

## **BAB III METODOLOGI PENELITIAN**

### **3.1. Tempat Penelitian**

Tempat penelitian yang dilakukan pada Djayataruna *Cafe & Archery* . Dengan terdiri *Cafe* yang cukup Besar secara keseluruhan juga memiliki jumlah Pelanggan yang banyak, hal tersebut tentu perlu memperhatikan pengolahan data *cafe*. Sehingga dapat terciptanya Sistem Yang membantu pengolahan informasi *Cafe*.

### **3.2. Metode Pengumpulan Data**

Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan penelitian. Pengumpulan data dalam penelitian ini menggunakan beberapa metode yaitu:

a. Wawancara (*Interview*)

Wawancara dilakukan dengan cara menanyakan pertanyaan secara langsung kepada pemilik Djayataruna *Cafe & Archery* yakni Saka Fatihan Djayataruna untuk memperoleh informasi mengenai Pengelolaan dana *Cafe*.

b. Dokumentasi (*Documentation*)

Pengumpulan data dilakukan melalui berbagai sumber, termasuk dokumen yang terkait dengan *cafe*. Selain itu, dilakukan pula dokumentasi berupa foto buku menu, produk makanan dan minuman, area panahan, dan area *cafe* guna melengkapi data yang diperlukan

c. Tinjauan Pustaka (*Library Research*)

Mempelajari kumpulan buku-buku yang dilakukan dengan cara membaca literatur-literatur dan tata bahasa yang baik yang ada kaitannya dengan data yang dibutuhkan.

### **3.3. Alat dan Bahan Penelitian**

Alat penelitian merupakan bentuk pendukung dalam penelitian yang terdiri dari perangkat lunak dan perangkat keras sebagai berikut :

#### **3.3.1. Perangkat Keras**

1. *Prosesor Intel® Core™ i5-3330 Cache 6M, hingga 3,20 GHz*
2. *Memory RAM 4 GB*
3. *Harddisk 500 G*
4. *Monitor 14 inchi*
5. *Keyboard*
6. *Mouse*
7. *Logitech C270 HD Webcam*

#### **3.3.2 Perangkat Lunak**

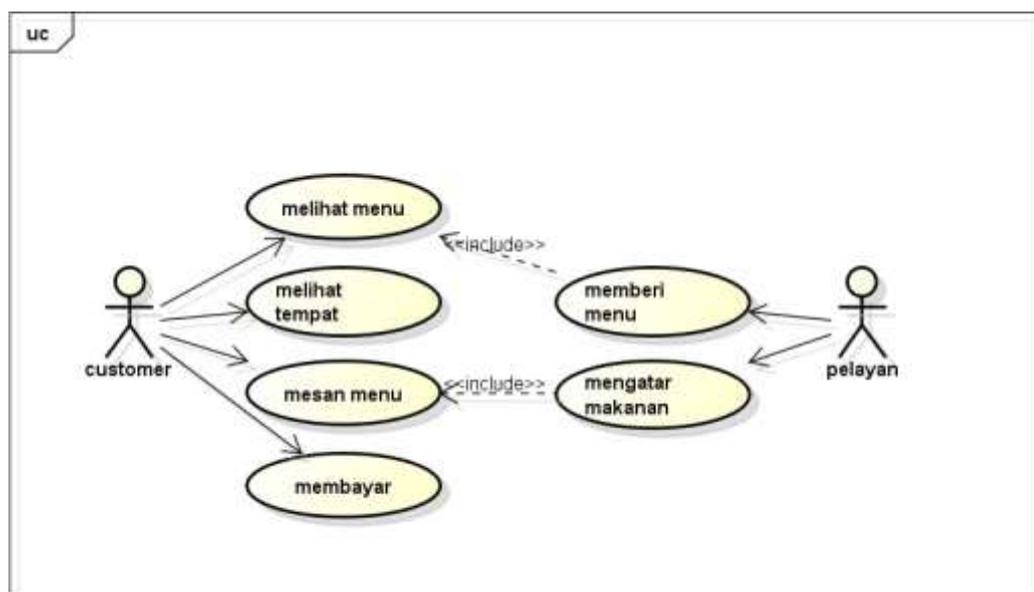
1. *Windows 10*
2. *Visual Studio Code 1.18.1*
3. *MySQL Database Version 5.7.17*
4. *Framework Codeigniter 3*
5. *Astah Community 6.6.4*
6. *WebCodeCamJQuery 2.7.0.*
7. *Qrcodelib Version: 0.0.2*
8. *Manycam Virtual Webcam 4.0*

### 3.4. Rancangan Sistem

Perancangan dilakukan dengan menggunakan *Diagram Unified Modelling Language* yang terdiri dari *use case Diagram*, berikut adalah rancangan sistem berjalan.

#### 3.4.1. Use Case *Diagram* Sistem Berjalan

Sebelumnya, proses pelayanan dilakukan secara manual, di mana *customer* melihat buku menu serta harus memesan dan pelayan mencatat secara langsung, sehingga rentan terhadap kesalahan dan memakan waktu lebih lama. Berdasarkan penggambaran sistem berjalan dapat dilihat terdapat dua bagian aktor seperti *customer* dan pelayan. Penggambaran sistem berjalan dilakukan menggunakan *Diagram use case* bertujuan untuk mempermudah memahami fungsi saat ini berdasarkan masing-masing aktor. Proses pesan pertama dilakukan oleh *customer*, ketika datang langsung menuju menu dan memilih menu. apabila sudah selesai maka akan di pesan lalu pelayan akan memberi pembayaran struck akan keluar. Use case *Diagram* berjalan merupakan pemodelan untuk menggambarkan alur sistem berjalan atau proses bisnis pada proses dapat di lihat pada Gambar 3.1:

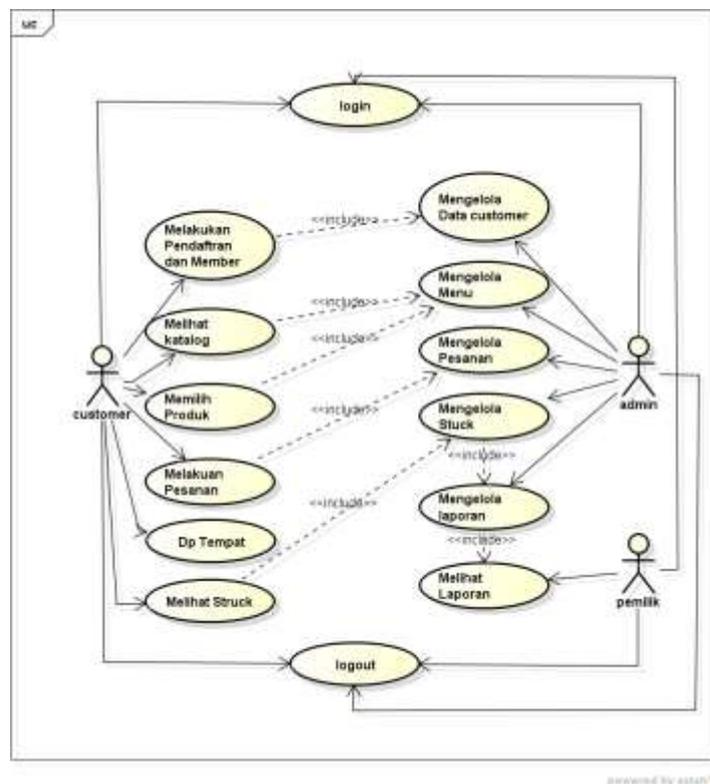


Gambar 3.1 *Use Case Diagram* Sistem Berjalan

### 3.4.2. Use Case Diagram Sistem Usulan

*Use case Diagram* sistem usulan menggambarkan sistem yang akan dibangun menunjukkan antar bagian:

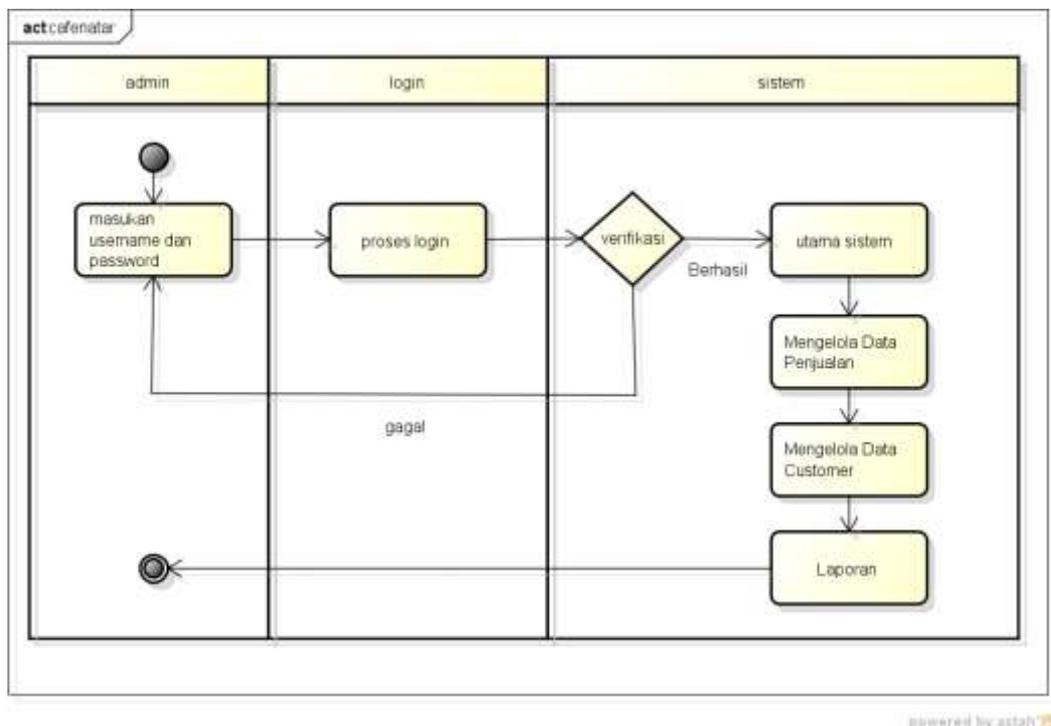
- Customer* dapat melakukan *Login* atau mendaftar akun jika belum memiliki akses, kemudian dapat melihat dan memilih menu, memesan, menerima struk pembayaran secara otomatis, dan keluar dari sistem menggunakan fitur *logout* setelah selesai.
- Pelayan memiliki akses untuk mengelola data *customer*, menu, pesanan, dan struk guna memastikan kesesuaian pesanan dengan pembayaran, serta dapat membuat laporan secara digital melalui fitur mengelola laporan.
- Pemilik memiliki akses untuk melihat laporan yang telah dikelola oleh pelayan. Laporan ini membantu pemilik dalam memantau operasional sistem dan membuat keputusan bisnis secara efektif. *Use case Diagram* sistem yang dibangun dapat di lihat pada Gambar 3.2:



Gambar 3. 2 *Use Case Diagram* Sistem Usulan

### 3.4.3 Activity Diagram Login Dan Sistem

*Activity Diagram* ini menggambarkan proses *Login* pada sistem oleh admin. Proses dimulai ketika admin memasukkan *Username* dan *password*. Kredensial yang dimasukkan kemudian diproses dan diverifikasi oleh sistem. Jika verifikasi berhasil, admin diarahkan ke halaman utama sistem, di mana tersedia fitur seperti pengelolaan data penjualan, pengelolaan data *customer*, dan pembuatan laporan. Jika verifikasi gagal, admin harus mengulangi proses *Login* dari awal. *Diagram* ini membantu memvisualisasikan alur *Login* secara terstruktur dan sistematis. *Activity Diagram Login & Sistem* dapat dilihat pada gambar 3.3:



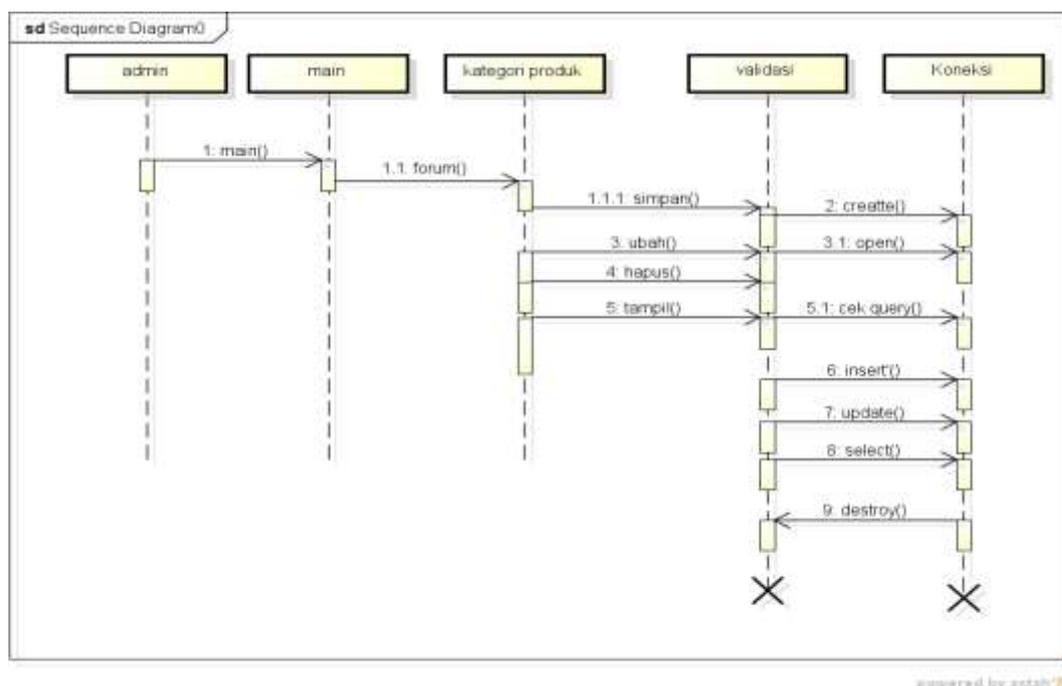
Gambar 3. 3 Activty Diagram & Login Sistem

### 3.4.4 Sequence Diagram

*Sequence Diagram* menggambarkan rangkaian alur proses suatu sistem dengan mengirimkan pesan ke bagian lain dalam alur hidup sistem. Berikut adalah gambaran rancangan sistem menggunakan *Sequence Diagram*:

#### 1. *Sequence Diagram* kategori produk :

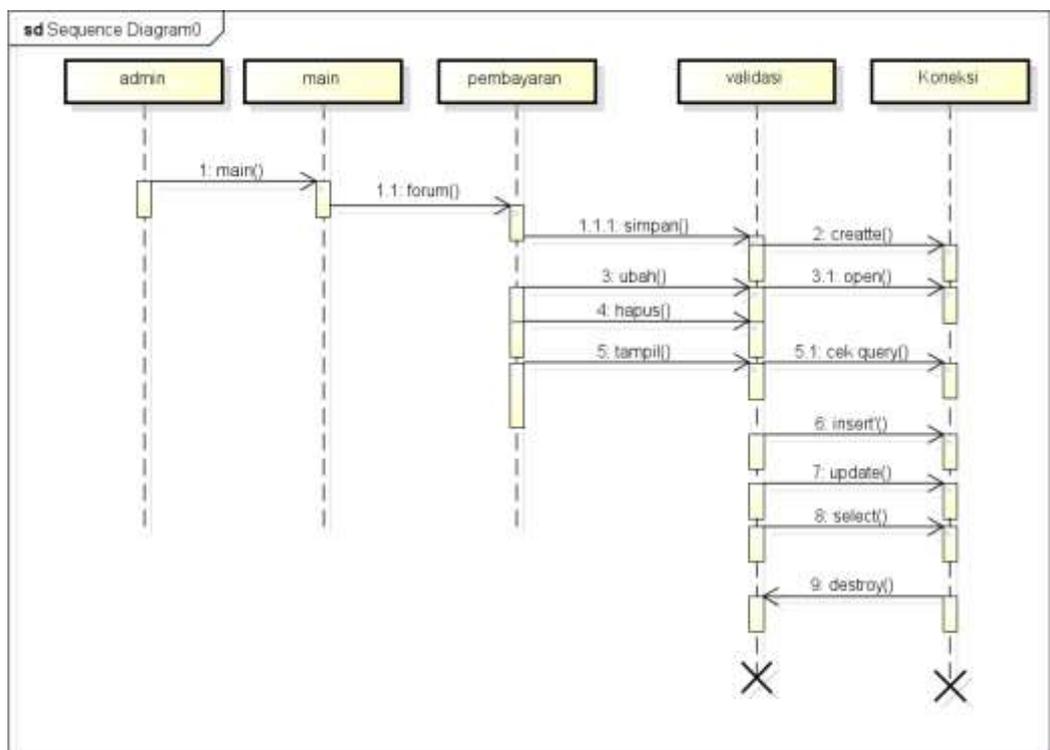
*Sequence Diagram* ini menggambarkan alur pengelolaan kategori produk yang dimulai dari admin memanggil fungsi main pada modul utama, dilanjutkan dengan pemanggilan fungsi forum pada modul kategori produk. Selanjutnya, modul kategori produk mengirimkan perintah ke modul *validasi* untuk menyimpan, mengubah, menghapus, atau menampilkan data. Modul *validasi* kemudian berkomunikasi dengan koneksi *database* untuk melakukan operasi seperti membuat koneksi, membuka koneksi, mengecek *query*, menjalankan perintah insert, update(), dan select(), serta menutup koneksi dengan perintah destroy(). *Diagram* ini menggambarkan interaksi terstruktur antarobjek untuk memastikan proses pengelolaan kategori produk berjalan dengan *validasi* dan koneksi *database* yang sesuai. Berikut adalah *sequence Diagram* kategori produk pada gambar 3.4 :



Gambar 3. 4 *Sequence Diagram* kategori produk

## 2. Sequence Diagram Pembayaran :

*Sequence Diagram* ini menggambarkan alur proses pembayaran yang dimulai dari admin memanggil fungsi main pada modul utama, yang kemudian melanjutkan ke fungsi forum di modul pembayaran. Modul pembayaran berinteraksi dengan modul *validasi* untuk menyimpan, mengubah, menghapus, atau menampilkan data pembayaran. Selanjutnya, modul *validasi* berkomunikasi dengan modul koneksi untuk melakukan proses seperti membuat koneksi (*create*), membuka koneksi (*open*), mengecek *query* (*cek query*), serta melakukan operasi *database* seperti *insert*, *update*, dan *select*. Setelah selesai, koneksi ditutup menggunakan fungsi *destroy*. *Diagram* ini menunjukkan interaksi antarobjek secara terstruktur untuk memastikan proses pembayaran berjalan dengan *validasi* dan pengelolaan *database* yang efisien. Berikut adalah *sequence Diagram* pembayaran pada gambar 3.5 :

