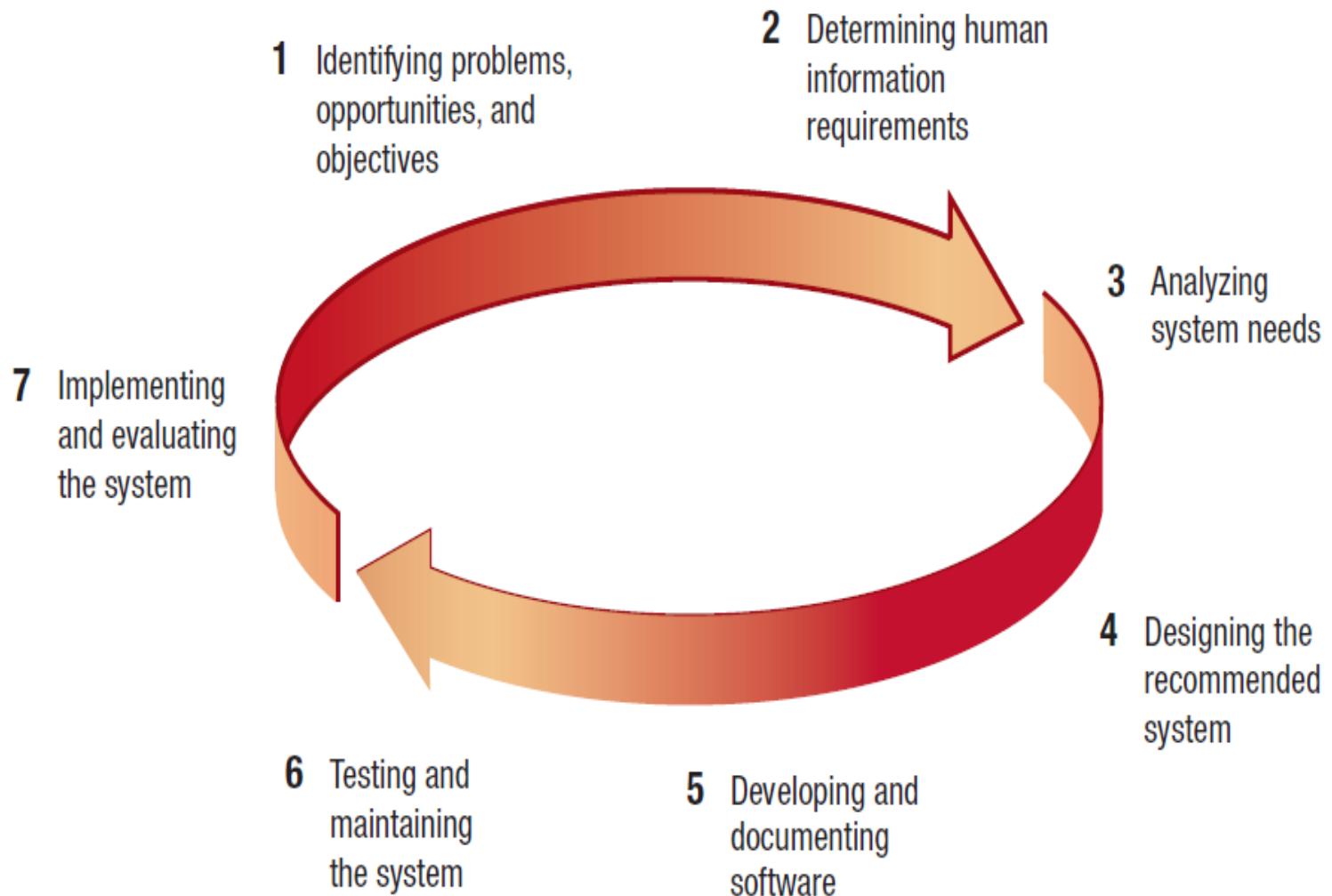


# SDLC – Seven Phases Aplikasi POS UKM Bidang Usaha Kuliner

Pengantar Sistem Informasi

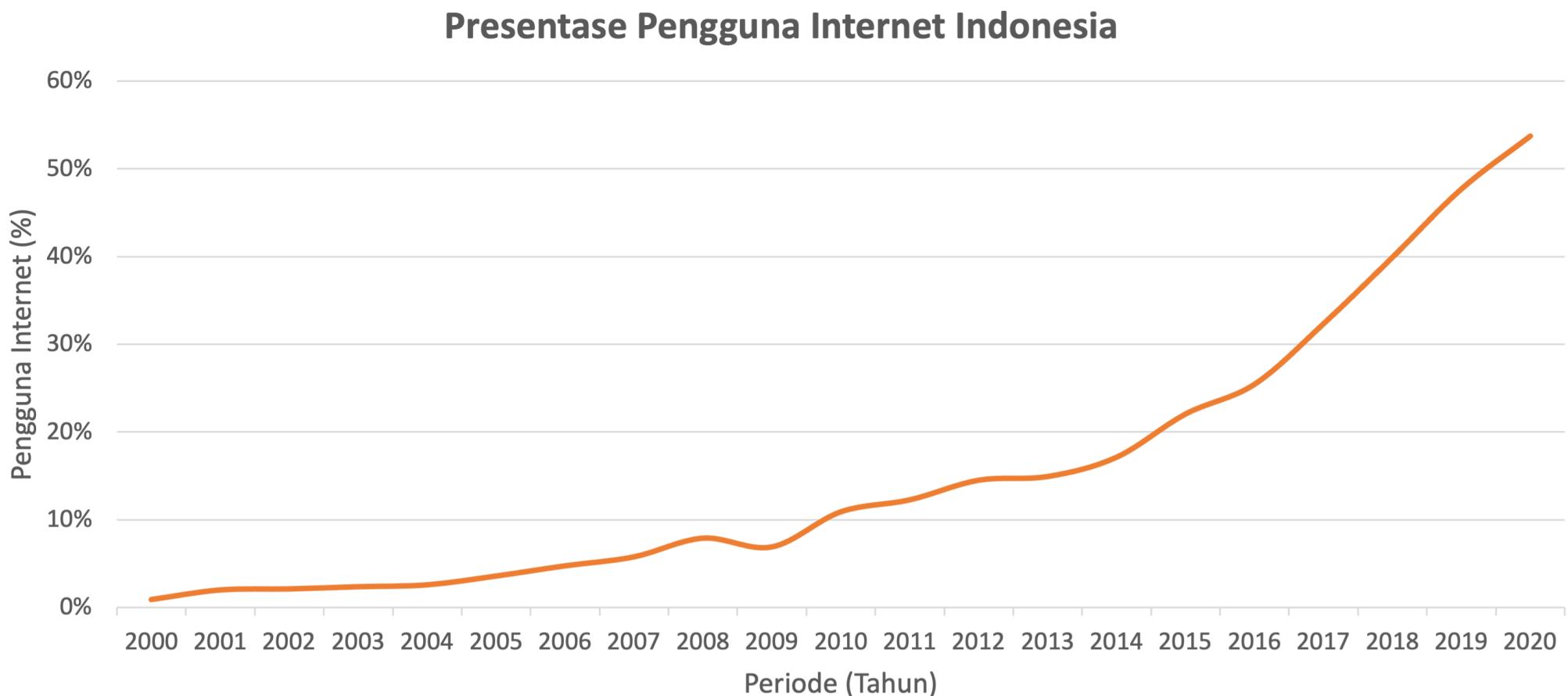
# SDLC – 7 Phase (Kendall)



# 1. Identifikasi Masalah, Lingkup Masalah, Capaian

## Identifikasi masalah

- Pengguna internet di Indonesia terus berkembang dari 1% pada tahun 2000 hingga mencapai 54% pada tahun 2020



