. Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi

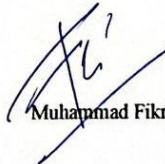
Tegal, 01 April 2024

Pihak Pertama



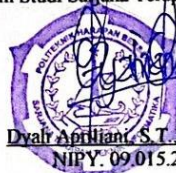
Sugeng Sulistianto

Pihak Kedua



Muhammad Fikri Hidayattullah, S.T., M.Kom.

Mengetahui  
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Apriliani, S.T., M.Kom.  
NIPY. 09.015.225

## SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama : Sugeng Sulistianto  
NIM : 20090091  
Program Studi : Sarjana Terapan Teknik Informatika

Pihak Kedua

Nama : Mirza Alim Mutasodirin, S.Kom., M.Kom  
Status : Dosen  
NIDN : 0605049004  
Jabatan Fungsional : -  
Pangkat/Golongan : -

Pada hari ini Kamis tanggal 01 April 2024 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing II Skripsi Pihak Pertama dengan syarat Pihak Pertama wajib melakukan bimbingan Skripsi minimal 1 minggu 1 kali kepada Pihak Kedua. Adapun waktu dan tempat pelaksanaan disepakati antar pihak. Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi.

Tegal, 01 April 2024

Pihak Pertama



Sugeng Sulistianto

Pihak Kedua



Mirza Alim Mutasodirin, S.Kom., M.Kom

Mengetahui  
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Anriliani, S.T., M.Kom  
NIPY. 09.015.225

## Lampiran 2. Surat Pernyataan Pengajuan HKI

### SURAT PERNYATAAN

Yang bertanda tangan di bawah ini, pemegang hak cipta:

1. Nama : Sugeng Sulistianto  
Kewarganegaraan : Indonesia  
Alamat : Jl. Pakulaut Timur, RT 03 RW 06, Desa Pakulaut,  
Kec. Margasari, Kab. Tegal, Provinsi Jawa Tengah.
2. Nama : Muhammad Fikri Hidayattullah, S.T., M.Kom-  
Kewarganegaraan : Indonesia  
Alamat : Jl. Glatik No.68, Randugunting, Kec. Tegal Selatan, Kota Tegal  
Provinsi Jawa Tengah, 52193.
3. Nama : Mirza Alim Mutasodirin, S.Kom., M.Kom.  
Kewarganegaraan : Indonesia  
Alamat : Griya Santika blok J no. 11, Pengabean, Kecamatan Dukuhturi,  
Kabupaten Tegal, Jawa Tengah.

Dengan ini menyatakan bahwa:

1. Karya Cipta yang saya mohonkan:  
Berupa : Program Komputer  
Berjudul : Sansekerta Vibe : E-Commerce Shop for Future

Tidak meniru dan tidak sama secara esensial dengan Karya Cipta milik pihak lain atau obyek kekayaan intelektual lainnya sebagaimana dimaksud dalam Pasal 68 ayat (2);

- Bukan merupakan Ekspresi Budaya Tradisional sebagaimana dimaksud dalam Pasal 38;
- Bukan merupakan Ciptaan yang tidak diketahui penciptanya sebagaimana dimaksud dalam Pasal 39;
- Bukan merupakan hasil karya yang tidak dilindungi Hak Cipta sebagaimana dimaksud dalam Pasal 41 dan 42;
- Bukan merupakan Ciptaan seni lukis yang berupa logo atau tanda pembeda yang digunakan sebagai merek dalam perdagangan barang/jasa atau digunakan sebagai lambang organisasi, badan usaha, atau badan hukum sebagaimana dimaksud dalam Pasal 65 dan;
- Bukan merupakan Ciptaan yang melanggar norma agama, norma susila, ketertiban umum, pertahanan dan keamanan negara atau melanggar peraturan perundang-undangan sebagaimana dimaksud dalam Pasal 74 ayat (1) huruf d Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.

2. Sebagai pemohon mempunyai kewajiban untuk menyimpan asli contoh ciptaan yang dimohonkan dan harus memberikan apabila dibutuhkan untuk kepentingan penyelesaian sengketa perdata maupun pidana sesuai dengan ketentuan perundang-undangan.

3. Karya Cipta yang saya mohonkan pada Angka 1 tersebut di atas tidak pernah dan tidak sedang dalam sengketa pidana dan/atau perdata di Pengadilan.
4. Dalam hal ketentuan sebagaimana dimaksud dalam Angka 1 dan Angka 3 tersebut di atas saya / kami bersedia secara sukarela bahwa:
  - a. Permohonan karya cipta yang saya ajukan dianggap ditarik kembali; atau
  - b. Karya Cipta yang telah terdaftar dalam Daftar Umum Ciptaan Direktorat Hak Cipta, Direktorat Jenderal Hak Kekayaan Intelektual, Kementerian Hukum Dan Hak Asasi Manusia R.I dihapuskan sesuai dengan ketentuan perundang-undangan yang berlaku.
  - c. Dalam hal kepemilikan Hak Cipta yang dimohonkan secara elektronik sedang dalam berperkara dan/atau sedang dalam gugatan di Pengadilan maka status kepemilikan surat pencatatan elektronik tersebut ditangguhkan menunggu putusan Pengadilan yang berkekuatan hukum tetap.

Demikian Surat pernyataan ini saya/kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.

Tegal, 30 Juli 2024



(Sugeng Sulistianto)  
Pemegang Hak Cipta \*

(Muhammad Fikri Hidayattullah, S.T., M.Kom)  
Pemegang Hak Cipta \*

(Mirza Alim Mutasodirin, S.Kom., M.Kom.)  
Pemegang Hak Cipta \*

### Lampiran 3. Surat Pengalihan Hak Cipta

#### SURAT PENGALIHAN HAK CIPTA

Yang bertanda tangan di bawah ini:

1. Nama : Sugeng Sulistianto  
Kewarganegaraan : Indonesia  
Alamat : Ds. Pakulaut RT 003/ RW 006 Kec. Margasari, Kab. Tegal
2. Nama : Muhammad Fikri Hidayattullah, S.T., M.Kom.  
Kewarganegaraan : Indonesia  
Alamat : Jl. Glatik No.68, Randugunting, Kec. Tegal Selatan, Kota Tegal, Provinsi Jawa Tengah
2. Nama : Mirza Alim Mutasodirin, S.Kom., M.Kom.  
Kewarganegaraan : Indonesia  
Alamat : Griya Santika blok J no. 11, Pengabean, Kec. Dukuhturi, Kabupaten Tegal, Jawa Tengah

Adalah **Pihak I** selaku pencipta, dengan ini menyerahkan karya ciptaan saya kepada:

Nama : Pusat Penelitian dan Pengabdian Masyarakat (P3M)  
Alamat : Jl. Mataram No. 9 Pesurungan Lor Kota Tegal

Adalah **Pihak II** selaku Pemegang Hak Cipta berupa Program Komputer dengan judul "Sansekertta Vibe : E-Commerce Shop for Future". untuk didaftarkan di Direktorat Hak Cipta dan Desain Industri, Direktorat Jenderal Kekayaan Intelektual, Kementerian Hukum dan Hak Asasi Manusia Republik Indonesia.

Demikian Surat pernyataan ini kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.


Tegal, 30 Juli 2024  
Pencipta

Pemegang Hak Cipta  
Karya P3M  
  
(Dr. Aldi Budi Riyanta, S.Si., M.T)



(Sugeng Sulistianto)

  
(Muhammad Fikri Hidayattullah, S.T., M.Kom.)

  
(Mirza Alim Mutasodirin, S.Kom., M.Kom.)

Lampiran 4. Syarat Pengajuan HKI

*MANUAL BOOK*  
*SANSEKERTTA VIBE E-COMMERCE*



## DAFTAR ISI

1. PENDAHULUAN	
1.1. Tujuan Pembuatan Dokumen.....	5
1.2. Deskripsi Umum Aplikasi .....	5
1.3. Deskripsi Dokumen.....	5
2. PERANGKAT YANG DIBUTUHKAN .....	6
2.1. Perangkat Lunak.....	6
3. MENU DAN CARA PENGGUNAAN.....	6
2.1. Struktur Menu.....	6
User	
1. Home.....	7
2. Register.....	7
Admin	
1. Verifikasi Email.....	10
2. Verifikasi Email.....	10
3. Login.....	10
4. <i>Dashboard</i> .....	10
5. Store Info .....	10
6. Kategori.....	10
7. Produk .....	11
8. Costumer .....	11
9. Cart.....	11
10. Order .....	11
11. <i>Payments</i> .....	11
12. <i>Shipment</i> atau Pengiriman .....	11
2.2. Cara Penggunaan .....	12
User	



1. Halaman Home.....	12
2. Halaman Register.....	13
3. Verifikasi Email.....	13
4. Halaman Login.....	14
5. Halaman Profile User .....	15
6. Halaman Kategori Produk.....	17
7. Detail Produk.....	17
8. Halaman <i>Cart</i> .....	18
9. Halaman Order .....	19
10. Pembayaran.....	21
11. Pengiriman/Shipment .....	23
<b>Admin</b>	
1. Registrasi.....	24
2. Verifikasi Email.....	25
3. Login.....	25
4. Halaman <i>Dashboard</i> admin.....	26
5. Store Info .....	27
6. Halaman Kategori.....	28
7. Halaman Produk.....	30
8. Halaman Data Customer.....	32
9. Halaman <i>Cart</i> .....	32
10. Halaman Order .....	33
11. Halaman Payment.....	35
12. Halaman Shipment atau Pengiriman.....	36

## DAFTAR GAMBAR

Gambar 1. Halaman <i>Home</i> .....	12
Gambar 2. Halaman Register .....	13
Gambar 3. Verifikasi Email .....	14
Gambar 4. Halaman Login.....	14
Gambar 5. Halaman <i>Profile User</i> .....	16
Gambar 6. <i>Point</i> .....	16
Gambar 7. Halaman Kategori Produk.....	17
Gambar 8. Detail Produk.....	18
Gambar 9. <i>Cart Icon</i> .....	18
Gambar 10. Halaman Cart .....	19
Gambar 11. Tampilan menu order yang belum terverifikasi .....	20
Gambar 12. Tampilan menu order yang sudah terverifikasi .....	21
Gambar 13. Halaman pembayaran .....	22
Gambar 14. Halaman pemilihan bank.....	22
Gambar 15. Halaman <i>payment successful</i> .....	23
Gambar 16. Halaman info paket .....	24
Gambar 17. Halaman registrasi.....	24
Gambar 18. Verifikasi email .....	25
Gambar 19. Halaman Login .....	26
Gambar 20. Halaman Dashboard admin .....	26
Gambar 21. Store Info .....	27
Gambar 22. Halaman Kategori .....	28
Gambar 23. Halaman data kategori.....	28
Gambar 24. Halaman edit kategori .....	29
Gambar 25. Halaman <i>delete</i> atau hapus data kategori .....	29
Gambar 26. Halaman Produk.....	30
Gambar 27. Halaman Tambah produk.....	30
Gambar 28. Halaman edit produk .....	31
Gambar 29. Halaman delete atau hapus data produk .....	31
Gambar 30. Halaman Data Customer .....	32
Gambar 31. Halaman Cart .....	33
Gambar 32. Halaman <i>Cart</i> .....	33
Gambar 33. Halaman mengatur pengiriman .....	34
Gambar 34. Halaman setelah mengatur pengiriman .....	35
Gambar 35. Halaman Payment.....	35
Gambar 36. Halaman setelah melakukan pembayaran .....	36
Gambar 37. Halaman Shipment atau Pengiriman .....	36

## **PENDAHULUAN**

### **1.1. Tujuan Pembuatan Dokumen**

Dokumen *user manual* ini dibuat untuk memberikan penjelasan dan gambaran atau sebagai panduan kepada pengguna mengenai penggunaan Aplikasi *E-Commerce* Sansekertta berbasis website dengan implementasi Gamifikasi.

### **1.2. Deskripsi Umum Aplikasi**

Aplikasi ini dirancang untuk menjadikan efisiensi memperkenalkan produk dari Brand Sansekertta menggunakan *E-Commerce* berbasis website kepada pengguna atau costumer dengan fitur gamifikasi didalam *E-Commerce*. Gamifikasi di dalam *E-Commerce* Sansekertta ini berbentuk Loyalty Program, yaitu untuk meningkatkan loyalitas pelanggan dengan memberikan poin untuk setiap pembelian, yang kemudian dapat ditukar dengan produk lainnya.

### **1.3. Deskripsi Dokumen**

Dokumen ini dibuat untuk memberikan panduan penggunaan Aplikasi Rekomendasi Film Asia

#### 1. Bagian 1

Bagian ini berisi informasi umum yang meliputi tujuan pembuatan dokumen, deskripsi umum sistem, dan deskripsi dokumen.

#### 2. Bagian 2

Bagian ini berisi perangkat yang dibutuhkan untuk penggunaan aplikasi meliputi perangkat lunak.

### 3. Bagian 3

Bagian ini berisi panduan atau cara penggunaan.

## 1. PERANGKAT YANG DIBUTUHKAN

### 2.1. Perangkat Lunak

Untuk menjalankan aplikasi ini, berikut beberapa perangkat lunak yang dibutuhkan yaitu:

1. Aplikasi ini dijalankan pada platform web browser pada perangkat
2. Untuk menjalankan website *E-Commerce* dengan lancar, diperlukan koneksi internet yang stabil.

## 2. MENU DAN CARA PENGGUNAAN

### 2.1. Struktur Menu

Ada 2 pengguna dalam *E-Commerce* Sansekertta yaitu *admin* dan *user*.

Berikut struktur menu yang ada pada *E-Commerce* Sansekertta adalah sebagai berikut:

#### *User*

##### 1. Home

Menu ini akan menampilkan halaman awal atau Home ketika *E-Commerce* Sansekertta pertama kali diakses dibrowser.

## 2. Register

Menu ini digunakan untuk mendaftarkan akun baru. Pengguna harus mengisi informasi seperti nama, email, password, dan confirm password.

### 1) Verifikasi Email

Setelah melakukan register maka user akan mendapatkan email untuk melakukan verifikasi email yang sudah didaftarkan pada saat melakukan registrasi akun.

### 2) Login

Menu ini digunakan untuk masuk ke aplikasi dengan akun yang sudah terdaftar. Pengguna harus memasukkan email dan kata sandi yang telah didaftarkan.

### 3) Profile User

Halaman ini akan tampil ketika user telah melakukan login pada aplikasi *E-Commerce* Sansekertta, yang berisikan profile, keranjang, order, pembayaran, dan pengiriman. Lengkapi data profile user agar saat melakukan pembelian produk admin bisa langsung memprosesnya. Profile juga akan menampilkan point yang dimiliki oleh user, point akan didapatkan dari user ketika sudah melakukan pembelian

produk di *E-Commerce* Sansekertta, dimana point tersebut akan bertambah dari 10% harga dari pembelian produk.

4) Kategori Produk

Menu ini akan menampilkan halaman kategori produk dari *E-Commerce* Sansekertta, user bisa memilih kategori produk yang ingin dibeli.

5) Detail Produk

Menu ini menampilkan detail dari produk *E-Commerce* Sansekertta, ketika user melakukan klik pada salah satu produk, yang berisikan nama dari produk, harga, foto produk, keterangan produk, stock produk, dan tambah ke keranjang.

6) Cart

Menu ini digunakan untuk melihat produk yang sudah masuk kedalam keranjang dengan jumlah yang sudah ditambahkan oleh user.

7) Order

Menu yang menampilkan produk yang sudah diorder, dengan menampilkan beberapa detail order produk dari *E-Commerce* Sansekertta, dan untuk melakukan pembayaran tunggu konfirmasi dari admin untuk melakukan pembayaran.