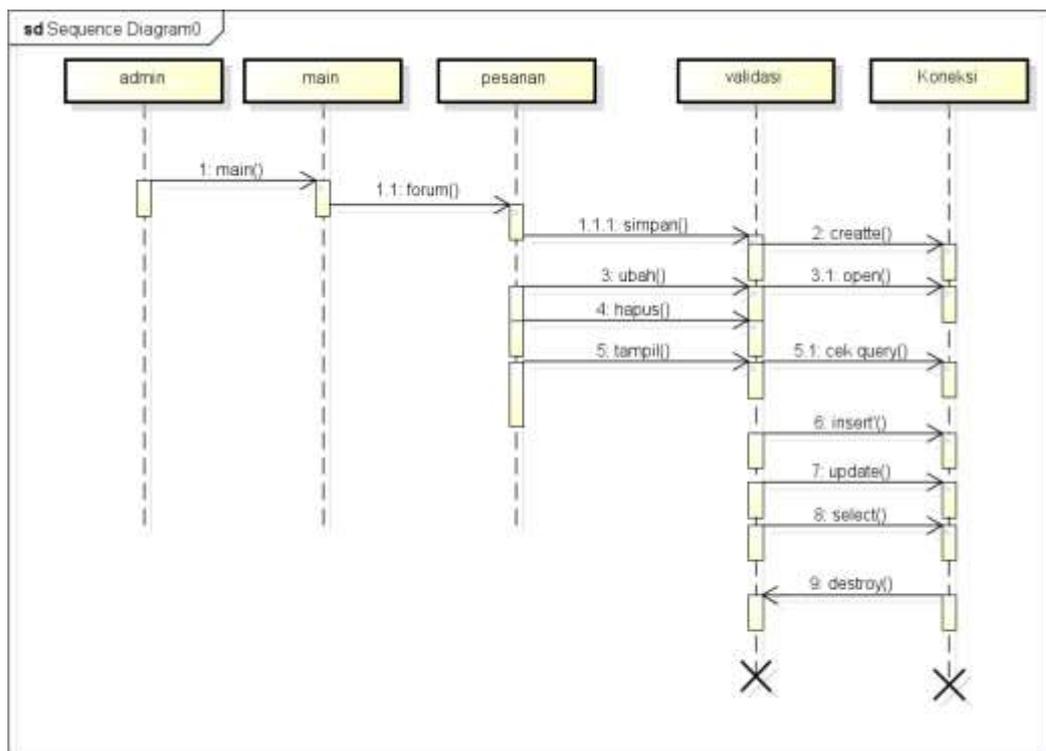


Gambar 3.5 *Sequence Diagram* pembayaran

### 3.4.6 Sequence Diagram pemesanan :

*Sequence Diagram* pemesanan menunjukkan alur interaksi antar objek dalam sebuah sistem pemesanan. *Diagram* ini dimulai dari aktor "admin" yang memanggil fungsi main pada kelas main. Kemudian, fungsi ini memanggil forum di kelas pesanan, diikuti dengan operasi CRUD seperti simpan, ubah, hapus, dan tampil yang mengakses kelas *validasi* untuk memeriksa validitas data. Kelas *validasi* terhubung dengan kelas Koneksi untuk menjalankan operasi *database* seperti *create*, *open*, *cek query*, *insert*, *update*, *select*, dan *destroy*. Semua langkah ini memastikan data dikelola secara terstruktur dan terintegrasi dalam sistem.

Berikut adalah *sequence Diagram* pemesanan pada gambar 3.6 :