# 1. Identifikasi Masalah, Lingkup Masalah, Capaian

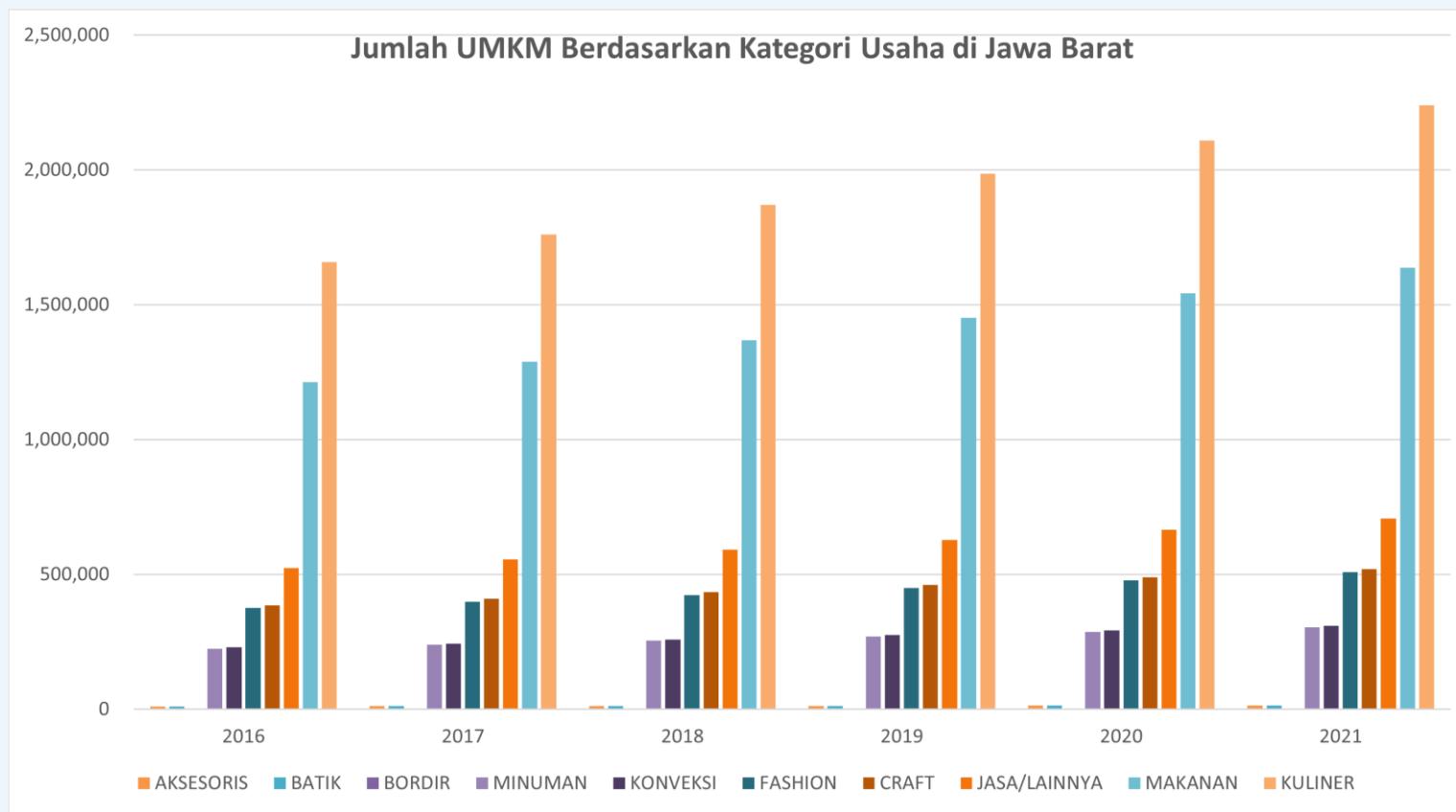
## Identifikasi masalah

- Berkembangnya penggunaan *cloud computing* di Indonesia.
- Implementasi dari *cloud computing* salah satunya ialah pada perangkat lunak POS (Point Of Sales)
- Mesin *cash register* modern memiliki biaya yang relatif lebih mahal bagi pelaku usaha terlebih jika dibandingkan dengan biaya *Cloud-Computing* POS yang dapat diakses melalui berbagai perangkat.