8) Pembayaran

Menu ini akan tampil ketika order produk dari user sudah di verifikasi oleh admin. Setelah order produk di verifikasi oleh admin, user bisa langsung melakukan pembayaran dengan beberapa metode pembayaran yang tersedia di *E-Commerce* Sansekertta.

9) Pengiriman

Menu ini menampilkan pengirim yang dilakukan oleh *E-Commerce* Sansekertta, yang berisi order id, ongkos kirim, jasa pengiriman, estimasi hari, pengecekan resi, dan status pengiriman.

***Admin***

1. Register

Menu ini digunakan untuk mendaftarkan akun baru. Admin harus mengisi informasi seperti nama, email, password, dan confirm password.

## 2. Verifikasi Email

Setelah melakukan register maka user akan mendapatkan email untuk melakukan verifikasi email yang sudah didaftarkan pada saat melakukan registrasi akun.

## 3. Login

Menu ini digunakan untuk masuk ke *dashboard* dengan akun yang sudah terdaftar sebagai admin. Admin harus memasukkan email dan kata sandi yang telah didaftarkan.

## 4. *Dashboard*

Halaman ini merupakan halaman pembuka untuk halaman *dashboard* admin yang dimana disini akan menampilkan nama user, kategori, produk, dan total order.

## 5. Store Info

Pada halaman ini akan menampilkan data dari alamat toko Sansekerta untuk menentukan pengiriman dari lokasi toko ketempat tujuan user.

## 6. Kategori

Pada halaman ini akan menampilkan halaman yang digunakan untuk menambahkan kategori untuk produk, edit kategori produk, hapus kategori produk.



#### 7. Produk

Pada halaman ini akan menampilkan halaman yang digunakan untuk menambahkan produk, edit produk, dan hapus produk.

#### 8. Costumer

Halaman ini akan menampilkan customer atau user yang menggunakan *E-Commerce* Sansekertta. Disini admin bisa daftar customer pada halaman customer.

#### 9. Cart

Pada halaman ini akan menampilkan data dari user yang sudah memasukkan produk ke dalam keranjang.

#### 10. Order

Pada halaman order atau pembayaran ini, menampilkan data dari order yang sudah dilakukan oleh user pada saat akan membeli produk. Disini admin bertugas mengecek status order, verifikasi order, pengiriman, dan hapus order.

#### 11. *Payments*

Pada halaman ini menampilkan data dari order yang sedang diproses sampai order yang sudah dibayar oleh user.

#### 12. *Shipment* atau Pengiriman

Pada halaman ini menampilkan data pengiriman dari pembayaran user yang sudah berhasil.

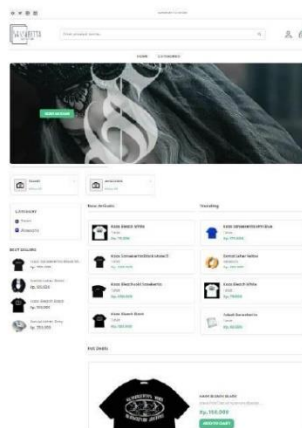
## 2.2. Cara Penggunaan.

### User

#### 1. Halaman Home

Pada saat pertama kali mengakses website *E-Commerce* Sansekertta akan langsung masuk ke halaman Home atau *dashboard* dari *E-Commerce*. Halaman Home ini menampilkan beberapa menu dan fungsinya, sebagai berikut :

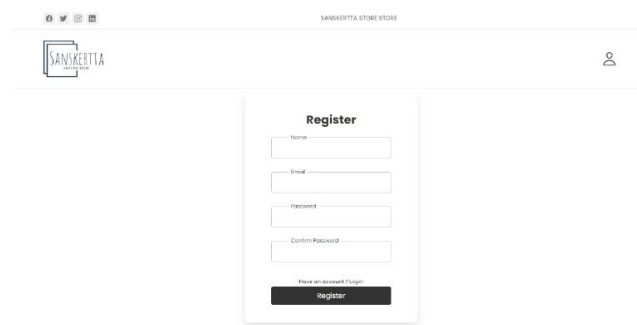
- a. Menu pencarian disini berguna untuk mencari produk yang ingin costumer beli.
- b. Ada menu kategori untuk menampilkan beberapa produk di dalam kategori.



Gambar 1. Halaman *Home*

## 2. Halaman Register

Halaman register pada website *E-Commerce* Sansekertta setiap user atau pengguna wajib membuat akun untuk melakukan transaksi di *E-Commerce* Sansekertta. Untuk mendaftar pengguna atau *user* memasukkan data nama, *email*, *password*, dan *confirm password*, setelah itu click register untuk melakukan pendaftaran akun user.

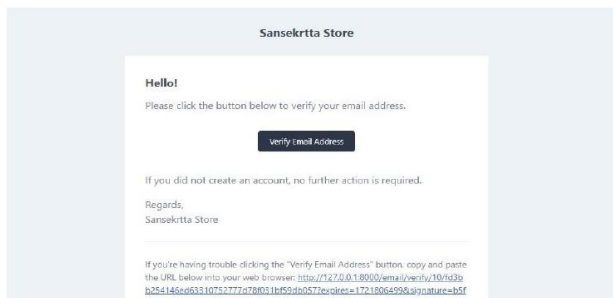


The image shows a screenshot of the 'Register' page on the Sansekertta website. The page has a white background with a header containing the Sansekertta logo on the left and a user profile icon on the right. The main content is a 'Register' form with the following fields: 'Name', 'Email', 'Password', and 'Confirm Password'. Below these fields is a 'Register' button. The form is centered on the page.

Gambar 2. Halaman Register

## 3. Verifikasi Email

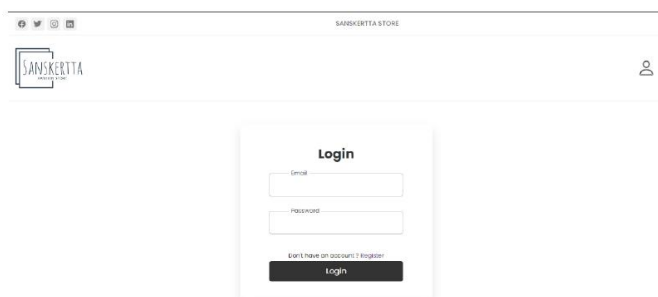
Setelah melakukan registrasi menggunakan akun email di website *E-Commerce* maka, akun email tersebut akan otomatis mendapatkan verifikasi pada email yang telah didaftarkan pada akun di website *E-Commerce*. Selanjutnya klik *verify email address*, maka akan otomatis ke *redirect* ke halaman login.



Gambar 3. Verifikasi Email

#### 4. Halaman Login

Halaman login di *E-Commerce* Sanskr̥tta, digunakan untuk memasukan akun yang sudah didaftarkan oleh user melalui registrasi. Untuk melakukan login, pengguna atau user harus memasukkan email dan password yang sudah terdaftar pada halaman registrasi sebelumnya.



Gambar 4. Halaman Login

## 5. Halaman Profile User

Halaman *profile user* ini, akan tampil beberapa menu seperti pada gambar dibawah :

- Profile, terdapat data user profile dihalaman ini harus wajib diisi, dengan mengisi semua data yang ada di menu profile ini, maka user bisa menggunakan akun tersebut untuk melakukan pembelian di *E-Commerce* Sansekertta.
- Cart, cart di halaman ini bertujuan untuk menampilkan produk yang sudah ditambahkan user.
- Order, halaman order ini akan menampilkan data order dari halaman cart yang telah diorder oleh user.
- Payments, payment atau pembayaran disini menampilkan deskripsi dari menu order untuk dilakukan pembayaran.
- Shipment, shipment atau pengiriman disini bertujuan untuk menampilkan produk yang sudah berhasil di order dan menampilkan deskripsi pengiriman sampai dengan nomor resi.
- Point, point disini adalah fitur gamifikasinya, jadi *E-Commerce* Sansekertta itu menerapkan sistem gamifikasi yaitu, loyalty program. Point loyalty program ini bisa didapatkan ketika membeli produk, dan ketika user sudah membeli produk maka

akan mendapatkan point 10% dari harga pembelian produk di *E-Commerce* Sansekertta. Dimana Point ini bisa digunakan sebagai diskon untuk user ketika ingin membeli produk di Sansekertta.

Profile Cart Order Payment Shipment

**Name**  
Pelanggan Satu

**Email**  
pelanggan1@mail.id

**Password**  
Password

**Photo**  
Choose File No file chosen

**Province**  
Java Tengah

**City**  
Salatiga

**Phone**  
081211113

**Postal Code**  
50711

**Gender**  
Man

**Detail Address**  
reimbang

Point Kamu : 750  
Shop  
Logout

Save Reset

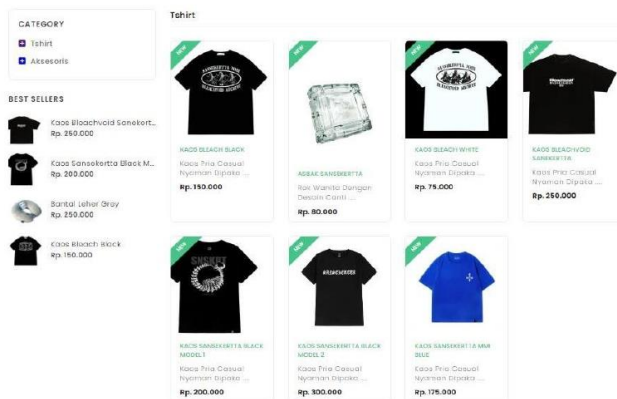
Gambar 5. Halaman *Profile User*

Point Kamu : 2500

Gambar 6. *Point*

## 6. Halaman Kategori Produk

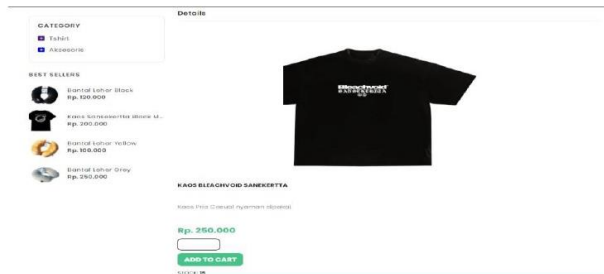
Halaman ini menampilkan halaman produk yang berada di dalam kategori produk.



Gambar 7. Halaman Kategori Produk

## 7. Detail Produk

Halaman detail produk akan menampilkan halaman detail dari produk untuk melihat deskripsi produk, harga, dan stock produk.



Gambar 8. Detail Produk

## 8. Halaman *Cart*

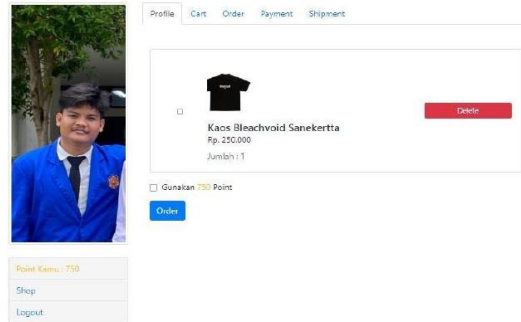
Untuk mengakses halaman cart, klik gambar berikut pada halaman *dashboard*.



Gambar 9. *Cart Icon*

Setelah memasuki halaman cart akan tampil halaman cart seperti gambar berikut



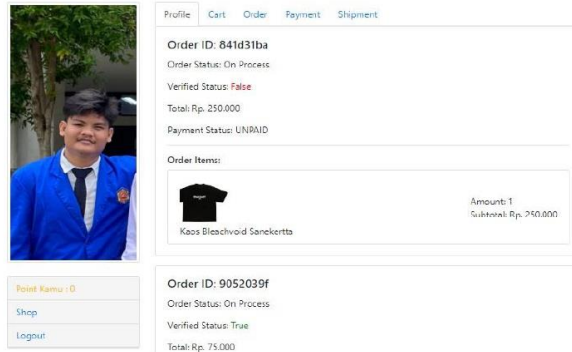


Gambar 10. Halaman Cart

Isi dari halaman cart adalah foto produk beserta harganya, dan ketika user mempunyai point, maka user bisa menggunakan point dari user untuk ditukarkan dengan potongan harga.

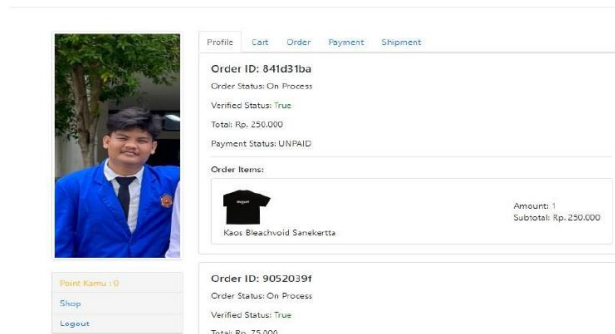
### 9. Halaman Order

Pada halaman order ini, user perlu menunggu verifikasi dari admin untuk bisa melanjutkan ke halaman pembayaran. Berikut adalah gambar ketika tampilan dimenu order yang belum di verifikasi oleh admin dimana verified statusnya adalah false.



Gambar 11. Tampilan menu order yang belum terverifikasi

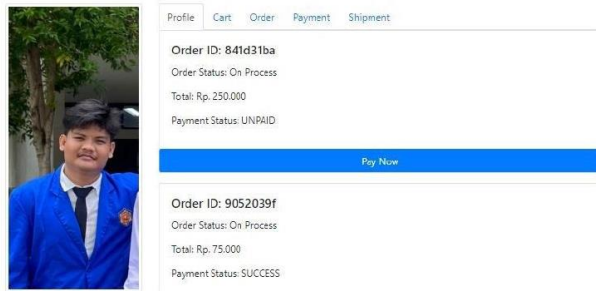
Ketika admin sudah melakukan verifikasi order pada halaman admin, maka tampilan pada menu order pada verified status berubah menjadi true, seperti gambar berikut ini:



Gambar 12. Tampilan menu order yang sudah terverifikasi

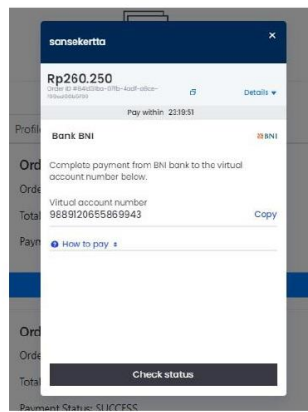
## 10. Pembayaran

Pada halaman pembayaran, ketika order produk dari user sudah di verifikasi oleh admin. Setelah order produk di verifikasi oleh admin, user bisa langsung melakukan pembayaran dengan beberapa metode pembayaran yang tersedia di *E-Commerce* Sansekertta. Berikut adalah gambar ketika akan melakukan pembayaran :



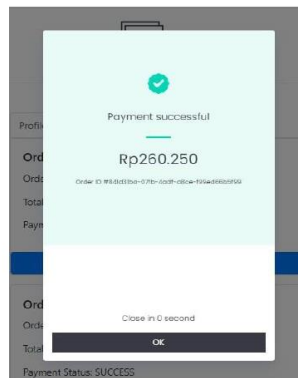
Gambar 13. Halaman pembayaran

Selanjutnya pilih Bank untuk melakukan pembayaran, seperti gambar berikut :



Gambar 14. Halaman pemilihan bank

Setelah melakukan pembayaran, maka akan menampilkan *payment successful*, seperti gambar berikut



Gambar 15. Halaman *payment successful*

#### 11. Pengiriman/Shipment

Setelah melakukan pembayaran berhasil, user dapat mengakses info paket pada menu pengiriman dengan detail seperti gambar berikut :



Gambar 16. Halaman info paket

## Admin

### 1. Registrasi

Halaman registrasi admin dari tampilannya sama dengan registrasi user, untuk membuat akun sebagai admin, bisa masukan nama, email, password, dan confirm password, selanjutnya lakukan registrasi.