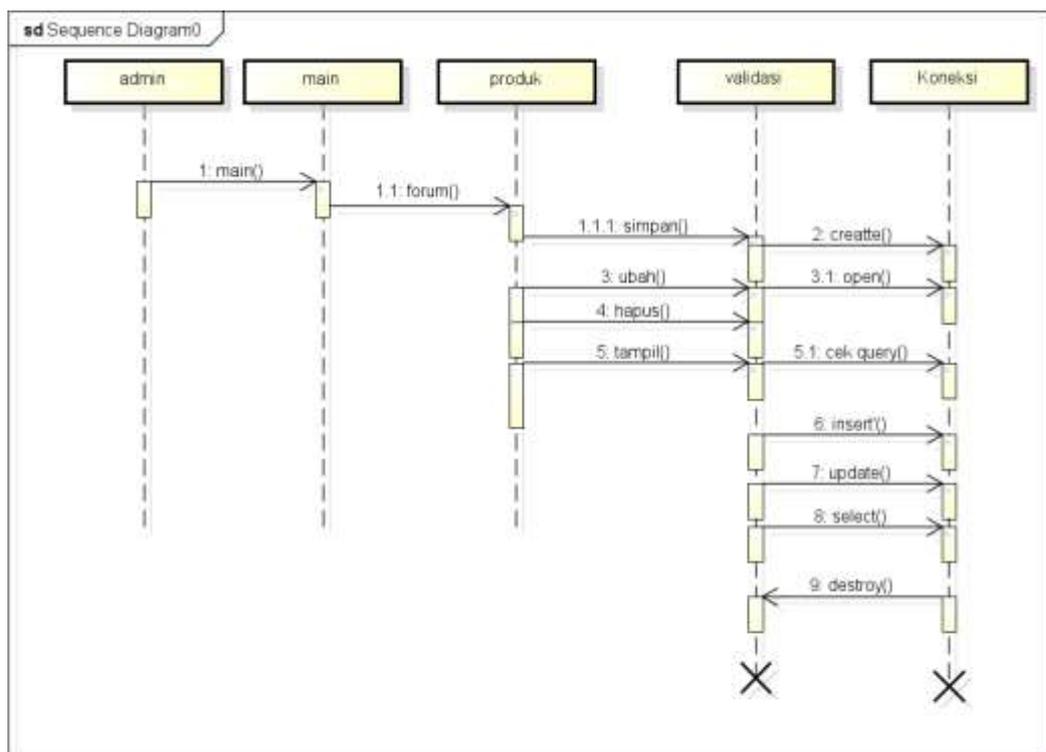


powered by astah

Gambar 3. 6 *Sequence Diagram* pemesanan

### 3.4.7 Sequence Diagram Produk :

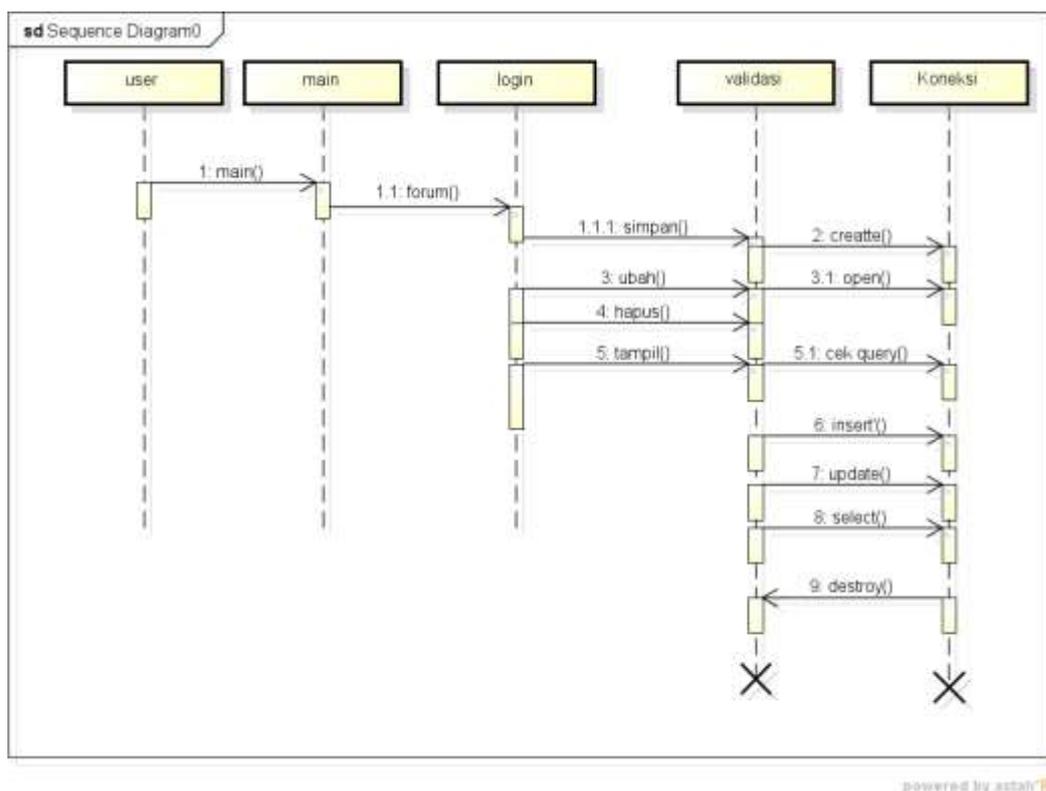
*Sequence Diagram* produk menggambarkan proses interaksi dalam sistem antara aktor "admin" dan beberapa kelas, yaitu main, produk, *validasi*, dan Koneksi. Proses dimulai dengan pemanggilan metode main oleh admin, dilanjutkan ke metode forum di kelas produk. Kelas produk menjalankan fungsi CRUD seperti simpan, ubah, hapus, dan tampil, dengan dukungan *validasi* dari kelas *validasi*. Kelas *validasi* berkomunikasi dengan kelas Koneksi untuk operasi *database*, seperti *create*, *open*, *cek query*, *insert*, *update*, *select*, dan *destroy*. Diagram ini mencerminkan alur kerja sistem yang mengintegrasikan *input* pengguna, *validasi* data, dan pengelolaan basis data.



Gambar 3. 7 *Sequence Diagram* produk

### 3.4.8 Sequence Diagram Login ;

*Sequence Diagram* yang menunjukkan alur interaksi sistem pada proses *Login* dan manajemen data. *Diagram* ini mencakup lima aktor utama: *User*, *main*, *Login*, *validasi*, dan *Koneksi*. *User* memulai dengan memanggil fungsi *main*, yang dilanjutkan ke forum pada *Login*. *Login* memproses fungsi seperti *simpan*, *ubah*, *hapus*, dan *tampil* yang melibatkan *validasi* data melalui fungsi seperti *cek query*. Selanjutnya, modul koneksi mengelola operasi *database*, termasuk *create*, *open*, *insert*, *update*, *select*, dan diakhiri dengan *destroy* untuk menutup koneksi. *Diagram* ini menggambarkan urutan panggilan metode secara terstruktur untuk memastikan data dikelola dengan baik dalam sistem. Berikut *Sequence Diagram Login* pada gambar 3.8:

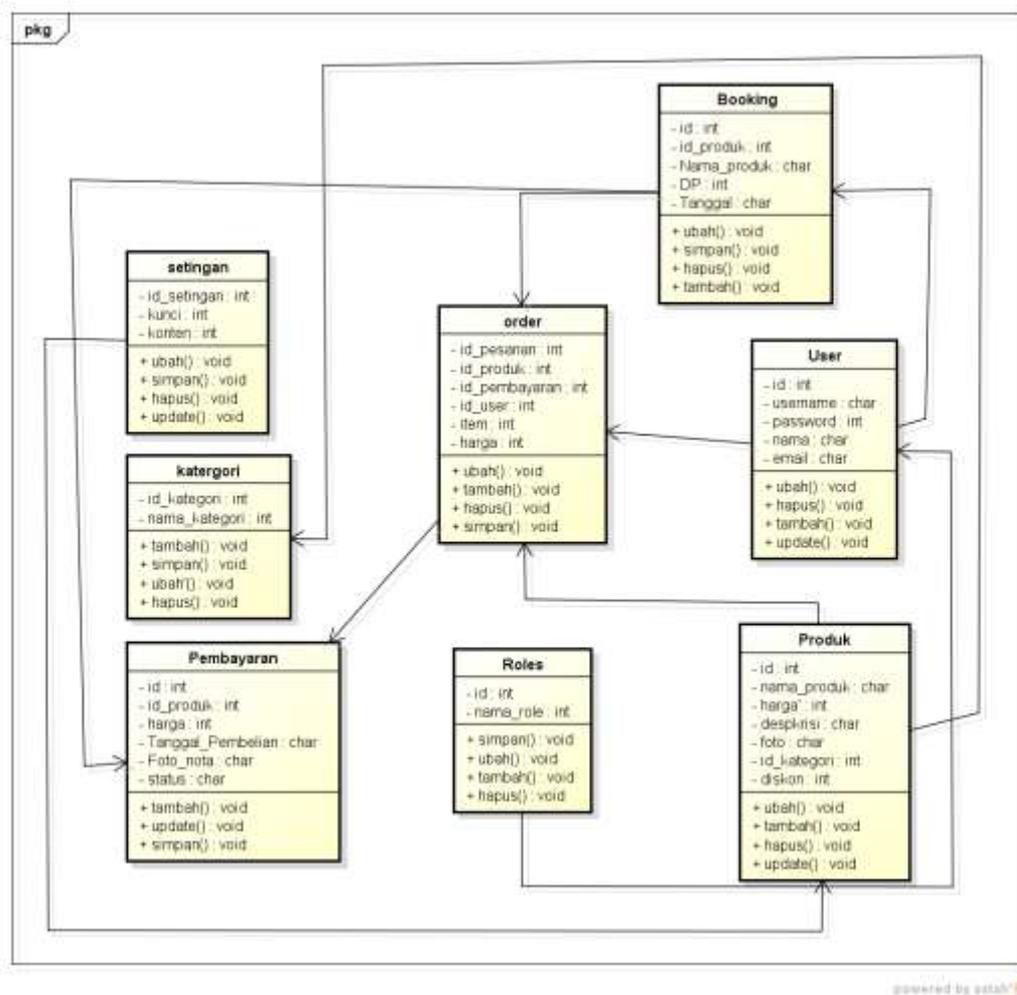


Gambar 3. 8 Sequence Diagram logi

### 3.4.9 Class Diagram

*Diagram class* menjelaskan struktur dari sistem yang saya rancang. Terdapat tujuh kelas utama, yaitu *setingan*, Kategori, Order, Pembayaran, *roles*, *User*, dan Produk. Setiap kelas memiliki atribut dan metode, seperti fungsi tambah, simpan, ubah, dan hapus untuk operasi CRUD. Relasi antar kelas dirancang untuk mendukung alur kerja sistem, misalnya, kelas Order terhubung dengan *User*, Produk, dan Pembayaran untuk memproses transaksi. Produk terkait dengan Kategori untuk pengelompokan, sedangkan *roles* digunakan untuk mengatur peran pengguna. Kelas *setingan* berfungsi untuk menyimpan pengaturan sistem. Dengan struktur ini, sistem dapat berjalan secara terorganisir dan modular.

Berikut *Class Diagram* pada gambar 3.9:



Gambar 3. 9 *Class Diagram*

### 3.4.10 Kamus Data

Kamus data bagian dari pendeskripsian terhadap tabel-tabel yang digunakan pada sistem yang dibangun seperti berikut :

#### 1. Kamus Data Produk

Nama *Database* : cafenatar  
Nama Tabel : kamus data  
produk Primary key : id\_produk  
Foreign key : -

Tabel 3. 1 Kamus Data Produk

Nama <i>Field</i>	<i>Type</i>	<i>Size</i>	<i>Dexcription</i>
id_produk	<i>Int</i>	10	Sebagai id produk
Nama_produk	<i>char</i>	20	Sebagai nama
Harga	<i>int</i>	50	Sebagai harga
deskripsi	<i>char</i>	50	Sebagai deskripsi
foto	<i>char</i>	20	Sebagai foto
Id_kategori	<i>int</i>	20	Sebagai kategori
diskon	<i>int</i>	50	Sebagai diskon

#### 2. Kamus Data Pembayaran

Nama *Database* : cafenatar  
Nama Tabel : pembayaran  
Primary key : id\_produk  
Foreign key : -

Tabel 3. 2 Kamus Data Pembayaran

Nama Field	Type	Size	Dexcription
id_pembayaran	int	50	Sebagai idepembayaran
Id_produk	int	20	Sebagai id_produk
Harga	char	20	Sebagai harga
Tanggal_pembelian	char	15	Sebagai tanggal
Foto_nota	char	20	Sebagai foto nota
status	char	20	Sebagai status

#### 3. Kamus Data *User*

Nama Database

: cafenatar

Nama Tabel : *User*

Primary key : *id\_User*

Foreign key :-

Tabel 3. 3 Kamus Data *User*

Nama Field	Type	Size	Description
<i>id_User</i>	<i>Int</i>	50	Sebagai id <i>User</i>
<i>Username</i>	<i>char</i>	20	Sebagai nama <i>User</i>
<i>Password</i>	<i>Int</i>	50	Sebagai paswoard <i>User</i>
<i>nama</i>	<i>char</i>	50	Sebagai <i>User</i>
<i>email</i>	<i>char</i>	20	Sebagai email <i>User</i>

4. Kamus Data Pesanan

Nama Database : cafenatar

Nama Tabel : Pesanan

Primary key : *id\_Pesanan*

Foreign key :-

Tabel 3. 4 Kamus Data Pesanan

nama field	type	size	dexcription
<i>id_Pesanan</i>	<i>int</i>	15	sebagai id Pesanan
<i>id_produk</i>	<i>int</i>	15	sebagai id_produk
<i>Id_pembayaran</i>	<i>int</i>	20	sebagai Pembayaran
<i>Id_User</i>	<i>int</i>	15	sebagai id_produk
<i>item</i>	<i>int</i>	20	sebagai item
<i>harga</i>	<i>int</i>	50	sebagai harga

5. Kamus Data Kategori produk

Nama Database : cafenatar

Nama Tabel : kategori produk

Primary key : *id\_kategori\_produk*

Foreign key :-

Tabel 3. 5 Kamus Data Kategori Produk

<i>Nama Field</i>	<i>Type</i>	<i>Size</i>	<i>Dexcription</i>
id_kategori_produk	<i>int</i>	15	Sebagai id kategori
Nama_kategori	<i>int</i>	50	Sebagai Nama Kategori