# 1. Identifikasi Masalah, Lingkup Masalah, Capaian

## Identifikasi masalah

- pelaku UMKM memiliki 2 tantangan utama yaitu masalah finansial dan nonfinansial terkait manajemen yang ada di dalam organisasi itu sendiri



# 1. Identifikasi Masalah, Lingkup Masalah, Capaian

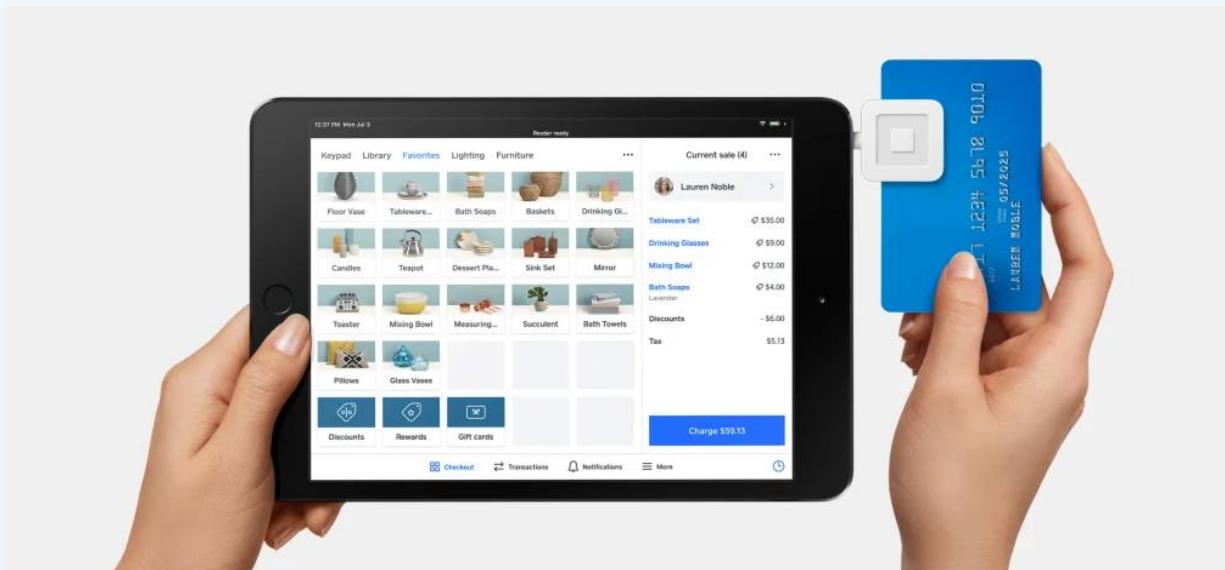
## Identifikasi masalah

- Minimnya wawasan mengenai *quality control* produksi yang dikarenakan oleh keterbatasan wawasan dan pelatihan terkait perkembangan teknologi yang terjadi.
- Minimnya wawasan terkait pemasaran yang disebabkan oleh keterbatasan informasi mengenai pasar, selain itu karena keterbatasan pelaku UMKM untuk memberikan produk atau jasa yang sesuai keinginan pasar.
- Terbatasnya pengetahuan terkait keuangan dan akutansi.
- Secara umum pelaku UMKM tidak mengadakan studi kelayakan, menganalisis perputaran uang, dan berbagai penelitian lainnya yang diperlukan untuk menjalani sebuah aktivitas bisnis.
- Kekurangan informasi bisnis yang dijalankan oleh pelaku UMKM

# 1. Identifikasi Masalah, Lingkup Masalah, Capaian

## Lingkup masalah

- Sistem informasi *Point of Sale* dan *Business Intelligence* dapat membantu para pelaku UMKM untuk memperoleh informasi terkait keinginan pasar, produk yang perlu disediakan untuk pasar, keuangan, analisa perputaran keuangan dan informasi bisnis lainnya



# 1. Identifikasi Masalah, Lingkup Masalah, Capaian

## Capaian

- Studi pengumpulan fakta/data melalui teknik:
  - ✓ wawancara
  - ✓ pengamatan
- Analisis kebutuhan sistem
- Pembangunan perangkat lunak aplikasi *point of sale*
  - ✓ Analisis kebutuhan PL
  - ✓ Perancangan PL
  - ✓ Implementasi PL
- Pengujian oleh pengguna akhir

## 2. Pengumpulan data/fakta

### **Lokasi pengumpulan data/fakta**

- Fin *Coffee* di Jalan Pasir Kaliki: 19 dan 25 November 2021
- Terminale Gelato di Jalan Trunojoyo: 25 November 2021.