Berikut adalah gambar registrasi :

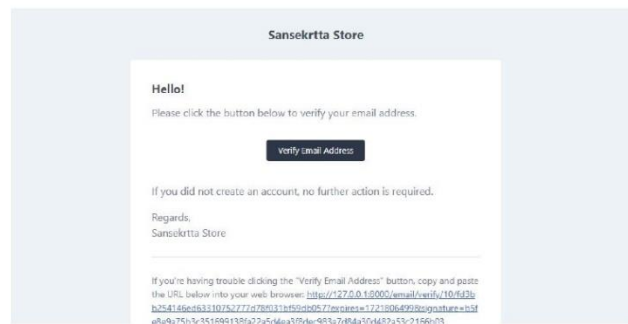


Gambar 17. Halaman registrasi

## 2. Verifikasi Email

Setelah melakukan registrasi menggunakan akun email di website *E-Commerce* maka, akun email tersebut akan otomatis mendapatkan verifikasi pada email yang telah didaftarkan pada akun di website *E-Commerce*. Selanjutnya klik verify email address, maka akan otomatis ke redirect ke halaman login.

Berikut adalah gambar verifikasi email :

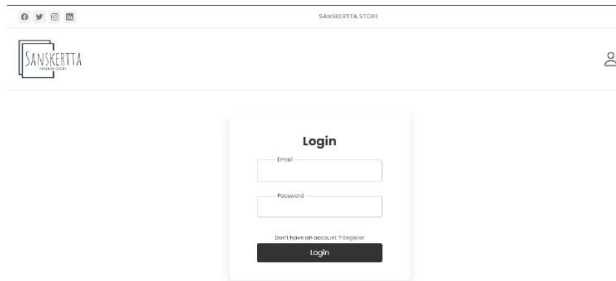


Gambar 18. Verifikasi email

## 3. Login

Setelah melakukan verifikasi email, selanjutnya melakukan login menggunakan email yang sudah terverifikasi sebelumnya, masukan email dan password lalu klik login.

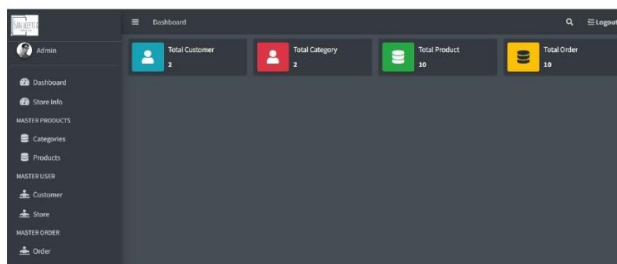
Berikut adalah gambar dan tampilan darihalaman login:



Gambar 19. Halaman Login

#### 4. Halaman *Dashboard* admin

Halaman ini merupakan halaman pembuka untuk halaman *dashboard* admin yang dimana disini akan menampilkan total user, total kategori, total produk, dan total order.

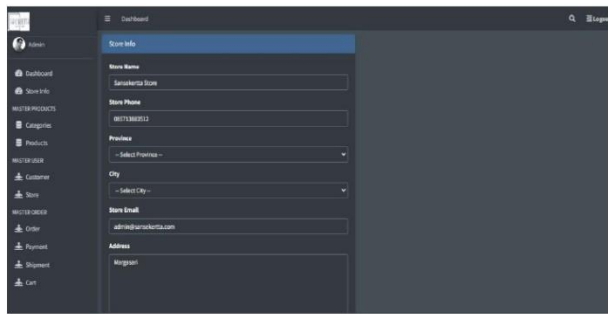


Gambar 20. Halaman Dashboard admin



## 5. Store Info

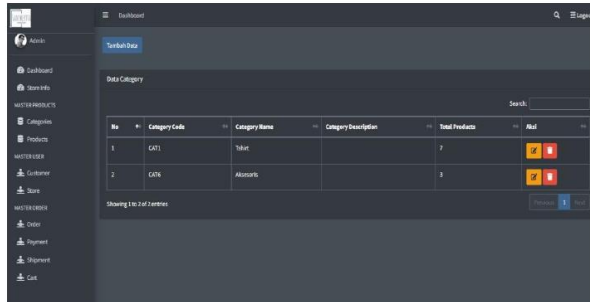
Pada halaman ini akan menampilkan data dari alamat toko Sansekertta untuk menentukan pengiriman dari lokasi toko ketempat tujuan user, hal ini wajib karena dengan menambahkan data toko akan memudahkan ongkos kirim untuk pembelian produk pada toko *E- Commerce* Sansekertta.



The image shows a screenshot of a web application dashboard. On the left is a dark sidebar with a navigation menu containing items like 'Dashboard', 'Store Info', 'Products', 'Customers', 'Orders', 'Payments', and 'Cart'. The main content area is titled 'Store Info' and contains a form with the following fields: 'Store Name' (filled with 'Sansekertta Store'), 'Store Phone' (filled with '08773303211'), 'Province' (a dropdown menu with 'Select Province'), 'City' (a dropdown menu with 'Select City'), 'Store Email' (filled with 'admin@sansekertta.com'), and 'Address' (filled with 'Kraggan').

Gambar 21. Store Info

## 6. Halaman Kategori



No	Category Code	Category Name	Category Description	Total Products	Atd
1	CAT1	Milk		7	
2	CAT2	Almonds		3	

Gambar 22. Halaman Kategori

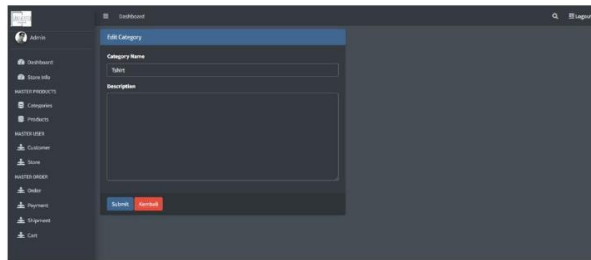
Pada halaman kategori ini admin dapat menambahkan data kategori seperti pada gambar berikut :



Gambar 23. Halaman data kategori

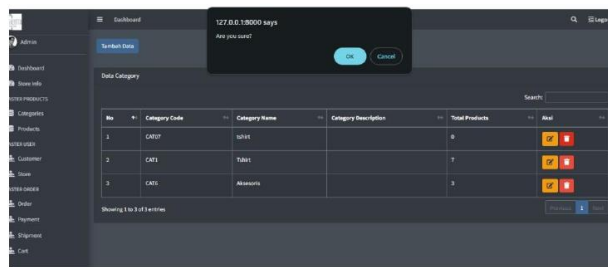
Selain bisa menambahkan kategori, halaman ini bisa melakukan edit kategori, klik halaman edit kategori, isi edit kategori yang ingin diubah, lalu klik submit untuk melakukan penyimpanan data edit

kategori. Berikut adalah gambar dari edit produk:



Gambar 24. Halaman edit kategori

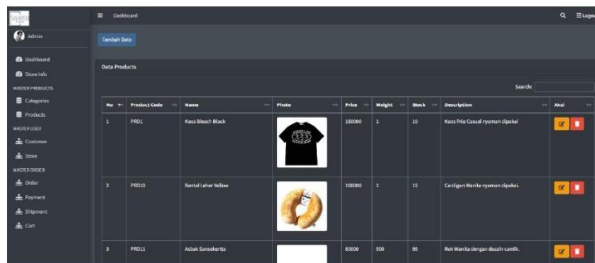
Selain bisa menambahkan dan edit kategori, halaman kategori untuk admin ini bisa melakukan delete atau hapus data kategori, klik icon delete dan akan muncul alert seperti gambar berikut :












Gambar 25. Halaman *delete* atau hapus data kategori,

## 7. Halaman Produk

Pada halaman ini akan menampilkan halaman yang digunakan untuk menambahkan produk, edit produk, dan hapus produk. Berikut tampilan halaman data produk:



No	Product Code	Name	Photo	Price	Weight	Stock	Description	Action
1	PROD1	Kaos Hitam		10000	1	10	Kaos Polo Casual Premium Quality	 
2	PROD2	Kardigan Hitam		10000	1	10	Kardigan Wanita Premium Quality	 
3	PROD3	Kaos Putih		10000	1	10	Kaos Wanita dengan Busui Comfort	 

Gambar 26. Halaman Produk

Halaman Tambah produk, untuk mengakses halaman ini, klik tambah produk yang ada di halaman data produk, berikut halaman tambahproduk dapat dilihat pada gambar :



**Add New Product**

Product Category:

Product Photo:

Product Name:

Product Price:

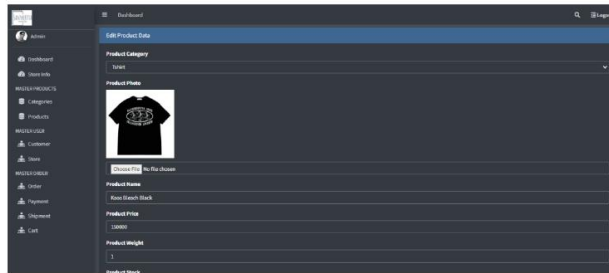
Product Weight:

Product Stock:

Product Description:

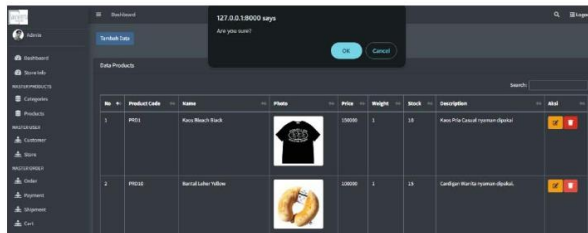
Gambar 27. Halaman Tambah produk

Halaman edit produk, halaman ini sesuai namanya dapat berguna untuk mengedit produk, dengan cara klik icon edit dan isi bagian yang akan diedit. Berikut adalah gambar dari edit produk:



Gambar 28. Halaman edit produk

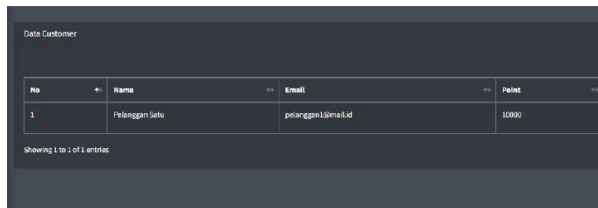
Selain bisa menambahkan dan edit produk, halaman produk untuk admin ini bisa melakukan delete atau hapus data produk, klik icon delete dan akan muncul alert seperti gambar berikut



Gambar 29. Halaman delete atau hapus data produk

## 8. Halaman Data Customer

Halaman ini akan menampilkan customer atau user yang menggunakan E Commerce Sansekertta. Disini admin bisa daftar customer pada halaman customer.



The screenshot shows a dark-themed interface with a table titled "Data Customer". The table has four columns: "No", "Nama", "Email", and "Point". There is one row of data with the following values: "1", "Pelanggan Satu", "pelanggan1@gmail.id", and "10000". Below the table, it says "Showing 1 to 1 of 1 entries".

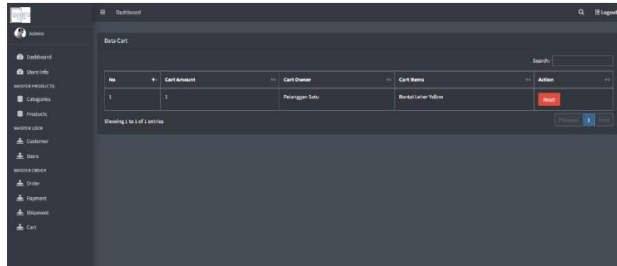
No	Nama	Email	Point
1	Pelanggan Satu	pelanggan1@gmail.id	10000

Showing 1 to 1 of 1 entries

Gambar 30. Halaman Data Customer

## 9. Halaman Cart

Pada halaman ini akan menampilkan data dari user yang sudah memasukkan produk ke dalam keranjang.



Gambar 31. Halaman Cart

## 10. Halaman Order

Pada halaman order atau pembayaran ini, menampilkan data dari order yang sudah dilakukan oleh user pada saat akan membeli produk. Disini admin bertugas mengecek status order, verifikasi order, pengiriman, dan hapus order.

Berikut adalah halaman order ketika admin belum menyiapkan pembayaran:

No	Order Status	Order Total	Order Payment Status	Order Date	Action
1	On Process	200000	UNPAID	False	<a href="#">Detail</a>
2	On Process	100000	UNPAID	False	<a href="#">Detail</a>
3	On Process	0	UNPAID	False	<a href="#">Detail</a>

Gambar 32. Halaman Cart

Pada proses ini admin akan melakukan verifikasi pada orderan yang masuk, disini tugas admin akan mengatur pengiriman, mulai dari kurir, servis yang pas untuk produk yang dikirim, setelah selesai selanjutnya klik icon verif dan orderan sudah siap untuk dibayar oleh customer. Setelah customer membayar maka dari *E-Commerce* Sansekerta melakukan pengiriman ke jasa pengiriman yang telah di verif oleh admin, setelah mendapatkan resi maka pada halaman order ini bisa diupdate untuk memasukkan resi pengirimannya, jadi user bisa melakukan lacak paket pada website jasa pengiriman yang terteta. Berikut adalah gambar ketika admin melakukan tugas mengatur pengiriman.



The image shows a 'Shipment' form with the following fields and values:

Field	Value
Courier	JRE
Service	JTR - JRE Tracking
Price	75000
Estimate	67
Resi	

Buttons at the bottom: Verif (blue), Cancel (red), Save (orange).

Gambar 33. Halaman mengatur pengiriman

Setelah admin mengatur pengiriman maka tampilan pada halaman order akan berubah pada column inverified yang menjadi true seperti gambar berikut:



No	Order Status	Order Total	Order Payment Status	Verified	Action
1	On Process	25000	UNPAID	True	<a href="#">Detail</a>

Gambar 34. Halaman setelah mengatur pengiriman

### 11. Halaman Payment

Pada halaman ini menampilkan data dari order yang sedang diproses sampai order yang sudah dibayar oleh user. Berikut adalah tampilan halaman payment

No	Customer	Status Order	Total Payment	Payment Status
1	Penerimaan Salu	On Process	20000	UNPAID
2	Penerimaan Salu	On Process	10000	UNPAID

Gambar 35. Halaman Payment

Setelah pembayaran dilakukan oleh user maka, column payment status akan berubah menjadi succes, seperti pada gambar berikut:

No	Customer	Status Order	Total Payment	Payment Status
1	Pelanggan Satu	On Process	20000	SUCCESS
2	Pelanggan Satu	On Process	20000	SUCCESS

Showing 1 to 2 of 2 entries

Gambar 36. Halaman setelah melakukan pembayaran

## 12. Halaman Shipment atau Pengiriman

13. Pada halaman ini menampilkan data pengiriman dari pembayaran user yang sudah berhasil.

No	Order ID	Price	Carrier	Estimate	Service	Real
1	130348-032-400-640-3644778a75	7000	jne	67	JNE	
2	130348-032-400-640-3644778a75	3000	jne	4.640	Pu-Reguler	

Showing 1 to 2 of 2 entries

Gambar 37. Halaman Shipment atau Pengiriman



**DOKUMEN TEKNIKAL**  
**IMPLEMENTASI GAMIFICATION**  
**PADA E-COMMERCE (STUDI KASUS**  
**SANSEKERTTA)**

Dibuat oleh:  
Sugeng Sulistianto  
Muhammad Fikri Hidayatullah, S.T., M.Kom.  
Mirza Alim Mutasodirin, S.Kom., M.Kom.