6. Kamus Data *roles*

Nama *Database* : cafenatar

Nama Tabel : *roles*

Primary key : id\_produk

Foreign key :-

Tabel 3. 6 Kamus Data *roles*

<i>Nama Field</i>	<i>Type</i>	<i>Size</i>	<i>Dexcription</i>
Id_produk	<i>int</i>	50	Sebagai id <i>roles</i>
Nama_roles	<i>int</i>	20	Sebagai nama <i>roles</i>

7. Kamus Data *setingan*

Nama *Database* : cafenatar

Nama Tabel : *setingan*

Primary key : id\_ *setingan*

Foreign key :-

Tabel 3. 7 Kamus Data *setingan*

<i>Nama Field</i>	<i>Type</i>	<i>Size</i>	<i>Dexcription</i>
Id_ <i>setingan</i>	<i>int</i>	15	sebagai id <i>setingan</i>
Konten	<i>Int</i>	20	sebagai konten
kunci	<i>int</i>	20	sebagai kunci

## 8. Kamus Data Booking

Nama <i>Database</i>	: cafenatar
Nama Tabel	: Booking
Primary key	: id_Booking
Foreign key	:-

Tabel 3. 8 Kamus Data Booking

Nama <i>Field</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
Id_Booking	<i>int</i>	15	sebagai id Booking
Id_Produk	<i>Int</i>	20	sebagai Id produk
Nama_Produk	<i>Varchar</i>	20	sebagai Nama Produk
DP	<i>int</i>	50	sebagai dp
Tanggal	<i>date</i>	-	sebagai Tanggal

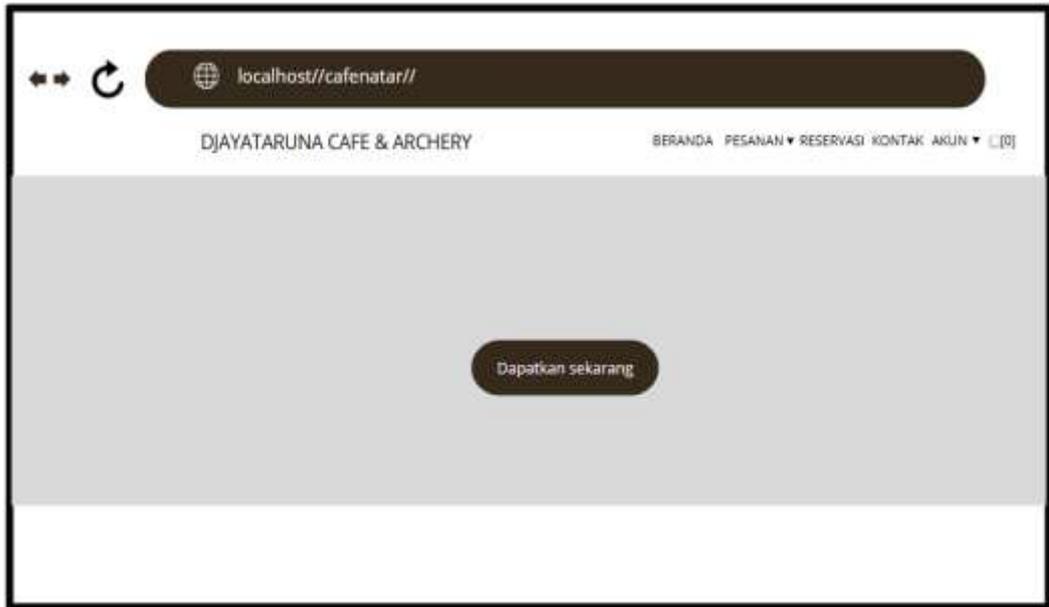
### 3.4.10 Desain Program

Berikut ini adalah rancangan sistem yang dibangun dalam *website* yang akan di implementasikan , berikut ini adalah rancangan sistem yaitu :

#### 1 . Rancangan Tampilan Utama

Rancangan Tampilan utama adalah tampilan antarmuka sebuah situs web lokal dengan nama "DJAYATARUNA CAFE & ARCHERY," yang kemungkinan digunakan untuk mempromosikan sebuah kafe atau tempat rekreasi. Pada bagian atas terdapat menu navigasi utama yang memuat opsi seperti "BERANDA," "PESANAN," "KONTAK," dan "AKUN" untuk memudahkan pengguna menjelajahi halaman. Di tengah halaman terdapat area kosong yang menonjolkan sebuah tombol bertuliskan "Dapatkan sekarang," yang berfungsi sebagai ajakan kepada pengguna untuk melakukan tindakan lebih lanjut. URL di bagian atas *browser* menunjukkan bahwa situs ini sedang diakses melalui server lokal dengan alamat "localhost/cafenatar." Desainnya terlihat sederhana, memberikan kesan fokus pada elemen utama untuk pengalaman pengguna yang efektif.

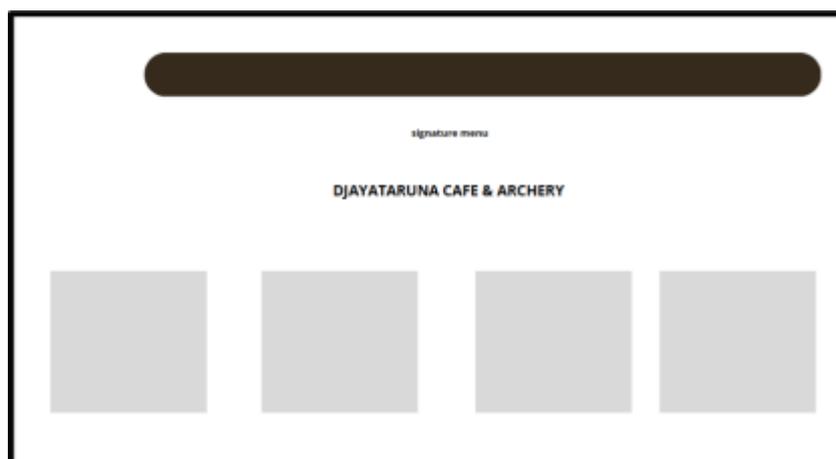
Berikut Tampilan utama pada gambar 3.10:



Gambar 3. 10 Tampilan utama

## 2 . Rancangan Tampilan produk

Rancangan tampilan produk adalah halaman yang berisikan produk berupa makanan , minuman dan area cafe. Di halaman ini di rancang untuk menampilkan dan mempermudah pengguna untuk mengakses *website*. Berikut rancangan tampilan produk pada gambar 3.11