### **Pengamatan UKM Jasa Bidang Kuliner di kota Bandung**

- Standar prosedur operasional

### **Wawancara**

- Clement selaku pemilik Fin Coffee
- Kania selaku staf kasir gerai Terminale Gelato

### **Studi dokumen**

- Pencatatan data transaksi penjualan
- Pencatatan informasi transaksi

## 2. Pengumpulan data/fakta

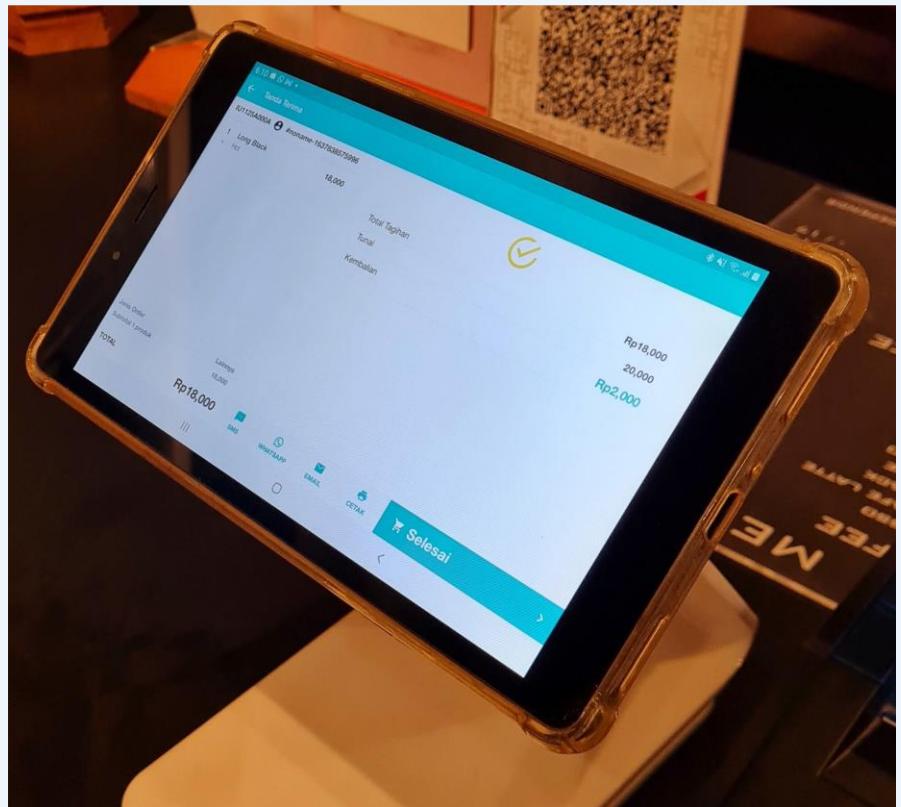
### Hasil pengumpulan fakta/data

- Fin *Coffee* gerai kopi modern yang menyajikan berbagai produk kopi, kue-kue dan makanan ringan seperti kentang goreng.
- Memiliki 2 cabang yaitu di Jalan Pasir Kaliki no 115 dan Jalan Astana Anyar no 13.
- Setiap cabang memiliki tempat pencatatan transaksi penjualan dan inventaris yang berbeda atau terpisah. Pencatatan transaksi sendiri sudah dilakukan dengan menggunakan POS, tetapi sistem yang digunakan pada kedua cabang berbeda.
- Pada cabang pasir kaliki menggunakan merek Majoo sedangkan pada Astana Anyar menggunakan merek Moka POS.

## 2. Pengumpulan data/fakta

### Fin Coffee

- Tampilan *point of sale* memuat informasi pada tanda terima (struk) yang diterima pelanggan
- Informasi memuat nama staf, nama produk, jumlah produk, total tagihan, jenis pembayaran, dan jumlah kembalian.
- Struk diberikan kepada konsumen melalui sms, whatsapp, email atau dicetak pada kertas.



## 2. Pengumpulan data/fakta

### Fin Coffee

- Produk sekitar 150 varian makanan ringan, kue-kue, semua rasa roti bakar, dan kombinasi varian produk kopi.
- pemilik toko dapat mengunduh laporan penjualan keseluruhan pada periode tertentu dalam bentuk *spreadsheet*

1. Tabel kode pesanan dan staf yang mencatat

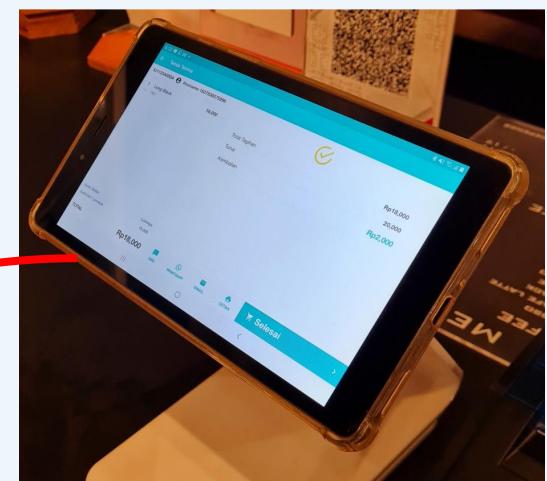
ID pesanan	Staf yang bertanggung jawab
IU1125A000A	#noname-1637838547996

2. Tabel produk yang dipesan dalam transaksi

Jumlah produk	Nama produk	Harga produk
1	Long Black(Hot)	Rp 18.000

3. Tabel total harga transaksi

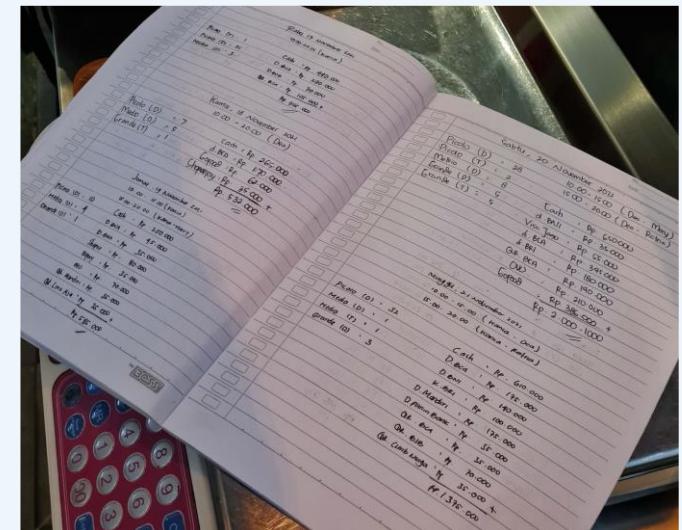
Total Tagihan	Rp 18.000
Tunai	Rp 20.000
Kembalian	Rp 2.000