## DAFTAR ISI

DAFTAR ISI .....	2
DAFTAR GAMBAR .....	4
PENDAHULUAN .....	5
1.1. Tujuan Pembuatan Dokumen .....	5
1.2. Deskripsi Umum Aplikasi .....	5
1.3. Deskripsi Dokumen .....	5
1. PERANGKAT YANG DIBUTUHKAN .....	6
2.1. Perangkat Lunak .....	6
2. MENU DAN CARA PENGGUNAAN .....	6
2.1. Struktur Menu .....	6
2.2. Cara Penggunaan .....	12
1. Halaman Home .....	12
2. Halaman Register .....	13
3. Verifikasi Email .....	13
4. Halaman Login .....	14
5. Halaman Profile User .....	15
6. Halaman Kategori Produk .....	17
7. Detail Produk .....	17
9. Halaman Order .....	19
10. Pembayaran .....	21
11. Pengiriman/Shipment .....	23
Admin .....	24
2. Verifikasi Email .....	25
3. Login .....	25
5. Store Info .....	27
6. Halaman Kategori .....	28
7. Halaman Produk .....	30
8. Halaman Data Customer .....	32

9.	Halaman Cart .....	32
10.	Halaman Order .....	33
11.	Halaman Payment .....	35
12.	Halaman Shipment atau Pengiriman .....	36
User .....		44
1.	Halaman Home .....	44
2.	Halaman Register .....	45
3.	Verifikasi Email .....	46
4.	Halaman Login .....	48
5.	Halaman Profile User .....	49
6.	Halaman Kategori Produk .....	52
7.	Detail Produk .....	53
8.	Halaman Cart .....	55
9.	Halaman Order .....	57
10.	Pembayaran .....	59
11.	Pengiriman/Shipment .....	61

## DAFTAR GAMBAR

Gambar 1. <i>Source Code Home</i> .....	44
Gambar 2. Tampilan <i>Home</i> .....	45
Gambar 3. <i>Source Code Register</i> .....	46
Gambar 4. Tampilan Register .....	46
Gambar 5. <i>Source Code Verify Email</i> .....	47
Gambar 6. Tampilan <i>Verify Email</i> .....	48
Gambar 7. <i>Source Code Login</i> .....	48
Gambar 8. Tampilan <i>Login</i> .....	49
Gambar 9. <i>Source Code User</i> .....	50
Gambar 10. Tampilan <i>Profile User</i> .....	51
Gambar 11. <i>Source Code Point</i> .....	51
Gambar 12. Tampilan <i>Point</i> .....	52
Gambar 13. <i>Source Code Kategori Produk</i> .....	52
Gambar 14. halaman kategori produk .....	53
Gambar 15. <i>Source Code Detail Produk</i> .....	54
Gambar 16. Tampilan Detail Produk .....	54
Gambar. 17 <i>Cart Icon</i> .....	55
Gambar 18. Tampilan <i>Cart</i> .....	55
Gambar 19. <i>Source Code Cart</i> .....	56
Gambar 20. Tampilan Menu <i>Order Yang Belum Terverifikasi</i> .....	57
Gambar 21. Tampilan Menu <i>Order Yang Sudah Terverifikasi</i> .....	57
Gambar 22. Gambar <i>Source Code Order</i> .....	58
Gambar 23. Tampilan Pembayaran .....	59
Gambar 24. Halaman Pemilihan Bank .....	59
Gambar 25. Halaman <i>Payment Successful</i> .....	60
Gambar 26. <i>Source Code Payment</i> .....	60
Gambar 27. Tampilan <i>Info Paket</i> .....	61
Gambar 28. <i>Source Code Shipment</i> .....	61
Gambar 29. <i>Source Code Rajaongkir</i> .....	62
Gambar 30. <i>Source Code Admin Login</i> .....	63
Gambar 31. Tampilan <i>Login</i> .....	63
Gambar 32. <i>Tampilan Verifikasi Email</i> .....	64
Gambar 33. <i>Source Code Verifikasi Email</i> .....	65
Gambar 34. <i>Source Code Dashboard Admin</i> .....	66
Gambar 35. Tampilan <i>Dashboard Admin</i> .....	67
Gambar 36. <i>Source Code Store Info</i> .....	67
Gambar 37. Tampilan <i>Store Info</i> .....	68
Gambar 38. <i>Source Code Kategori</i> .....	69
Gambar 39. Tampilan Kategori .....	69
Gambar 40. Halaman Data Kategori .....	70
Gambar 41. Halaman Edit Kategori .....	70
Gambar 42. Halaman <i>Delete</i> Atau Hapus Data Kategori .....	70
Gambar 43. <i>Source Code Produk</i> .....	71

Gambar 44. Halaman Produk .....	72
Gambar 45. Halaman Tambah produk .....	72
Gambar 46. Halaman Edit Produk .....	72
Gambar 47. Halaman <i>Delete</i> Atau Hapus Data Produk .....	73
Gambar 48. <i>Source Code</i> Data <i>Customer</i> .....	74
Gambar 49. Halaman Data <i>Customer</i> .....	74
Gambar 50. <i>Source Code Cart</i> .....	75
Gambar 51. Halaman <i>Cart</i> .....	76
Gambar 52. Halaman <i>Order</i> .....	76
Gambar 53. Halaman Verifikasi <i>Order</i> .....	77
Gambar 54. Halaman <i>Order</i> Terverifikasi .....	77
Gambar 55. <i>Source Code Order</i> .....	78
Gambar 56. Halaman <i>Payment</i> .....	78
Gambar 57. Halaman <i>Payment Succes</i> .....	79
Gambar 58. <i>Source Code</i> Pemesanan .....	80
Gambar 59. <i>Source Code</i> Pembayaran .....	80
Gambar 60. <i>Source Code Shipment</i> .....	81
Gambar 61. <i>Source Code</i> Raja Ongkir .....	82
Gambar 62. Halaman <i>Shipment</i> .....	82

### **Profil**

Sansekertta adalah sebuah website *E-Commerce* yang dibuat untuk memudahkan dan mengenalkan produk kepada masyarakat dengan solusi membuat sebuah *website E-Commerce* dengan fitur Gamifikasi, yaitu fitur *loyalty program*, dimana setiap *customer* membeli produk dari *E-Commerce* Sansekertta akan mendapatkan 10% point dari total pembelian produk dari Sansekertta. *Website E-Commerce* Sansekertta ini dibuat dengan *framework* laravel dengan tambahan *php*, dan *java script*.

### **Latar Belakang**

Perkembangan industri di Indonesia yang merupakan Negara berkembang saat ini sedang tumbuh dan berkembang dengan pesat. Industri kecil sangat membutuhkan media untuk mempublikasikan produk-produk yang di hasilkan salah satunya adalah melalui media internet. karena hal tersebut peranan dari media publikasi contohnya *website E-Commerce* sangat dibutuhkan dalam rangka mempromosikan produk dan barang. *E-commerce* merupakan salah satu kegiatan transaksi bisnis baik barang dan jasa yang dilakukan secara elektronik dengan menggunakan jaringan internet. Karena kemampuan *E-Commerce* yang bisa menjadi solusi dari industri – industri kecil contohnya brand asal kota Bandung yaitu Sanskertta yang bergerak didalam bidang fashion menjadi tergiur dengan adanya trend toko online melalui *website* atau *E-Commerce*.

Kata Sansekertta sendiri diambil dari film yang berjudul *The Lord of The Rings* di karakter (*The Lord of The Nasgul*) yang berarti Raja penyihir hitam agnar di utara eriador, Sansekertta juga memiliki arti sebagai bahasa kesucian. Dari segi produksi Sansekertta melakukan produksinya melalui vendor yang berada di bandung, setiap produknya Sansekertta memiliki vendor yang berbeda – berbeda mulai dari produksi *tshirt*, *tote bag*, bantal serviks, asbak, dll. Sansekertta sendiri juga memberlakukan beberapa produk dengan sistem *limited edition* artinya produk tertentu yang ada di toko Sansekertta dapat dipesan dengan cara sistem *pre order*.

Karena Sansekertta lahir sebagai industri kecil, Sansekertta selama menjual produknya masih menggunakan sistem konvensional, yaitu pembeli melakukan transaksi pembelian produk di toko *offline* Sansekertta. Sansekertta mulai tertarik dengan adanya toko online atau *E-Commerce* itu bukan tanpa alasan, pasalnya *owner* Sansekertta mengatakan akhir – akhir ini toko *offline* nya menjadi kurang ramai pengunjung dan pembeli. *Owner* Sansekertta pun merasa kalau sudah mulai merasa kalah saing dengan beberapa brand yang berada di daerah Bandung. Oleh karena itu *E-Commerce* menjadi satu – satunya solusi untuk mengatasi permasalahan. Karena *E-Commerce* mempunyai kemampuan yang dapat digunakan untuk mempromosikan produk – produk Sanskertta hingga bisa menjadikan toko Sansekertta menjadi lebih luas dikenal banyak masyarakat.

Tetapi itu saja belum cukup untuk menarik perhatian pelanggan dan pengunjung untuk membeli produk dari Sansekertta, maka *E-Commerce* ini membutuhkan sesuatu yang menarik pelanggan yaitu dengan menerapkan sistem *Gamification* atau Gamifikasi, Gamifikasi sendiri adalah suatu produk, cara berpikir, proses, pengalaman, cara desain, dan sistem, yang sekaligus terlibat, dimana menggunakan elemen game untuk menyelesaikan



masalah non game. Sistem Gamifikasi yang akan digunakan dalam website *E-Commerce* Sansekertta ini menggunakan sistem point dan dapat ditukar menjadi discount pada saat pembelian, jadi pengguna bisa membeli produk dari Sansekertta untuk mendapatkan point sebesar 10% dari pembelian, sehingga point dapat ditukarkan dengan *discount* dari Sansekertta. Dengan adanya *website E-Commerce* Sansekertta menggunakan Gamifikasi sebagai daya tarik pelanggan seharusnya dengan ini dapat membuat toko menjadi lebih dikenal lebih luas apalagi dengan keunikannya dimana adanya sistem point menjadi discount di dalam website e-commerce Sansekertta.

**Manfaat**

Penelitian ini diharapkan dapat memberikan manfaat bagi :

**Masyarakat:**

- a. Membuat masyarakat mendapatkan informasi tentang Sansekertta melalui website E-Commerce Sansekertta.
- b. Dengan fitur gamifikasi “loyalty program” menggunakan sistem point ketika membeli produk dari Sansekertta akan mendapatkan point untuk ditukar menjadi diskon harga produk.

**Peneliti :**

- a. Memberikan dukungan dan solusi teknologi yang mudah dikenal masyarakat untuk meningkatkan pengenalan produk Sansekertta.
- b. Memberikan informasi mengenai efektivitas membuat *E-Commerce* menggunakan *framework laravel*.

**Spesifikasi Teknis**

Spesifikasi teknik meliputi :

1. Modul Pengguna
2. Source Code

Berikut uraian spesifikasi untuk pembangunan aplikasi :

1. Windows Ram 8 Gb
2. Visual Studio Code
3. Web Browser
4. Framework laravel
5. PHP
6. Java Scripts
7. Raja Ongkir
8. Midtrans
9. XAMPP
10. Ngrok

**Berikut uraian spesifikasi modul :**

1. Modul Pengguna
2. Source Code

*User*

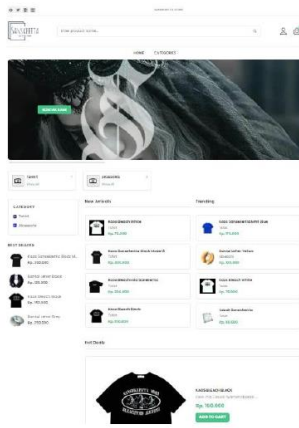
**1. Halaman Home**

Pada saat pertama kali mengakses *website E-Commerce* Sansekerta akan langsung masuk ke halaman home atau dashboard dari *E-Commerce*. Halaman home ini menampilkan beberapa menu dan fungsinya. Berikut penjelasan dari *source* pada halaman home.

- a. namespace `App\Http\Controllers`; mendefinisikan namespace untuk *controller*, sesuai konvensi Laravel.
- b. use `Illuminate\Http\Request`; dan use `Auth`; mengimpor kelas *Request* dan *Auth* dari Laravel.
- c. Fungsi constructor dipanggil saat *instance* dari *HomeController* dibuat.
- d. `$this->middleware('auth')`; menetapkan *middleware* `auth` pada semua metode dalam *controller*, memastikan hanya pengguna terotentikasi yang dapat mengaksesnya.
- e. Metode `index` dipanggil ketika pengguna mengakses rute terkait. `Auth::user()` mendapatkan pengguna yang sedang terotentikasi. `Auth::user()->roles->pluck('name')[0]` mendapatkan peran pertama pengguna. Jika perannya `customer`, pengguna diarahkan ke rute `welcome`; jika tidak, diarahkan ke rute `dashboard`.

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use Auth;
7
8 class HomeController extends Controller
9 {
10     /**
11      * Create a new controller instance.
12      *
13      * @return void
14      */
15     public function __construct()
16     {
17         $this->middleware('auth');
18     }
19     /**
20      * Show the application dashboard.
21      *
22      * @return \Illuminate\Contracts\Support\Renderable
23      */
24     public function index()
25     {
26         if(Auth::user()->roles->pluck('name')[0] == 'customer'){
27             return redirect()->route('welcome');
28         }
29         return redirect()->route('dashboard');
30     }
31 }
```

Gambar 1. Source Code Home



Gambar 2. Tampilan Home

## 2. Halaman Register

Halaman register pada website *E-Commerce* Sansekertta setiap user atau pengguna wajib membuat akun untuk melakukan transaksi di *E-Commerce* Sansekertta. Untuk mendaftar pengguna atau *user* memasukkan data nama, *email*, *password*, dan *confirm password*, setelah itu click register untuk melakukan pendaftaran akun user.

Berikut source code dari halaman registrasi pada halaman controller registrasi:

- a. namespace `App\Http\Controllers\Auth`; mendefinisikan namespace untuk *controller* otentikasi sesuai konvensi Laravel.
- b. *use statements* mengimpor kelas-kelas yang dibutuhkan dalam *controller*.
- c. Kelas `RegisterController` mewarisi dari *Controller* dasar Laravel.
- d. *Trait RegisterUsers* menyediakan fungsionalitas untuk registrasi pengguna.
- e. Properti `$redirectTo` menentukan rute tujuan setelah registrasi berhasil, yaitu *HOME*.
- f. Constructor dipanggil saat instance `RegisterController` dibuat.
- g. `$this->middleware('guest')`; menetapkan *middleware* `guest`, memastikan hanya pengguna yang belum terotentikasi yang dapat mengakses metode dalam *controller*.
- h. Metode validator memvalidasi data registrasi, menerima *array \$data*, dan mengembalikan *instance* validator dengan aturan validasi.
- i. Metode `create` membuat pengguna baru setelah validasi berhasil, menerima *array \$data*, dan mengembalikan instance model `User`.



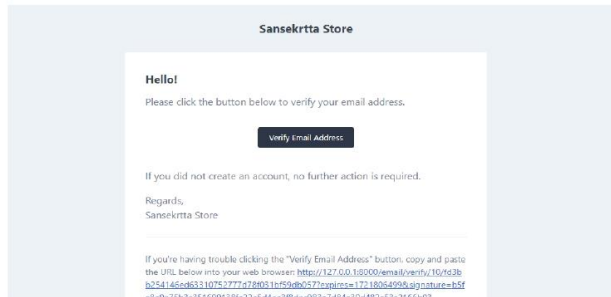
- b. Use statements mengimpor kelas-kelas yang dibutuhkan dalam controller.
- c. Kelas *VerificationController* mewarisi dari *Controller* dasar Laravel.
- d. Trait *VerifiesEmails* menyediakan fungsionalitas untuk verifikasi email dan pengiriman ulang email verifikasi.
- e. Properti *\$redirectTo* menentukan rute tujuan setelah verifikasi email berhasil, yaitu *HOME*.
- f. Constructor dipanggil saat instance *VerificationController* dibuat.
- g. *\$this->middleware('auth');* menetapkan middleware *auth* pada semua metode dalam *controller*, memastikan hanya pengguna terautentikasi yang dapat mengaksesnya.
- h. *\$this->middleware('signed')->only('verify');* menetapkan *middleware signed* pada metode *verify*, memastikan URL verifikasi ditandatangani secara digital.
- i. *\$this->middleware('throttle:6,1')->only('verify', 'resend');* menetapkan middleware *throttle* pada metode *verify* dan *resend*, membatasi jumlah permintaan maksimal 6 kali per menit untuk mencegah penyalahgunaan.