Gambar 3. 11 Tampilan produk

### 3. Rancangan Halaman Pemesanan

Rancangan pemesanan adalah tampilan ketika melakukan pemesanan dimana menampilkan harga, produk dan deskripsi. Berikut tampilan halaman pemesanan pada gambar 3.12



Gambar 3. 12 Halaman pemesanan 1

### 4. Rancangan Pendaftaran Akun

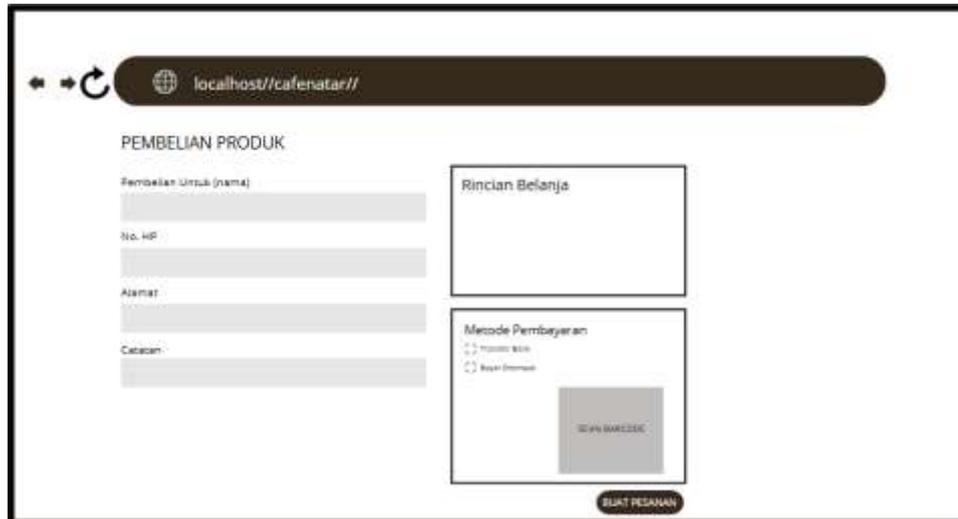
Rancangan pendaftaran akun adalah pelanggan melakukan pendaftaran akun ketika memesan menu. Berikut Rancangan Pendaftaran Akun pada gambar 3.13



Gambar 3. 13 Pendaftaran Akun 1

## 5. Rancangan Pembayaran

Rancangan dibangun supaya pengguna dapat melakukan pembayaran Transfer Bank maupun bayar di tempat. Berikut Rancangan pada gambar 3.14



Gambar 3. 14 Rancangan Pembayaran 1

## 6. Rancangan Dasbor Pelanggan

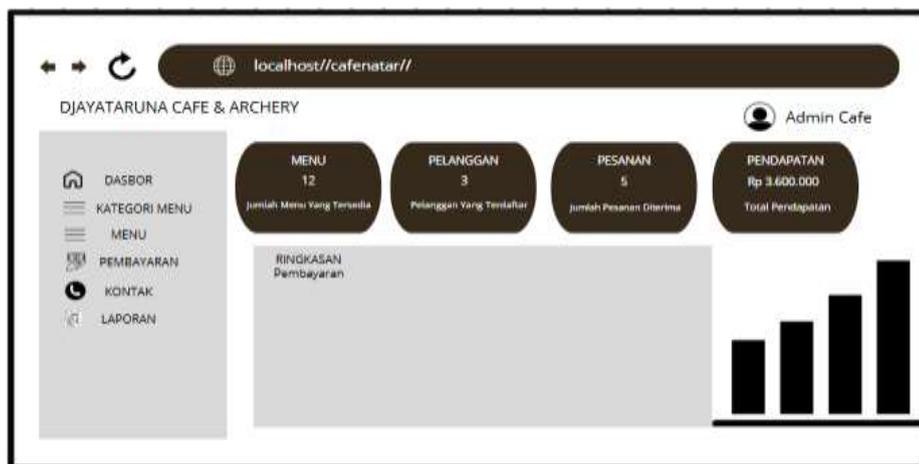


Gambar 3. 15Rancangan Dasbor pelanggan 1

Rancangan dasbor pelanggan dimana berisikan proses order, pembayaran, dan review. Berikut Rancangan dasbor pelanggan dapa dilihat pada gambar 3.15

## 7. Rancangan Antarmuka Dasbor Admin

Rancangan antarmuka untuk halaman dasbor admin Djayataruna Cafe & Archery. Antarmuka halaman admin untuk situs web "DJAYATARUNA CAFE & ARCHERY," yang berisi panel navigasi di sisi kiri dengan menu seperti dasbor, kategori menu, pembayaran, kontak, dan laporan. Berikut adalah rancangan antarmuka dasbor admin pada gambar 3.16



Gambar 3. 16Tampilan Antarmuka Dasbor Ad 1

## 8. Rancangan halaman kategori menu admin

Rancangan halaman kategori terdapat daftar produk yang ditampilkan dalam bentuk kotak pratinjau dengan teks deskripsi di bawahnya yang mencantumkan nama dan harga. Berikut adalah rancangan halaman kategori menu pada gambar 3.17



Gambar 3. 17Rancangan halaman kategori m 1

## 9. Rancangan halaman kategori menu admin

Rancangan halaman kategori terdapat daftar produk yang ditampilkan dalam bentuk kotak pratinjau dengan teks deskripsi di bawahnya yang mencantumkan nama dan harga. Berikut adalah rancangan halaman kategori menu pada gambar 3.17



Gambar 3. 18Rancangan halaman kategori m 2

## 10. Rancangan halaman pesanan pada sistem admin

Rancangan antarmuka pesanan terdapat panel navigasi dengan nama pengguna serta beberapa menu seperti dasbor, pesanan, pembayaran, dan ulasan, serta di bagian utama terdapat tabel dengan judul dasbor yang berisi informasi pesanan termasuk nomor, aksi, ID, tanggal, jumlah pesanan, total pembayaran, dan metode pembayaran..

Berikut adalah Rancangan halaman pesanan pada sistem admin pada gambar 3.18



Gambar 3. 19 Rancangan halaman pesanan pa 1