## 2. Pengumpulan data/fakta

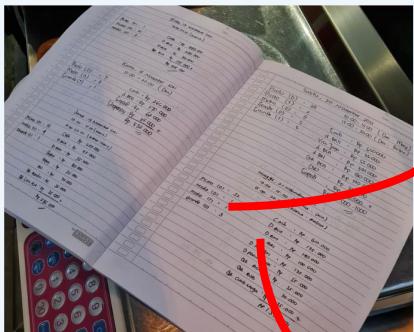
# Terminale Gelato

- UKM yang menjual berbagai produk es krim gelato.
  - Pencatatan transaksi penjualan harian menggunakan sistem POS, catatan kertas dan buku.
  - POS untuk mencatat penjualan harian
  - Catatan kertas atau buku digunakan untuk mencatatkan sisa berat gelato dan mencatat jenis pembayaran yang digunakan serta jumlahnya setiap hari.



## 2. Pengumpulan data/fakta

### Terminale Gelato



1. Tabel staf yang bekerja beserta keterangan waktu

Minggu, 21 November 2021	Staf yang bekerja
10.00 - 15.00	(Kania - Dea)
15.00 - 20.00	(Kania - Ratna)

2. Tabel jenis produk yang terjual beserta tipe penjualannya

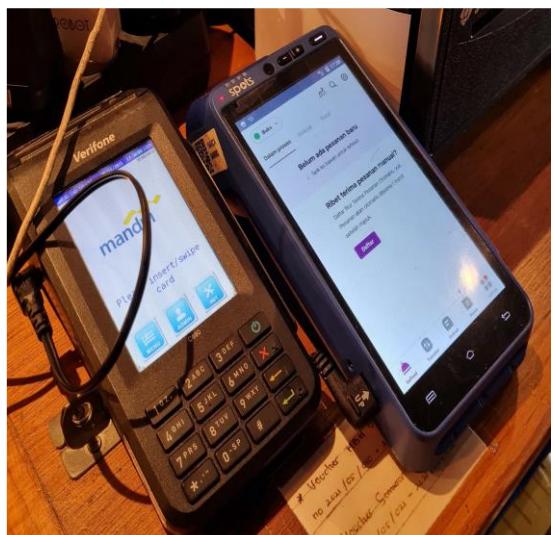
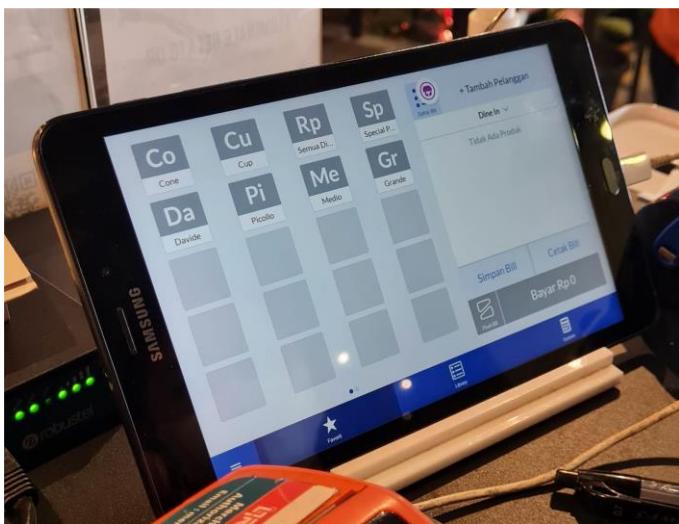
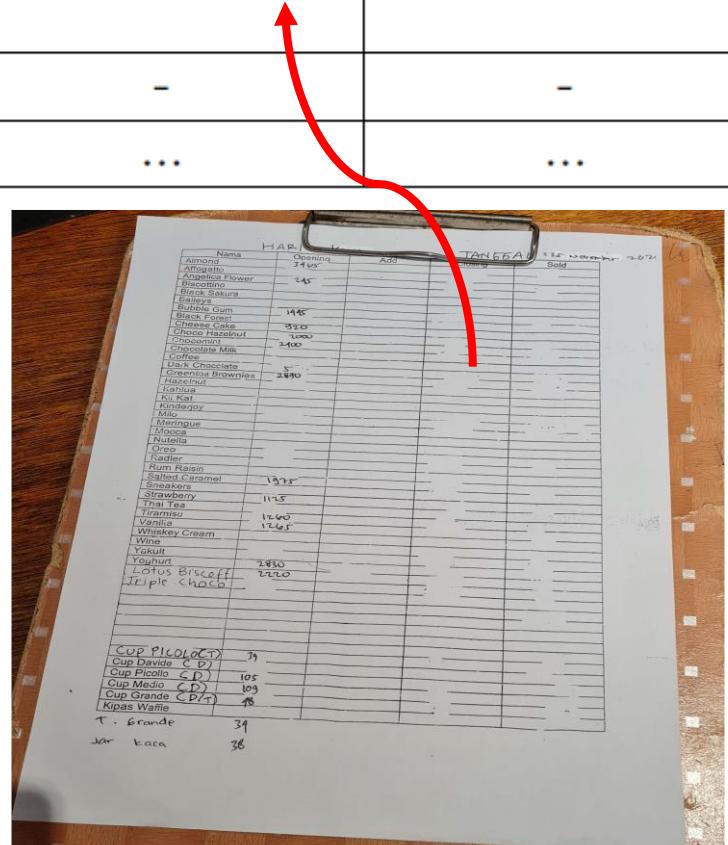
Nama ukuran produk	Jumlah produk yang terjual
Picolo ( <i>Dine-in</i> )	32
Medio ( <i>Dine-in</i> )	1
Medio ( <i>Take away</i> )	1
Grande ( <i>Dine-in</i> )	3

3. Tabel jenis transaksi dan jumlah transaksinya

Tipe transaksi	Jumlah transaksi
Cash	Rp 610.000
Debit BCA	Rp 175.000
Debit BNI	Rp 140.000
Kredit BRI	Rp 100.000
...	...