```

1 <?php
2
3 namespace App\Http\Controllers\Auth;
4
5 use App\Http\Controllers\Controller;
6 use App\Providers\RouteServiceProvider;
7 use Illuminate\Foundation\Auth\VerifiesEmails;
8
9 class VerificationController extends Controller
10 {
11     /**
12      * Small Verification Controller
13      *
14      * @var string
15      * @var string
16      * @var string
17      * @var string
18      * @var string
19      * @var string
20      * @var string
21      * @var string
22      * @var string
23      * @var string
24      * @var string
25      * @var string
26      * @var string
27      * @var string
28      * @var string
29      * @var string
30      * @var string
31      * @var string
32      * @var string
33      * @var string
34      * @var string
35      * @var string
36      * @var string
37      * @var string
38      * @var string
39      * @var string
40      * @var string
41      * @var string
42      * @var string
43      * @var string
44      * @var string
45      * @var string
46      * @var string
47      * @var string
48      * @var string
49      * @var string
50      * @var string
51      * @var string
52      * @var string
53      * @var string
54      * @var string
55      * @var string
56      * @var string
57      * @var string
58      * @var string
59      * @var string
60      * @var string
61      * @var string
62      * @var string
63      * @var string
64      * @var string
65      * @var string
66      * @var string
67      * @var string
68      * @var string
69      * @var string
70      * @var string
71      * @var string
72      * @var string
73      * @var string
74      * @var string
75      * @var string
76      * @var string
77      * @var string
78      * @var string
79      * @var string
80      * @var string
81      * @var string
82      * @var string
83      * @var string
84      * @var string
85      * @var string
86      * @var string
87      * @var string
88      * @var string
89      * @var string
90      * @var string
91      * @var string
92      * @var string
93      * @var string
94      * @var string
95      * @var string
96      * @var string
97      * @var string
98      * @var string
99      * @var string
100     */
101     protected $redirectTo = RouteServiceProvider::HOME;
102
103     /**
104      * Create a new controller instance.
105      *
106      * @return void
107      */
108     public function __construct()
109     {
110         $this->middleware('auth');
111         $this->middleware('signed')->only('verify');
112         $this->middleware('throttle:6,1')->only('verify', 'resend');
113     }
114 }

```

Gambar 5. Source Code Verify Email



Gambar 6. Tampilan *Verify Email*

#### 4. Halaman Login

Halaman login di *E-Commerce* Sansekrta, digunakan untuk memasukan akun yang sudah didaftarkan oleh user melalui registrasi. Untuk melakukan login, pengguna atau user harus memasukkan email dan password yang sudah terdaftar pada halaman registrasi sebelumnya. Berikut penjelasan dari *source code* dari halaman *Login*.

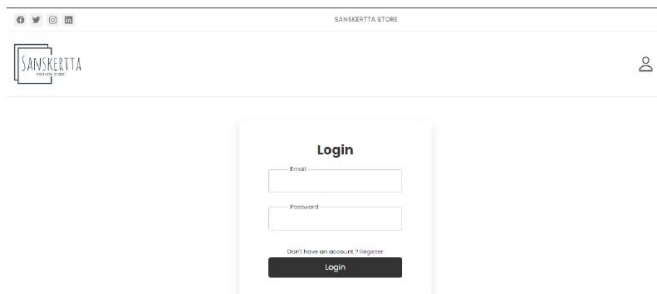
*Controller* ini menggunakan *trait AuthenticatesUsers* untuk menyediakan fungsionalitas otentikasi yang lengkap. Properti *\$redirectTo* menentukan rute tujuan setelah login berhasil, yaitu *HOME*. Constructor memastikan hanya pengguna yang belum terautentikasi yang dapat mengakses metode *login*, sementara metode *logout* dapat diakses oleh pengguna yang sudah terautentikasi.

```

1  <?php
2
3  namespace App\Http\Controllers\Auth;
4
5  use App\Http\Controllers\Controller;
6  use Illuminate\Foundation\Auth\AuthenticatesUsers;
7
8  class LoginController extends Controller
9  {
10
11     //
12     // Login Controller
13     //
14
15     /**
16      * This controller handles authenticating users for the application and
17      * redirecting them to your home screen. The controller uses a trait
18      * to conveniently provide its functionality to your applications.
19      */
20
21     //
22     use AuthenticatesUsers;
23
24     //
25     // Where to redirect users after login.
26     //
27     public $redirectTo;
28
29     protected $redirectTo = RouteServiceProvider::HOME;
30
31     /**
32      * Create a new controller instance.
33      *
34      * @return void
35      */
36     public function __construct()
37     {
38         $this->redirectTo = config('auth.logout');
39     }
40
41 }

```

Gambar 7. *Source Code Login*



Gambar 8. Tampilan *Login*

### 5. Halaman *Profile User*

Halaman *profile user* ini, akan tampil beberapa menu seperti pada gambar dibawah :

- Profile, terdapat data user profile dihalaman ini harus wajib diisi, dengan mengisi semua data yang ada di menu profile ini, maka user bisa menggunakan akun tersebut untuk melakukan pembelian di *E-Commerce Sanskerta*.
- Cart, cart di halaman ini bertujuan untuk menampilkan produk yang sudah ditambahkan user.
- Order, halaman order ini akan menampilkan data order dari halaman cart yang telah diorder oleh user.
- Payments, payment atau pembayaran disini menampilkan deskripsi dari menu order untuk dilakukan pembayaran.
- Shipment, shipment atau pengiriman disini bertujuan untuk menampilkan produk yang sudah berhasil di order dan menampilkan deskripsi pengiriman sampai dengan nomor resi.
- Point, point disini adalah fitur gamifikasinya, jadi *E-Commerce Sanskerta* itu menerapkan sistem gamifikasi yaitu, loyalty program. Point loyalty program ini bisa didapatkan ketika membeli produk, dan ketika user sudah membeli produk maka akan mendapatkan point 10% dari harga pembelian produk di *E-Commerce Sanskerta*. Dimana Point ini bisa digunakan sebagai diskon untuk user ketika ingin membeli produk di Sanskerta.





Profile | Cart | Order | Payment | Shipment

**Name**

**Email**

**Password**

**Photo**  
 No file chosen

**Province**

**City**

**Phone**

**Postal Code**

**Gender**

**Detail Address**

Points Kamu : 750  
[Shop](#)  
[Logout](#)

Gambar 10. Tampilan *Profile User*

Model Discount di Laravel mewakili tabel discounts dalam database. Model ini memiliki beberapa atribut yang dapat diisi secara massal (id, order\_id, discount, discount\_origin) dan mendefinisikan relasi "banyak ke satu" dengan model Order. Ini berarti setiap diskon terkait dengan satu pesanan tertentu. Model ini juga menggunakan trait HasFactory untuk mendukung pembuatan instance model dalam pengujian atau seeding database.

```

1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Discount extends Model
9 {
10     use HasFactory;
11
12     protected $fillable = [
13         'id',
14         'order_id',
15         'discount',
16         'discount_origin',
17     ];
18
19     public function order(){
20         return $this->belongsTo(Order::class, 'order_id');
21     }
22 }
23

```

Gambar 11. *Source Code Point*

Point Kamu : 2500

Gambar 12. Tampilan *Point*

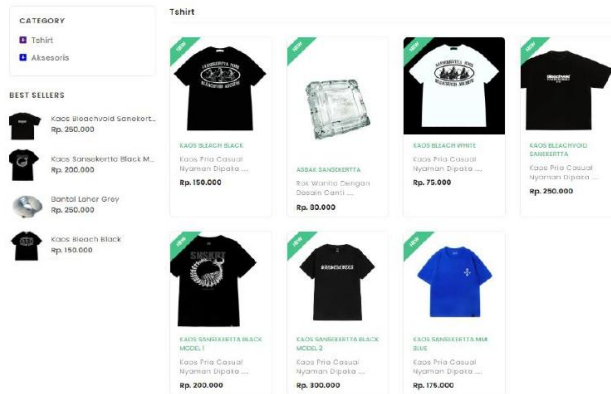
## 6. Halaman Kategori Produk

Halaman ini menampilkan halaman produk yang berada di dalam kategori produk. Berikut penjelasan dan gambar dari source code models dari halaman category:

Model Category di Laravel mewakili tabel categories dalam database. Model ini menggunakan code sebagai primary key dan mendefinisikan relasi one-to-many dengan model Product. Atribut code, name, dan description dapat diisi secara massal. Fungsi generateCode digunakan untuk menghasilkan kode kategori baru dengan format 'CATxx'. Model ini juga menggunakan trait HasFactory untuk mendukung pembuatan instance model dalam pengujian atau seeding database.

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Category extends Model
9 {
10     use HasFactory;
11
12     protected $fillable = [
13         'code',
14         'name',
15         'description'
16     ];
17     protected $primaryKey = 'code';
18     public $incrementing = false;
19     protected $keyType = 'string';
20
21     public function products()
22     {
23         return $this->hasMany(Product::class, 'category_code');
24     }
25
26     public static function generateCode()
27     {
28         $categories = Category::all();
29         $snowNum = 0;
30         foreach($categories as $c){
31             $currentNum = (int) substr($c->code, 3);
32             if($snowNum < $currentNum){
33                 $snowNum = $currentNum;
34             }
35         }
36         $snowCodeNumber = $snowNum + 1;
37         $snowCode = 'CAT' . str_pad($snowCodeNumber, 2, '0', STR_PAD_LEFT);
38
39         return $snowCode;
40     }
41 }
```

Gambar 13. *Source Code* Kategori Produk



GAMBAR 14. HALAMAN KATEGORI PRODUK

### 7. Detail Produk

Halaman detail produk akan menampilkan halaman detail dari produk untuk melihat deskripsi produk, harga, dan stock produk.

Berikut adalah penjelasan dan gambar dari source code dari halaman detail produk :

Model Product di Laravel mewakili tabel products dalam database. Model ini menggunakan code sebagai primary key dan mendefinisikan relasi many-to-one dengan model Category, serta relasi one-to-many dengan model Cart dan ProductPhoto. Atribut code, category\_code, name, photo, price, weight, stock, dan description dapat diisi secara massal. Fungsi generateCode digunakan untuk menghasilkan kode produk baru dengan format 'PRDxx'. Model ini juga menggunakan trait HasFactory untuk mendukung pembuatan instance model dalam pengujian atau seeding database.



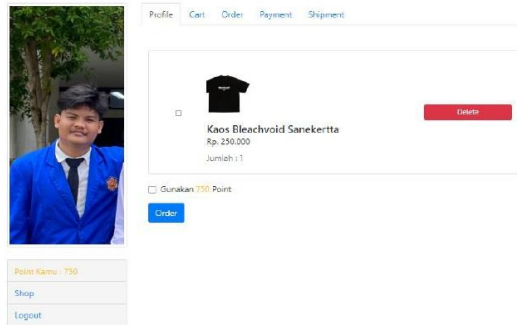
### 8. Halaman Cart

Untuk mengakses halaman cart, klik gambar berikut pada halaman *dashboard*.



Gambar. 17 *Cart Icon*

Setelah memasuki halaman cart akan tampil halaman cart seperti gambar berikut



Gambar 18. Tampilan *Cart*

Isi dari halaman cart adalah foto produk beserta harganya, dan ketika user mempunyai point, maka user bisa menggunakan point dari user untuk ditukarkan dengan potongan harga.

Berikut adalah code source code dari halaman cart, dimana ada 2 fungsi yaitu yang pertama menambahkan kerdalam keranjang dan hapus dari keranjang, berikut gambar dan penjelasan dari halaman cart:

```

1 public Function addtocart(Request $req)
2 {
3     $requestData = [
4         'product_code' => 'request:string',
5         'amount' => 'request:integer|min:1'
6     ];
7
8     $productCode = $req->product_code;
9     $amount = $req->amount;
10    $user = Auth::user();
11
12    // Cek apakah produk ada
13    $product = Product::find($productCode);
14
15    if (!$product) {
16        return response()->json(['message' => 'Product not found'], 404);
17    }
18
19    // Cek stok produk
20    if ($product->stock < $amount) {
21        return response()->json(['message' => 'Insufficient stock'], 400);
22    }
23
24    // Cek apakah produk sudah ada di keranjang
25    $cartItem = Cart::where('user_id', $user->id)
26        ->where('product_code', $productCode)
27        ->first();
28
29    if ($cartItem) {
30        // Update jumlah produk di keranjang
31        $cartItem->amount = $amount;
32        $cartItem->save();
33    } else {
34        // Tambahkan produk baru ke keranjang
35        $cartItem = new Cart();
36        $cartItem->user_id = $user->id;
37        $cartItem->product_code = $productCode;
38        $cartItem->amount = $amount;
39        $cartItem->save();
40    }
41
42    // Kurangi stok produk
43    // $product->stock = $amount;
44    // $product->save();
45
46    return response()->json(['message' => 'Product added to cart successfully'], 200);
47 }
48
49 public function deletefromcart(Request $request)
50 {
51     $request->validate([
52         'id' => 'required|integer'
53     ]);
54
55     if ($request->has('user_id' == Auth::user())) {
56         $cartItem = Cart::find($request->id);
57
58         if ($cartItem) {
59             return response()->json(['success' => true, 'message' => 'Item removed from cart'], 200);
60         }
61
62         return response()->json(['success' => false, 'message' => 'Item not found or not authorized'], 404);
63     }
64 }

```

Gambar 19. Source Code Cart

Fungsi addtocart:

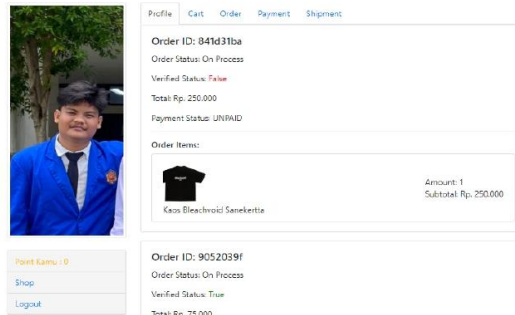
- a. Memvalidasi input request untuk memastikan product\_code dan amount disediakan.
- b. Mengambil kode produk dan jumlah dari request.
- c. Memastikan produk dengan kode tersebut ada di database.
- d. Memastikan stok produk mencukupi untuk jumlah yang diminta.
- e. Memeriksa apakah produk sudah ada di keranjang pengguna.
- f. Jika sudah ada, menambah jumlah produk di keranjang.
- g. Jika belum, menambahkan produk baru ke keranjang.
- h. Mengembalikan respon JSON yang menyatakan bahwa produk berhasil ditambahkan ke keranjang.

Fungsi deletefromcart:

- a. mengambil ID item keranjang dari request.
- b. Memastikan item keranjang ada dan milik pengguna yang sedang login.
- c. Menghapus item dari keranjang jika valid.
- d. Mengembalikan respon JSON yang menyatakan bahwa item berhasil dihapus atau item tidak ditemukan/tidak sah.

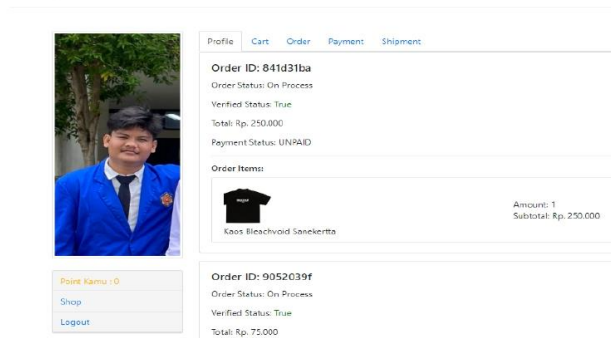
### 9. Halaman Order

Pada halaman order ini, user perlu menunggu verifikasi dari admin untuk bisa melanjutkan ke halaman pembayaran. Berikut adalah gambar ketika tampilan dimenu order yang belum di verifikasi oleh admin dimana verified statusnya adalah false.



Gambar 20. Tampilan Menu Order Yang Belum Terverifikasi

Ketika admin sudah melakukan verifikasi order pada halaman admin, maka tampilan pada menu order pada verified status berubah menjadi true, seperti gambar berikut ini:



Gambar 21. Tampilan Menu Order Yang Sudah Terverifikasi

Berikut adalah source code dan penjelasan pada halaman order :

```

1 public function makeorder(Request $request){
2     $cartId = $request->input('cart_id');
3     $user = Auth::user();
4     $orderId = Str::uuid();
5     // $discount = $user->details->point;
6     // return [$request->input($id), $discount];
7
8     if (empty($cartId)) {
9         // Create new order
10        $order = new Order();
11        $order->id = $orderId;
12        $order->user_id = $user->id;
13        $order->status = 'On Process';
14        $order->total = 0; // Initialize total
15        // $order->payment_status = 'unpaid';
16        $order->save();
17
18        $total = 0;
19
20        foreach ($cartId as $cartId) {
21            $cartItem = Cart::find($cartId);
22            if ($cartItem && $cartItem->user_id == $user->id) {
23                // Add cart item to order items
24                $orderDetail = new OrderDetail();
25                $orderDetail->order_id = $order->id;
26                $orderDetail->product_code = $cartItem->product_code;
27                $orderDetail->amount = $cartItem->amount;
28                $orderDetail->subtotal = $cartItem->amount * $cartItem->product->price;
29                $orderDetail->save();
30
31                // Update total order amount
32                $total += $orderDetail->subtotal;
33
34                // Remove item from cart
35                $cartItem->delete();
36            }
37        }
38
39        // Update order total
40        $order->total = $total;
41        $order->save();
42
43        $discount = new Discount();
44        $discount->order_id = $order->id;
45        $discount->discount = $user->details->point == 0 ? 0 : $user->details->point;
46        $discount->discount_origin = "From User Point";
47        $discount->save();
48
49        $userDetail = UserDetail::where('user_id', $user->id)->first();
50        $userDetail->point = 0;
51        $userDetail->save();
52
53        return response()->json(['success' => true, 'message' => 'Order placed successfully']);
54    }
55
56    return response()->json(['success' => false, 'message' => 'No items selected'], 400);
57 }

```

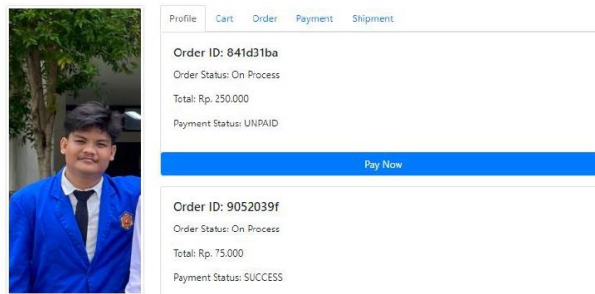
Gambar 22 Gambar Source Code Order

Fungsi makeorder menangani pembuatan pesan baru dengan mengambil item dari keranjang pengguna, mengkalkulasi total pesanan, menerapkan diskon berdasarkan poin pengguna, dan dapat menghapus item dari keranjang. Respon JSON dikembalikan untuk menginformasikan keberhasilan atau kegagalan proses pembuatan pesanan.



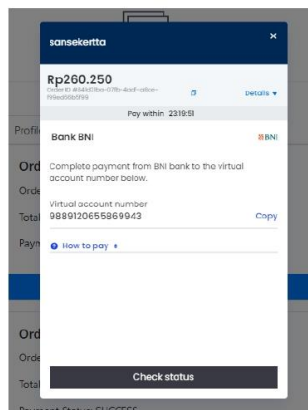
### 10. Pembayaran

Pada halaman pembayaran, ketika order produk dari user sudah di verifikasi oleh admin. Setelah order produk di verifikasi oleh admin, user bisa langsung melakukan pembayaran dengan beberapa metode pembayaran yang tersedia di *E-Commerce* Sanssekerta. Berikut adalah gambar ketika akan melakukan pembayaran :



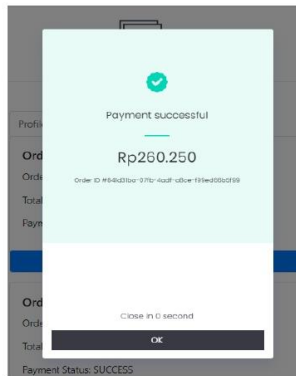
Gambar 23. Tampilan Pembayaran

Selanjutnya pilih Bank untuk melakukan pembayaran, seperti gambar berikut :



Gambar 24. Halaman Pemilihan Bank

Setelah melakukan pembayaran, maka akan menampilkan *payment succesful*, seperti gambar berikut



Gambar 25. Halaman *Payment Successful*

Berikut adalah source code dan penjelasan mengenai halaman pembayaran atau payment:

```

1 public function payorder(Request $req){
2     $order = Order::findOrFail($req->order_id);
3     $snapToken = $order->snap_token;
4     if (is_null($snapToken)) {
5         $midtrans = new CreateSnapTokenService($order);
6         $snapToken = $midtrans->getSnapToken();
7         $order->snap_token = $snapToken;
8         $order->save();
9     }
10    return response()->json(['order' => $order, 'snapToken' => $snapToken]);
11 }
12 }

```

Gambar 26 *Source Code Payment*

Fungsi `payorder` menangani proses pembayaran untuk suatu pesanan dengan langkah-langkah berikut:

- a. Mengambil pesanan berdasarkan `order_id` dari request.
- b. Memeriksa apakah pesanan sudah memiliki `snap_token`. Jika tidak, membuat Snap Token baru menggunakan layanan `CreateSnapTokenService`.
- c. Menyimpan Snap Token yang baru ke dalam pesanan.
- d. Mengembalikan respon JSON yang berisi informasi pesanan dan Snap Token.
- e. Fungsi ini memastikan bahwa setiap pesanan yang belum memiliki Snap Token akan dibuatkan Snap Token baru untuk keperluan pembayaran, dan kemudian mengembalikan informasi tersebut dalam format JSON.

## 11. Pengiriman/Shipment

Setelah melakukan pembayaran berhasil, user dapat mengakses info paket pada menu pengiriman. Pada halaman ini menggunakan fungsi dari API rajaongkir yang ditampilkan pada halaman shipment atau pengiriman. Berikut adalah tampilan shipment atau pengiriman :



Gambar 27. Tampilan Info Paket

Berikut adalah source code dan penjelasan dari halaman shipment atau pengiriman:

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Shipment extends Model
9 {
10     use HasFactory;
11
12     protected $fillable = [
13         'id',
14         'order_id',
15         'price',
16         'courier',
17         'estimate',
18         'service',
19         'resi'
20     ];
21 }
22
```

Gambar 28. Source Code Shipment

Model Shipment di Laravel mewakili tabel shipments dalam database dan digunakan untuk menyimpan informasi pengiriman. Model ini menggunakan trait HasFactory untuk mendukung pembuatan instance model melalui factory. Atribut yang dapat diisi secara massal (\$fillable) termasuk id, order\_id, price, courier, estimate, service, dan resi. Ini memungkinkan pengguna untuk dengan mudah menyimpan dan mengelola data pengiriman dalam aplikasi Laravel.



## Admin

### 1. Login

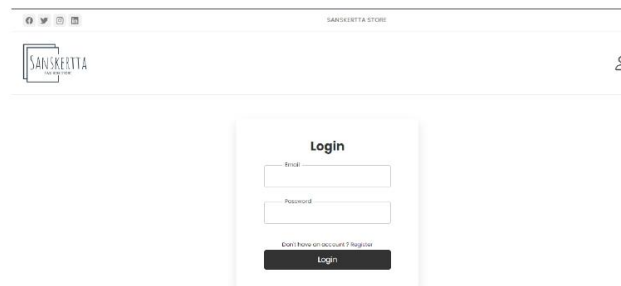
Di Sanskertta, admin untuk masuk ke dalam aplikasi harus melakukan login.

Berikut adalah gambar dan penjelasan dari source code login :

```
1 <?php
2
3 namespace App\Http\Controllers\Auth;
4
5 use App\Http\Controllers\Controller;
6 use Illuminate\Foundation\Auth\AuthenticatesUsers;
7
8 class LoginController extends Controller
9 {
10
11     /**
12      * Login Controller
13      *
14      * This controller handles authenticating users for the application and
15      * redirecting them to your home screen. The controller uses a trait
16      * to conveniently provide its functionality to your applications.
17      *
18      * @return void
19      */
20 }
21
22 use AuthenticatesUsers;
23
24 /**
25  * Where to redirect users after login.
26  *
27  * @var string
28  */
29 protected $redirectTo = RouteServiceProvider::HOME;
30
31 /**
32  * Create a new controller instance.
33  *
34  * @return void
35  */
36 public function __construct()
37 {
38     $this->middleware('guest')->only('login');
39 }
40
41 }
```

Gambar 30. Source Code Admin Login

Controller ini menggunakan trait `AuthenticatesUsers` untuk menyediakan fungsionalitas otentikasi yang lengkap. Properti `$redirectTo` menentukan rute tujuan setelah login berhasil, yaitu `HOME`. Constructor memastikan hanya pengguna yang belum terotentikasi yang dapat mengakses metode login, sementara metode logout dapat diakses oleh pengguna yang sudah terotentikasi.



Gambar 31. Tampilan Login

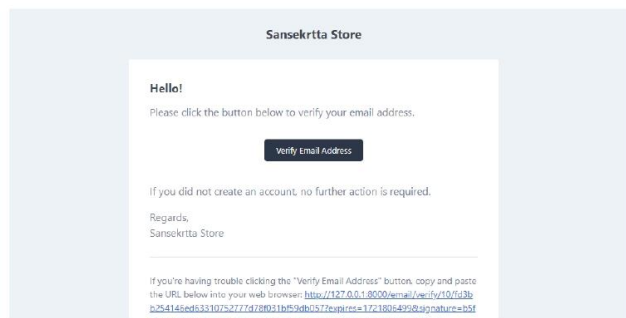
### 2. Verifikasi Email

Setelah melakukan registrasi menggunakan akun email di website E-

Commerce maka, akun email tersebut akan otomatis mendapatkan verifikasi pada email yang telah didaftarkan pada akun di website *E-Commerce*. Selanjutnya klik verify email address, maka akan otomatis ke redirect ke halaman login.

Berikut adalah gambar verifikasi email Berikut penjelasan mengenai source code verifikasi email:

- a. namespace App\Http\Controllers\Auth; mendefinisikan namespace untuk controller otentikasi, sesuai konvensi Laravel.
- b. use statements mengimpor kelas-kelas yang dibutuhkan dalam controller.
- c. Kelas VerificationController mewarisi dari Controller dasar Laravel.
- d. Trait VerifiesEmails menyediakan fungsionalitas untuk verifikasi email dan pengiriman ulang email verifikasi.
- e. Properti \$redirectTo menentukan rute tujuan setelah verifikasi email berhasil, yaitu HOME.
- f. Constructor dipanggil saat instance VerificationController dibuat.
- g. \$this->middleware('auth'); menetapkan middleware auth pada semua metode dalam controller, memastikan hanya pengguna terotentikasi yang dapat mengaksesnya.
- h. \$this->middleware('signed')->only('verify'); menetapkan middleware signed pada metode verify, memastikan URL verifikasi ditandatangani secara digital.
- i. \$this->middleware('throttle:6,1')->only('verify', 'resend'); menetapkan middleware throttle pada metode verify dan resend, membatasi jumlah permintaan maksimal 6 kali per menit untuk mencegah penyalahgunaan.



Gambar 32. Tampilan Verifikasi Email



### 3. Halaman Dashboard Admin

Halaman ini merupakan halaman pembuka untuk halaman *dashboard* admin yang dimana disini akan menampilkan total user, total kategori, total produk, dan total order. Berikut adalah penjelasan dari source code halaman dashboard dan tampilannya

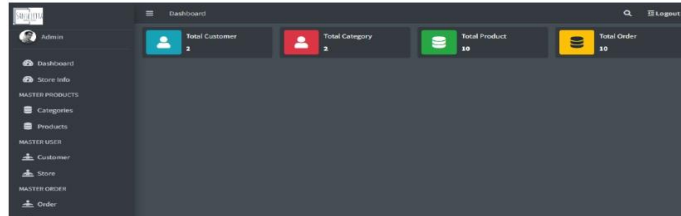
Kode ini menyediakan tiga metode utama:

- index: Mengambil data pelanggan, toko, kategori, produk, dan pesanan untuk ditampilkan di dashboard admin.
- storeinfo: Mengambil dan menampilkan informasi toko.
- storeinfoupdate: Mengupdate dan menyimpan informasi toko berdasarkan data yang diterima dari permintaan (request).

```
1 <?php
2
3 namespace App\Http\Controllers\Admin;
4
5 use App\Http\Controllers\Controller;
6 use Illuminate\Http\Request;
7 use App\Models\User;
8 use App\Models\Category;
9 use App\Models\Product;
10 use App\Models\Order;
11 use App\Models\StoreInfo;
12
13 class DashboardController extends Controller
14 {
15     public function index(){
16         $customer = User::role('customer')->get();
17         $store = User::role('store')->get();
18         $category = Category::all();
19         $product = Product::all();
20         $order = Order::all();
21
22         return view('admin.index', compact(['customer', 'store', 'category', 'product', 'order']));
23     }
24
25     public function storeinfo(){
26         $storeinfo = StoreInfo::find(1);
27
28         return view('admin.storeinfo', compact(['storeinfo']));
29     }
30
31     public function storeinfoupdate(Request $req){
32         $storeinfo = StoreInfo::find(1);
33
34         $storeinfo->name = $req->name;
35         $storeinfo->province = $req->province_name;
36         $storeinfo->province_code = $req->province;
37         $storeinfo->city = $req->city_name;
38         $storeinfo->city_code = $req->city;
39         $storeinfo->address = $req->address;
40         $storeinfo->phone = $req->phone;
41         $storeinfo->email = $req->email;
42
43         $storeinfo->save();
44
45         return redirect()->route('storeinfo');
46     }
47
48 }
```

Gambar 34. Source Code Dashboard Admin





Gambar 35. Tampilan *Dashboard Admin*

#### 4. Store Info

Pada halaman ini akan menampilkan data dari alamat toko Sansekertta untuk menentukan pengiriman dari lokasi toko ketempat tujuan user, hal ini wajib karena dengan menambahkan data toko akan memudahkan ongkos kirim untuk pembelian produk pada toko E-Commerce Sansekertta.

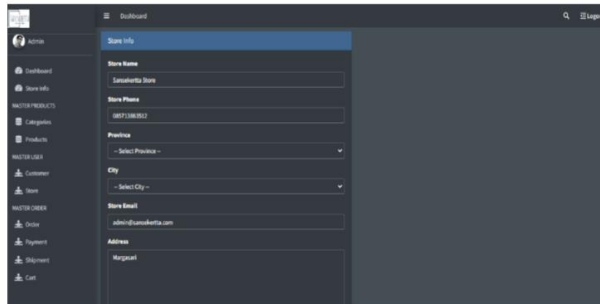
Berikut adalah tampilan dan source code pada halaman store info: Karena halaman store info dibutuhkan untuk melakukan perhitungan jarak tempuh antara alamat toko dan rumah user, maka di halaman store info perlu menggunakan API Raja Ongkir.