## 2. Pengumpulan data/fakta

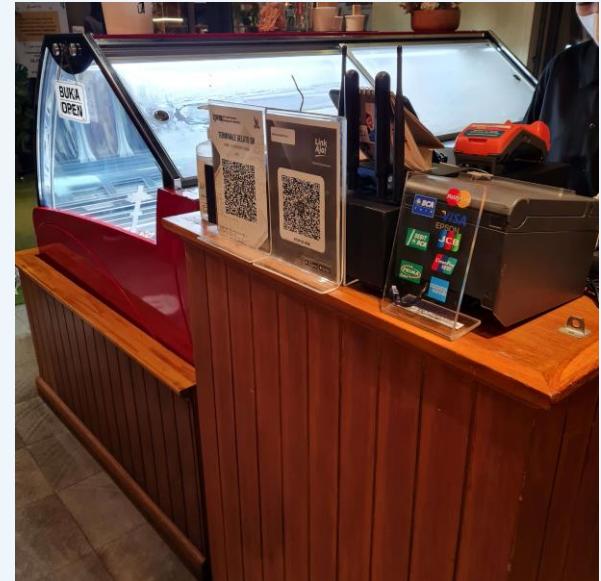
Nama	Opening(gram)	Add	Closing	Sold
Almond	3465	-	-	-
Affogatto	-	-	-	-
Angelica	245	-	-	-
Flower				
Biscottino	-	-	-	-
...	...	...	...	...



## 2. Pengumpulan data/fakta

### Terminale Gelato

- Memiliki *point of sale* dari Gojek (GoFood)
- Proses penjualan: makan di tempat atau pesanan *online*.