```

1 public function storeinfo(){
2     $storeinfo = StoreInfo::find(1);
3
4     return view('admin.storeinfo', compact(['storeinfo']));
5 }
6
7 public function storeinfoupdate(Request $req){
8     $storeinfo = StoreInfo::find(1);
9
10    $storeinfo->name = $req->name;
11    $storeinfo->province = $req->province_name;
12    $storeinfo->province_code = $req->province;
13    $storeinfo->city = $req->city_name;
14    $storeinfo->city_code = $req->city;
15    $storeinfo->address = $req->address;
16    $storeinfo->phone = $req->phone;
17    $storeinfo->email = $req->email;
18
19    $storeinfo->save();
20
21    return redirect()->route('storeinfo');
22 }
23 }
24

```

Gambar 36. *Source Code Store Info*



Gambar 37. Tampilan *Store Info*

### 5. Halaman Kategori

Halaman Kategori, pada halaman admin berfungsi untuk melakukan penambahan data untuk kategori, mengedit data kategori, dan menghapus data kategori

Berikut adalah tampilan dan source code dari halaman kategori:

Kode ini mengandung metode-metode utama untuk mengelola kategori di halaman admin:

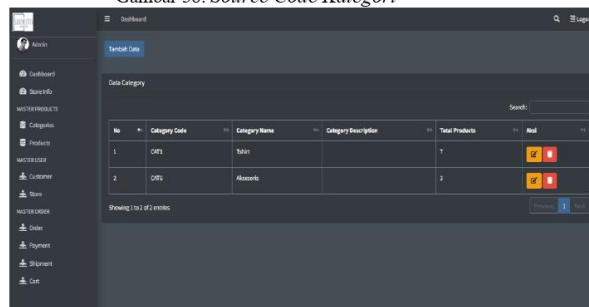
- index: Mengambil dan menampilkan semua kategori.
- add: Menampilkan form untuk menambah kategori baru dengan kode yang dihasilkan.
- create: Membuat dan menyimpan kategori baru.
- edit: Mengambil data kategori dan menampilkan form untuk mengedit kategori tersebut.
- update: Mengupdate dan menyimpan data kategori yang sudah ada.
- delete: Menghapus kategori berdasarkan kode yang diterima.

```

1 <?php
2
3 namespace App\Http\Controllers\Admin;
4
5 use Illuminate\Http\Request;
6 use Illuminate\View\View;
7 use App\Models\Category;
8
9 class CategoryController extends Controller
10 {
11     public function index()
12     {
13         $category = Category::first();
14         return view('admin.category.index', compact('category'));
15     }
16
17     public function add()
18     {
19         $code = Category::generateCode();
20         return view('admin.category.add', compact('code'));
21     }
22
23     public function create(Request $req)
24     {
25         $category = new Category();
26         $category->code = $req->code;
27         $category->name = $req->name;
28         $category->description = $req->description;
29         $category->save();
30     }
31     return redirect()->route('admin.category.index');
32
33     public function edit(Request $req)
34     {
35         $category = Category::findOrFail($req->code);
36         return view('admin.category.edit', compact('category'));
37     }
38
39     public function update(Request $req)
40     {
41         $category = Category::findOrFail($req->code);
42         $category->name = $req->name;
43         $category->description = $req->description;
44         $category->save();
45     }
46     return redirect()->route('admin.category.index');
47
48     public function delete(Request $req)
49     {
50         $category = Category::findOrFail($req->code);
51         $category->delete();
52     }
53     return redirect()->route('admin.category.index');
54 }
55
56 }

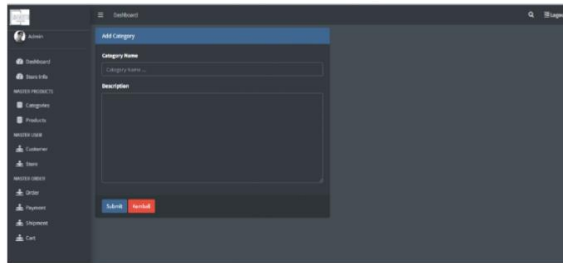
```

Gambar 38. Source Code Kategori



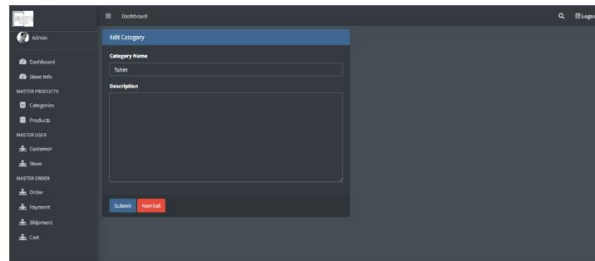
Gambar 39. Tampilan Kategori

Pada halaman kategori ini admin dapat menambahkan data kategori seperti pada gambar berikut :



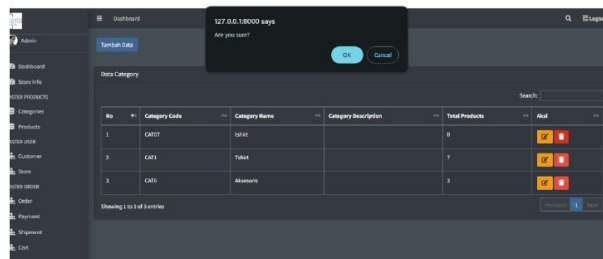
Gambar 40 Halaman Data Kategori

Selain bisa menambahkan kategori, halaman ini bisa melakukan edit kategori, klik halaman edit kategori, isi edit kategori yang ingin diubah, lalu klik submit untuk melakukan penyimpanan data edit kategori. Berikut adalah gambar dari edit produk:



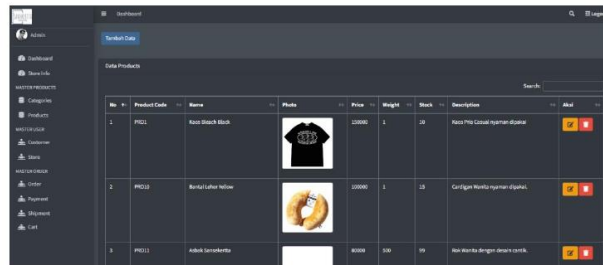
Gambar 41. Halaman Edit Kategori

Selain bisa menambahkan dan edit kategori, halaman kategori untuk admin ini bisa melakukan delete atau hapus data kategori, klik icon delete dan akan muncul alert seperti gambar berikut :



Gambar 42. Halaman *Delete* Atau Hapus Data Kategori





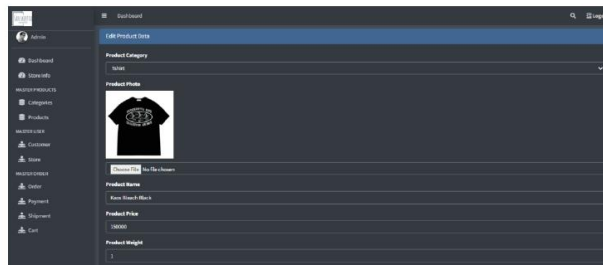
Gambar 44. Halaman Produk

Halaman Tambah produk, untuk mengakses halaman ini, klik tambah produk yang ada di halaman data produk, berikut halaman tambah produk dapat dilihat pada gambar :



Gambar 45. Halaman Tambah produk

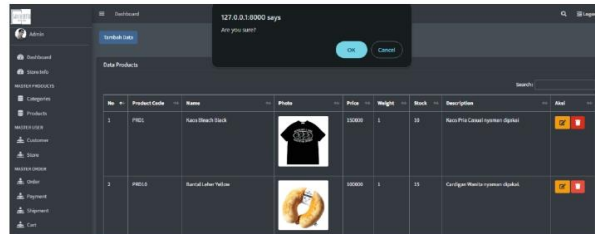
Halaman edit produk, halaman ini sesuai namanya dapat berguna untuk mengedit produk, dengan cara klik icon edit dan isi bagian yang akan diedit. Berikut adalah gambar dari edit produk:



Gambar 46. Halaman Edit Produk

Selain bisa menambahkan dan edit produk, halaman produk untuk admin ini bisa melakukan delete atau hapus data produk, klik icon delete dan akan muncul

alert seperti gambar berikut



Gambar 47. Halaman *Delete* Atau Hapus Data Produk

### 7. *Halaman Data Customer*

Halaman ini akan menampilkan customer atau user yang menggunakan E Commerce Sansekertta. Berikut adalah source code dan tampilan dari halaman data customer beserta penjelasannya :

Kode ini mengandung metode-metode utama untuk mengelola pelanggan di halaman admin:

- index: Mengambil dan menampilkan semua pelanggan.
- update: Mengupdate dan menyimpan data pelanggan berdasarkan data dari permintaan, lalu mengarahkan kembali pengguna ke halaman `admin.customer.index`.
- delete: Menghapus pelanggan berdasarkan ID dari permintaan dan mengarahkan kembali pengguna ke halaman `admin.customer.index`.

```

1 <?php
2
3 namespace App\Http\Controllers\Admin;
4
5 use App\Http\Controllers\Controller;
6 use Illuminate\Http\Request;
7 use App\Models\User;
8 use Illuminate\Support\Facades\Hash;
9
10 class CustomerController extends Controller
11 {
12     public function index()
13     {
14         $customers = User::role('customer')->get();
15         return view('admin.customer.index', compact('customers'));
16     }
17
18     public function add()
19     {
20         return view('admin.customer.add');
21     }
22
23     public function create(Request $req)
24     {
25         $customer = new User;
26         $customer->name = $req->name;
27         $customer->email = $req->email;
28         $customer->password = Hash::make($req->password);
29         $customer->save();
30         $customer->assignRole('pelanggan');
31         return redirect()->route('admin.customer.index');
32     }
33
34     public function edit(Request $req)
35     {
36         $customer = User::findOrFail($req->id);
37         return view('admin.customer.edit', compact('customer'));
38     }
39
40     public function update(Request $req)
41     {
42         $customer = User::findOrFail($req->id);
43         $customer->name = $req->name;
44         $customer->email = $req->email;
45         $customer->password = Hash::make($req->password);
46         $customer->save();
47         return redirect()->route('admin.customer.index');
48     }
49
50     public function delete(Request $req)
51     {
52         User::destroy($req->id);
53         return redirect()->route('admin.customer.index');
54     }
55 }
56
57
58

```

Gambar 48. Source Code Data Customer

Data Customer

No	Nama	Email	Point
1	Pelanggan Satu	pelanggan@mail.id	10000

Showing 1 to 1 of 1 entries

Gambar 49. Halaman Data Customer



## 8. Halaman Cart

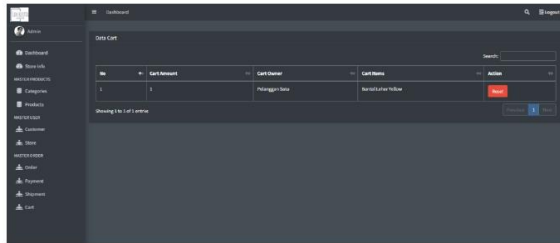
Pada halaman ini akan menampilkan data dari user yang sudah memasukkan produk ke dalam keranjang. Berikut adalah source code dan tampilan pada halaman cart :

Kode ini mengandung metode-metode utama untuk mengelola keranjang belanja di halaman admin:

- index: Mengambil dan menampilkan semua keranjang belanja.
- detail: Mengambil data keranjang belanja berdasarkan ID yang diterima dan menampilkan detail keranjang belanja tersebut.
- verif: Metode untuk melakukan verifikasi keranjang belanja, tetapi saat ini belum diimplementasikan.
- delete: Menghapus keranjang belanja berdasarkan ID yang diterima dan mengarahkan kembali pengguna ke halaman admin.cart.index.

```
1 <?php
2
3 namespace App\Http\Controllers\Admin;
4
5 use App\Http\Controllers\Controller;
6 use Illuminate\Http\Request;
7 use App\Models\Cart;
8
9 class CartController extends Controller
10 {
11     public function index(){
12         $cart = Cart::all();
13
14         return view('admin.cart.index', compact(['cart']));
15     }
16
17     public function detail($id){
18         $cart = Cart::findOrFail($id);
19
20         return view('admin.cart.detail', compact(['cart']));
21     }
22
23     public function verif(Request $req){
24
25     }
26
27     public function delete($id){
28         $cart = Cart::findOrFail($id);
29         if($cart){
30             $cart->destroy($id);
31         }
32
33         return redirect()->route('admin.cart.index');
34     }
35 }
36
```

Gambar 50. Source Code Cart



Gambar 51. Halaman *Cart*

### 9. Halaman *Order*

Pada halaman order atau pembayaran ini, menampilkan data dari order yang sudah dilakukan oleh user pada saat akan membeli produk. Disini admin bertugas mengecek status order, verifikasi order, pengiriman, dan hapus order.

Berikut adalah halaman order ketika admin belum menyiapkan pembayaran:

ID	Order Status	Order Total	Order Payment Status	Verified	Action
1	On Process	75000	UNPAID	False	Delete
2	On Process	10000	UNPAID	False	Delete
3	On Process	0	UNPAID	False	Delete

Gambar 52. Halaman *Order*

Pada proses ini admin akan melakukan verifikasi pada order yang masuk, disini tugas admin akan mengatur pengiriman, mulai dari kurir, servis yang pas untuk produk yang dikirim, setelah selesai selanjutnya klik icon verif dan orderan sudah siap untuk dibayar oleh customer. Setelah customer membayar maka dari *E-Commerce* Sansekerta melakukan pengiriman ke jasa pengiriman yang telah di verif oleh admin, setelah mendapatkan resi maka pada halaman order ini bisa diupdate untuk memasukkan resi pengirimannya, jadi user bisa melakukan lacak paket pada website jasa pengiriman yang terteta. Berikut adalah gambar ketika admin melakukan tugas mengatur pengiriman.

Gambar 53. Halaman Verifikasi *Order*

Setelah admin mengatur pengiriman maka tampilan pada halaman order akan berubah pada column inverified yang menjadi true seperti gambar berikut:

No	Order Status	Order Total	Order Payment Status	inverified	Action
1	On Process	25000	UNPAID	True	<a href="#">Detail</a>

Gambar 54. Halaman *Order* Terverifikasi

Berikut adalah source code dan penjelasan dari halaman Order atau pembayaran : Kode ini mengandung metode-metode utama untuk mengelola pesanan di halaman admin:

- index: Mengambil dan menampilkan semua pesanan.
- detail: Mengambil data pesanan dan informasi toko berdasarkan ID yang diterima dan menampilkan detail pesanan tersebut.
- verif: Memverifikasi pesanan, mengelola data pengiriman, membuat snap\_token jika diperlukan, menyimpan data, dan mengarahkan pengguna kembali ke halaman admin.order.index.
- delete: Menghapus pesanan berdasarkan ID yang diterima dan mengarahkan kembali pengguna ke halaman admin.order.index.

```

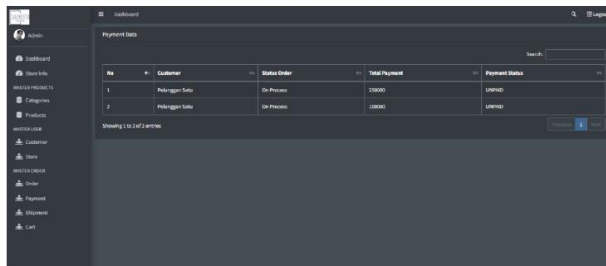
1  *
2  *
3  *
4  *
5  *
6  *
7  *
8  *
9  *
10 *
11 *
12 *
13 *
14 *
15 *
16 *
17 *
18 *
19 *
20 *
21 *
22 *
23 *
24 *
25 *
26 *
27 *
28 *
29 *
30 *
31 *
32 *
33 *
34 *
35 *
36 *
37 *
38 *
39 *
40 *
41 *
42 *
43 *
44 *
45 *
46 *
47 *
48 *
49 *
50 *
51 *
52 *
53 *
54 *
55 *
56 *
57 *
58 *
59 *
60 *
61 *
62 *
63 *
64 *
65 *
66 *
67 *
68 *
69 *
70 *
71 *
72 *
73 *
74 *
75 *
76 *
77 *
78 *
79 *
80 *
81 *
82 *
83 *
84 *
85 *
86 *
87 *
88 *
89 *
90 *
91 *
92 *
93 *
94 *
95 *
96 *
97 *
98 *
99 *
100 *

```

Gambar 55. Source Code Order

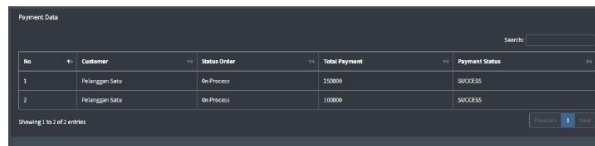
### 10. Halaman Payment

Pada halaman ini menampilkan data dari order yang sedang diproses sampai order yang sudah dibayar oleh user. Berikut adalah tampilan halaman payment



Gambar 56. Halaman Payment

Setelah pembayaran dilakukan oleh user maka, column payment status akan berubah menjadi succes, seperti pada gambar berikut:



The screenshot shows a table titled 'Payment Data' with a search bar at the top right. The table has five columns: 'No', 'Customer', 'Status Order', 'Total Payment', and 'Payment Status'. There are two rows of data. The first row has '1' in the 'No' column, 'Pelanggan Data' in 'Customer', 'On Process' in 'Status Order', '20000' in 'Total Payment', and 'SUCCESS' in 'Payment Status'. The second row has '2' in the 'No' column, 'Pelanggan Data' in 'Customer', 'On Process' in 'Status Order', '10000' in 'Total Payment', and 'SUCCESS' in 'Payment Status'. At the bottom left, it says 'Showing 2 to 2 of 2 entries'. At the bottom right, there are 'Previous' and 'Next' buttons.

No	Customer	Status Order	Total Payment	Payment Status
1	Pelanggan Data	On Process	20000	SUCCESS
2	Pelanggan Data	On Process	10000	SUCCESS

Gambar 57. Halaman *Payment Succes*

Berikut adalah source code dan penjelasan dari halaman payment atau pembayaran.

Terdapat dua controller untuk melakukan pembayaran, yang pertama Pesanan Controller dan Payment Controller

PesananController :

- Metode index mengambil dan menampilkan semua pesanan.
- Metode add menampilkan formulir untuk menambahkan pesanan.
- Metode generateKode menghasilkan kode unik untuk pesanan.
- Metode create membuat pesanan baru, menghitung total biaya, dan menyimpan hubungan layanan dengan pesanan.
- Metode view menampilkan detail pesanan berdasarkan ID.
- Metode edit menampilkan formulir untuk mengedit pesanan, termasuk layanan dan pelanggan terkait.
- Metode update memperbarui status pesanan dan memperbarui detail layanan terkait.
- Metode delete menghapus pesanan berdasarkan ID.

PembayaranController:

- Metode index mengambil dan menampilkan semua pesanan yang terkait dengan pembayaran.
- Metode add menampilkan formulir untuk menambahkan transaksi pesanan.
- Metode generateKode menghasilkan kode unik untuk pembayaran.
- Metode edit menampilkan formulir untuk mengedit pembayaran berdasarkan ID pesanan.
- Metode update memperbarui status pesanan dan detail pembayaran, termasuk membuat atau memperbarui entitas Pembayaran.
- Metode delete menghapus pembayaran berdasarkan ID.

```

1  * * * * *
2  * * * * *
3  * * * * *
4  * * * * *
5  * * * * *
6  * * * * *
7  * * * * *
8  * * * * *
9  * * * * *
10 * * * * *
11 * * * * *
12 * * * * *
13 * * * * *
14 * * * * *
15 * * * * *
16 * * * * *
17 * * * * *
18 * * * * *
19 * * * * *
20 * * * * *
21 * * * * *
22 * * * * *
23 * * * * *
24 * * * * *
25 * * * * *
26 * * * * *
27 * * * * *
28 * * * * *
29 * * * * *
30 * * * * *
31 * * * * *
32 * * * * *
33 * * * * *
34 * * * * *
35 * * * * *
36 * * * * *
37 * * * * *
38 * * * * *
39 * * * * *
40 * * * * *
41 * * * * *
42 * * * * *
43 * * * * *
44 * * * * *
45 * * * * *
46 * * * * *
47 * * * * *
48 * * * * *
49 * * * * *
50 * * * * *
51 * * * * *
52 * * * * *
53 * * * * *
54 * * * * *
55 * * * * *
56 * * * * *
57 * * * * *
58 * * * * *
59 * * * * *
60 * * * * *
61 * * * * *
62 * * * * *
63 * * * * *
64 * * * * *
65 * * * * *
66 * * * * *
67 * * * * *
68 * * * * *
69 * * * * *
70 * * * * *
71 * * * * *
72 * * * * *
73 * * * * *
74 * * * * *
75 * * * * *
76 * * * * *
77 * * * * *
78 * * * * *
79 * * * * *
80 * * * * *
81 * * * * *
82 * * * * *
83 * * * * *
84 * * * * *
85 * * * * *
86 * * * * *
87 * * * * *
88 * * * * *
89 * * * * *
90 * * * * *
91 * * * * *
92 * * * * *
93 * * * * *
94 * * * * *
95 * * * * *
96 * * * * *
97 * * * * *
98 * * * * *
99 * * * * *
100 * * * * *

```

Gambar 58. Source Code Pemesanan

```

1  * * * * *
2  * * * * *
3  * * * * *
4  * * * * *
5  * * * * *
6  * * * * *
7  * * * * *
8  * * * * *
9  * * * * *
10 * * * * *
11 * * * * *
12 * * * * *
13 * * * * *
14 * * * * *
15 * * * * *
16 * * * * *
17 * * * * *
18 * * * * *
19 * * * * *
20 * * * * *
21 * * * * *
22 * * * * *
23 * * * * *
24 * * * * *
25 * * * * *
26 * * * * *
27 * * * * *
28 * * * * *
29 * * * * *
30 * * * * *
31 * * * * *
32 * * * * *
33 * * * * *
34 * * * * *
35 * * * * *
36 * * * * *
37 * * * * *
38 * * * * *
39 * * * * *
40 * * * * *
41 * * * * *
42 * * * * *
43 * * * * *
44 * * * * *
45 * * * * *
46 * * * * *
47 * * * * *
48 * * * * *
49 * * * * *
50 * * * * *
51 * * * * *
52 * * * * *
53 * * * * *
54 * * * * *
55 * * * * *
56 * * * * *
57 * * * * *
58 * * * * *
59 * * * * *
60 * * * * *
61 * * * * *
62 * * * * *
63 * * * * *
64 * * * * *
65 * * * * *
66 * * * * *
67 * * * * *
68 * * * * *
69 * * * * *
70 * * * * *
71 * * * * *
72 * * * * *
73 * * * * *
74 * * * * *
75 * * * * *
76 * * * * *
77 * * * * *
78 * * * * *
79 * * * * *
80 * * * * *
81 * * * * *
82 * * * * *
83 * * * * *
84 * * * * *
85 * * * * *
86 * * * * *
87 * * * * *
88 * * * * *
89 * * * * *
90 * * * * *
91 * * * * *
92 * * * * *
93 * * * * *
94 * * * * *
95 * * * * *
96 * * * * *
97 * * * * *
98 * * * * *
99 * * * * *
100 * * * * *

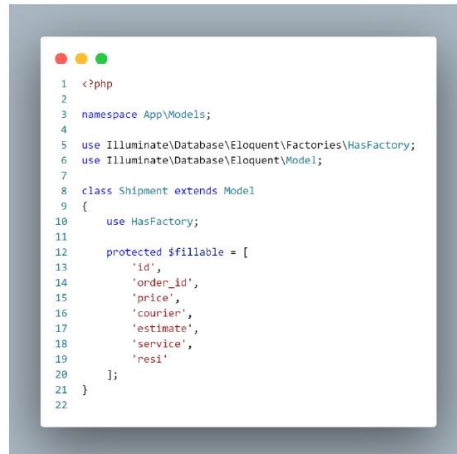
```

Gambar 59. Source Code Pembayaran

### 11. Halaman Shipment atau Pengiriman

Pada halaman ini menampilkan data pengiriman dari pembayaran user yang sudah berhasil.

Berikut adalah source code halaman shipment atau pengiriman:



```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Shipment extends Model
9 {
10     use HasFactory;
11
12     protected $fillable = [
13         'id',
14         'order_id',
15         'price',
16         'courier',
17         'estimate',
18         'service',
19         'resi'
20     ];
21 }
22
```

Gambar 60. *Source Code Shipment*

Model Shipment di Laravel mewakili tabel shipments dalam database dan digunakan untuk menyimpan informasi pengiriman. Model ini menggunakan trait HasFactory untuk mendukung pembuatan instance model melalui factory. Atribut yang dapat diisi secara massal (\$fillable) termasuk id, order\_id, price, courier, estimate, service, dan resi. Ini memungkinkan pengguna untuk dengan mudah menyimpan dan mengelola data pengiriman dalam aplikasi Laravel.

Berikut adalah penerapan dari RajaOngkir pada halaman shipment atau pengiriman seperti berikut :





## Lampiran 5. Sertifikat HKI yang Terbit

  
**REPUBLIK INDONESIA**  
**KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA**

### SURAT PENCATATAN CIPTAAN

Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan : EC00202475206, 31 Juli 2024

**Pencipta**  
Nama : **Sugeng Sulistianto, Muhammad Fikri Hidayattullah, S.T., M.Kom. dkk**

Alamat : Pakulaut Timur RT03 / RW06, Desa Pakulaut Kecamatan Margasari Kab.Tegal Provinsi Jawa Tengah 52463, Margasari, Tegal, Jawa Tengah, 52463

Kewarganegaraan : Indonesia

**Pemegang Hak Cipta**  
Nama : **Pusat Penelitian dan Pengabdian Masyarakat (P3M) Politeknik Harapan Bersama**

Alamat : Jalan Mataram No, 9, Pesurungan Lor, Kecamatan Margadana 52142, Margadana, Tegal, Jawa Tengah 52142

Kewarganegaraan : Indonesia

Jenis Ciptaan : **Program Komputer**

Judul Ciptaan : **Sansekertta Vibe : E-commerce For Future**

Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia : 31 Juli 2024, di Tegal

Jangka waktu perlindungan : Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.

Nomor pencatatan : 000650553

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.  
Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.

a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA  
DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL  
u.b  
Direktur Hak Cipta dan Desain Industri



IGNATIUS M.T. SILALAH  
NIP. 196812301996031001



Disclaimer:  
Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan.

**LAMPIRAN PENCIPTA**

No	Nama	Alamat
1	Sugeng Sulistianto	Pakulaut Timur RT03 / RW06, Desa Pakulaut Kecamatan Margasari Kab.Tegal Provinsi Jawa Tengah 52463, Margasari, Tegal
2	Muhammad Fikri Hidayattullah, S.T., M.Kom.	Jl. Glatik No.68, Randugunting Kec.Tegal Timur Kota Tegal Provinsi Jawa Tengah 52131, Tegal Timur, Tegal
3	Mriza Alim Mutasodirin, S.Kom., M.Kom.	Griya Santika Blok J No 11, Pengabean Kec.Dukuhturi Kab.Tegal Provinsi Jawa Tengah 52192, Dukuhturi, Tegal



Lampiran 6. Lembar Bimbingan Tugas Akhir



**SARJANA TERAPAN TEKNIK INFORMATIKA  
POKITEKNIK HARAPAN BERSAMA**

**LEMBAR BIMBINGAN TUGAS AKHIR**

**Nama** : Sugeng Sulistianto  
**NIM** : 20090091  
**No.Ponsel** : 085713883512  
**Judul TA** : Implementasi Gamification pada E-commerce Berbasis Website untuk Meningkatkan Penjualan (Studi Kasus Brand Sanssekertta)  
**Dosen Pembimbing I** : Muhammad Fikri Hidayattullah, S.T, M.Kom

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
1.	19/4 2024	proposal	1. Pertanyaan penelitian 2. PM 2. Brand Sanssekertta perbandingan dan Mitra (Brand Sanssekertta) 3. Awal Juni wajib submit	
2	27/5	Konsep	Fitur pembatas E-commerce yang ada;	
3	11/6	Protodipe	Gamification harus jelas bisnis prosesnya	



SARJANA TERAPAN TEKNIK INFORMATIKA  
POKITEKNIK HARAPAN BERSAMA

4.	25/7	Prototipe	Stok mesin Jabal	
5.	27/7	Prototipe	Ajumen HKI	
6.	31/7	Prototipe	Uj. fungsionalitas	
7.	1/8	Laporan	1. Tjgs 2. Pem. Cogan 3. Leskopi bab.	
8.	13/8	Laporan	Ujubi Lites Jepri ->	
9.	14/8	Laporan	Ace	

Tegal, 14 Agustus 2024  
Dosen Pembimbing I

Muhammad Fikri Hidayattullah, S.T., M.Kom  
NIPY. 09/016.307



SARJANA TERAPAN TEKNIK INFORMATIKA  
POKITEKNIK HARAPAN BERSAMA

LEMBAR BIMBINGAN TUGAS AKHIR

Nama : Sugeng Sulistianto  
NIM : 20090091  
No.Ponsel : 085713883512  
Judul TA : Implementasi Gamification pada E-commerce Berbasis  
Website untuk Meningkatkan Penjualan(Studi Kasus  
Brand Sansekertta)  
Dosen Pembimbing II : Mirza Alim Mutasodirin, S.Kom., M.Kom

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
1.	16-4-2024	Aplikasi	Boris authentication Sejuta lanjut CRUB	
2.	<del>27-5-2024</del>	<del>Aplikasi</del>	<del>Temp Place</del>	<del></del>
3	21-6-2024	Laporan	- Masalah yang ingin di selesaikan. - Tujuan pembatasan aplikasi. - Manfaat yang diharapkan jika masalah selesai	
4.	30-7-2024	Aplikasi	- progress aplikasi	
5	13-8-2024	Laporan	- Bimbingan Laporan	
6	14-8-2024	Laporan	Laporan Acc.	

Tegal, 14 Agustus 2024  
Dosen Pembimbing II

Mirza Alim Mutasodirin, S.Kom., M.Kom  
NIPY. 03.023.534