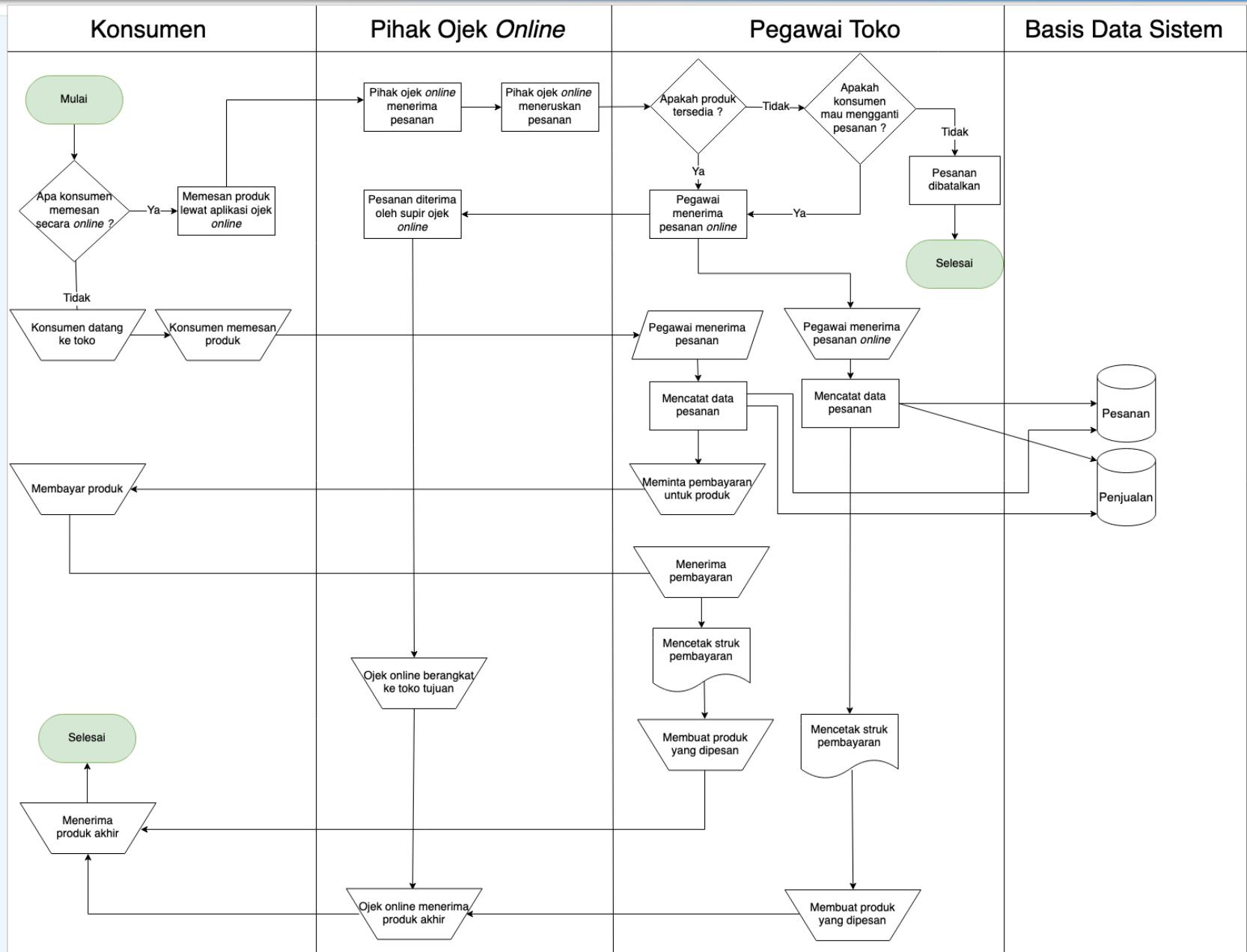
### 3. Analisis Kebutuhan Sistem

#### Kebutuhan Sistem

- Terdapat dua atau lebih cabang toko yang memiliki sistem pencatatan transaksi bisnis yang belum terintegrasi
- Masa kadaluwarsa stok produk yang relative pendek menyebabkan kerugian jika produk tidak sempat habis terjual. (Stok berlebih)
- Terjadi perlambatan penjualan produk. misalnya toko harus tutup pada keadaan tertentu saat seperti PPKM
- Sulit mencatatkan transaksi penjualan pada saat jam sibuk *input* data yang cukup banyak , jumlah sdm terbatas, waktu terbatas.
- Manajer sulit merangkum laporan penjualan dalam (POS belum mendukung fitur berkas *spreadsheet*).
- Keputusan sering dilakukan spontan, karena belum berbasis data transaksi

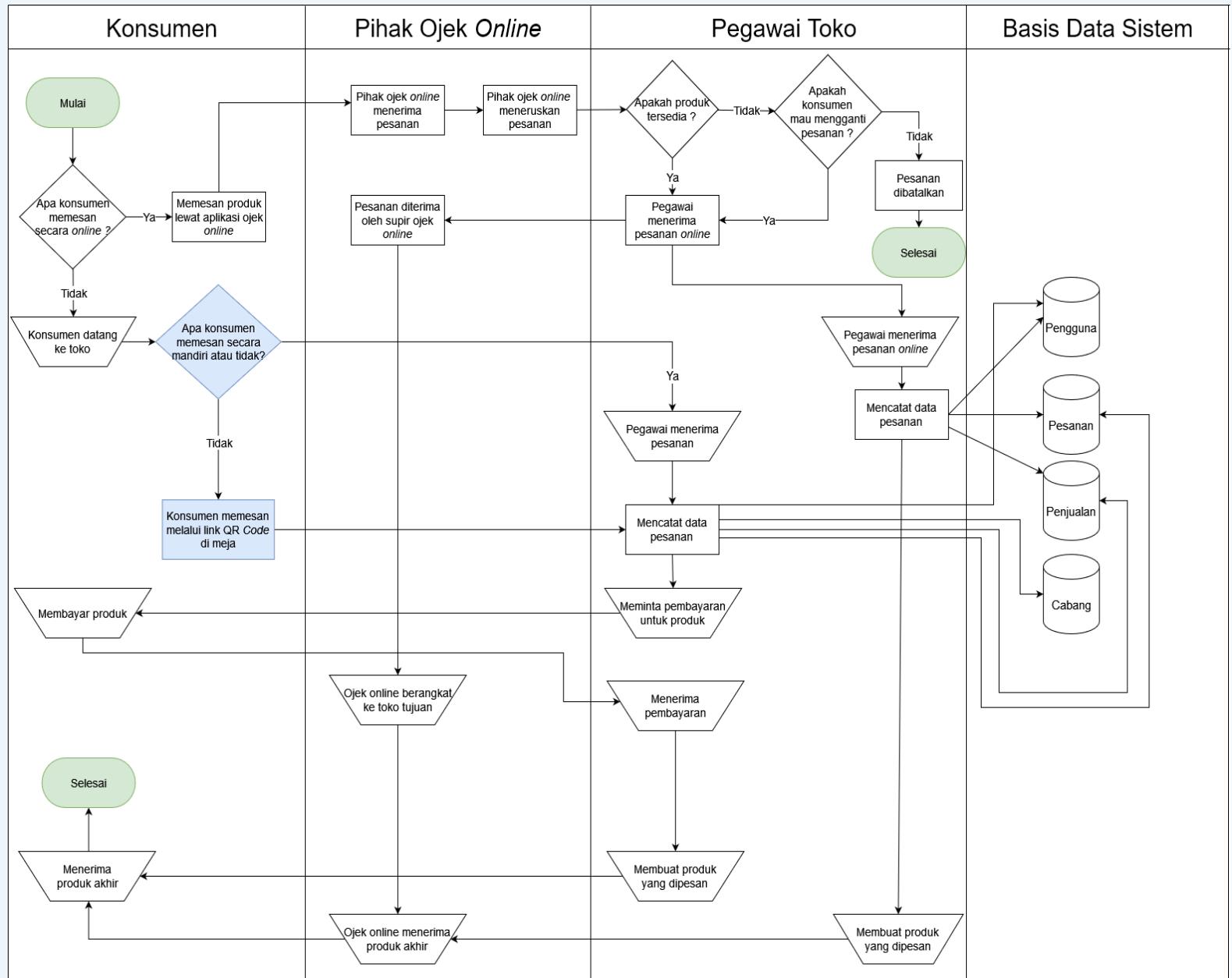
### 3. Analisis Kebutuhan Sistem

## Work Flow Diagram As-is system



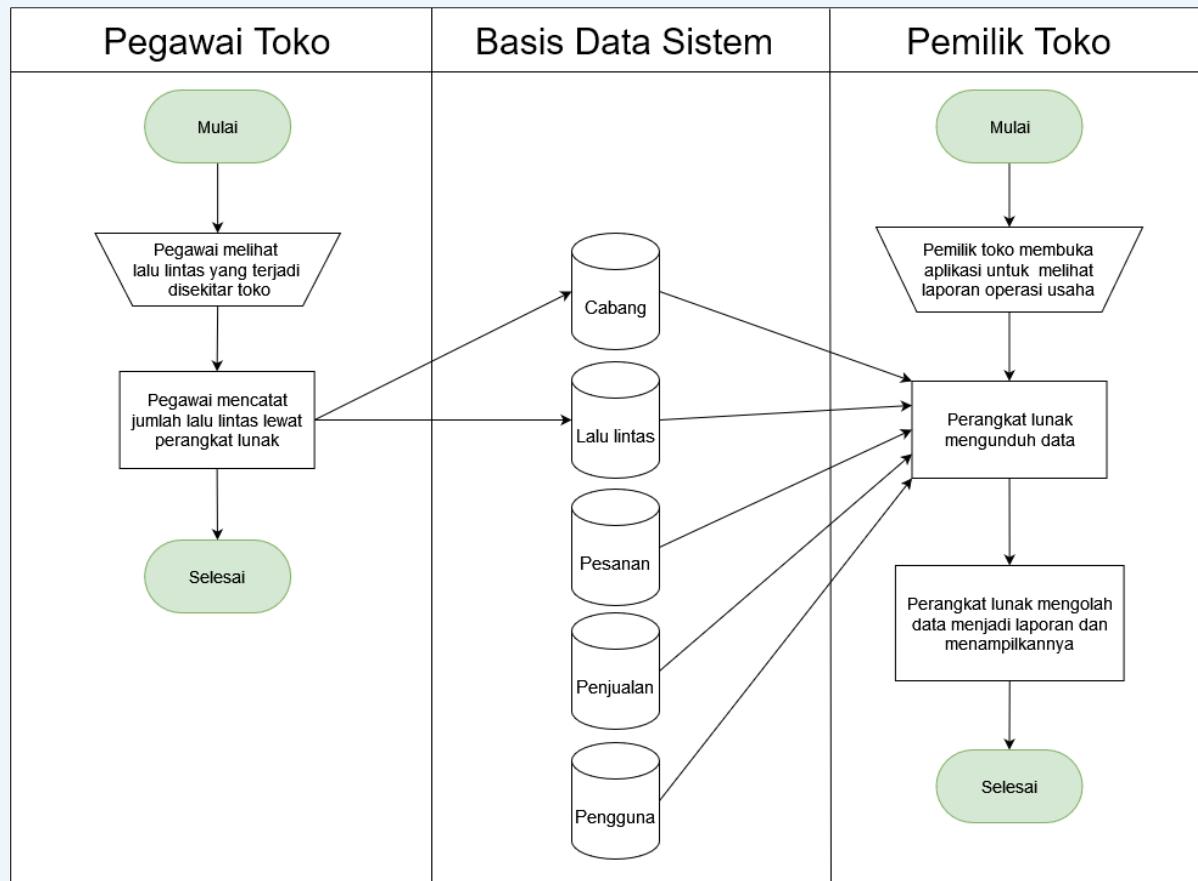
### 3. Analisis Kebutuhan Sistem

## Work Flow Diagram To-be System



### 3. Analisis kebutuhan Sistem Kecerdasan Bisnis

## Sistem Kecerdasan Bisnis



## 4. Perancangan Sistem

### Spesifikasi Pengguna

- Konsumen, pengguna dapat memilih pesanan yang diinginkan secara mandiri.
- Pegawai Toko, pengguna yang berperan untuk mengisi dan mengkonfirmasi data pesanan dan berperan pegawai toko
- Kategori: 3 jenis otorisasi yaitu pemilik toko, manajer, dan staf toko

### Spesifikasi Fungsional - Pemilik Toko

- 1.1. Melakukan *login* ke dalam perangkat lunak.
- 1.2. Mencatat data dari pesanan konsumen.
- 1.3. Mencatat data lalu lintas manusia, mobil dan motor.
- 1.4. Melihat daftar transaksi penjualan harian toko.
- 1.5. Mengelola data dari sebuah produk.

## 4. Perancangan Sistem

### Spesifikasi Fungsional – Pemilik Toko

- 1.6. Mengelola data dari sebuah cabang.
- 1.7. Mengelola data dari sebuah metode pembayaran.
- 1.8. Membuat QR *Code* untuk pemesanan mandiri.
- 1.9. Mengelola data dari antrian pemesanan mandiri.
- 1.10. Mendapatkan data penjualan dalam bentuk berkas *spreadsheet* seperti *file excel*
- 1.11. Mendapatkan laporan informasi relevan terkait bisnis pada masing-masing cabang gerai

## 4. Perancangan Sistem

### Spesifikasi Fungsional - Manajer

- 2.1. Melakukan *login* ke dalam perangkat lunak.
- 2.2. Mencatat data dari pesanan konsumen.
- 2.3. Mencatat data lalu lintas manusia, mobil dan motor.
- 2.4. Melihat daftar transaksi penjualan harian toko.
- 2.5. Mengelola data dari sebuah produk.
- 2.6. Mengelola data dari sebuah cabang.
- 2.7. Mengelola data dari sebuah metode pembayaran.
- 2.8. Membuat QR *Code* untuk pemesanan mandiri.
- 2.9. Mengelola data dari antrian pemesanan mandiri.

## 4. Perancangan Sistem

### Spesifikasi Fungsional - Staf Toko

- 3.1. Melakukan *login* ke dalam perangkat lunak.
- 3.2. Mencatat data dari pesanan konsumen.
- 3.3. Mencatat data lalu lintas manusia, mobil dan motor.
- 3.4. Melihat daftar transaksi penjualan harian toko.
- 3.5. Mengelola data dari antrian pemesanan mandiri.
- 3.6. Mengelola data status ketersedian dari sebuah produk.

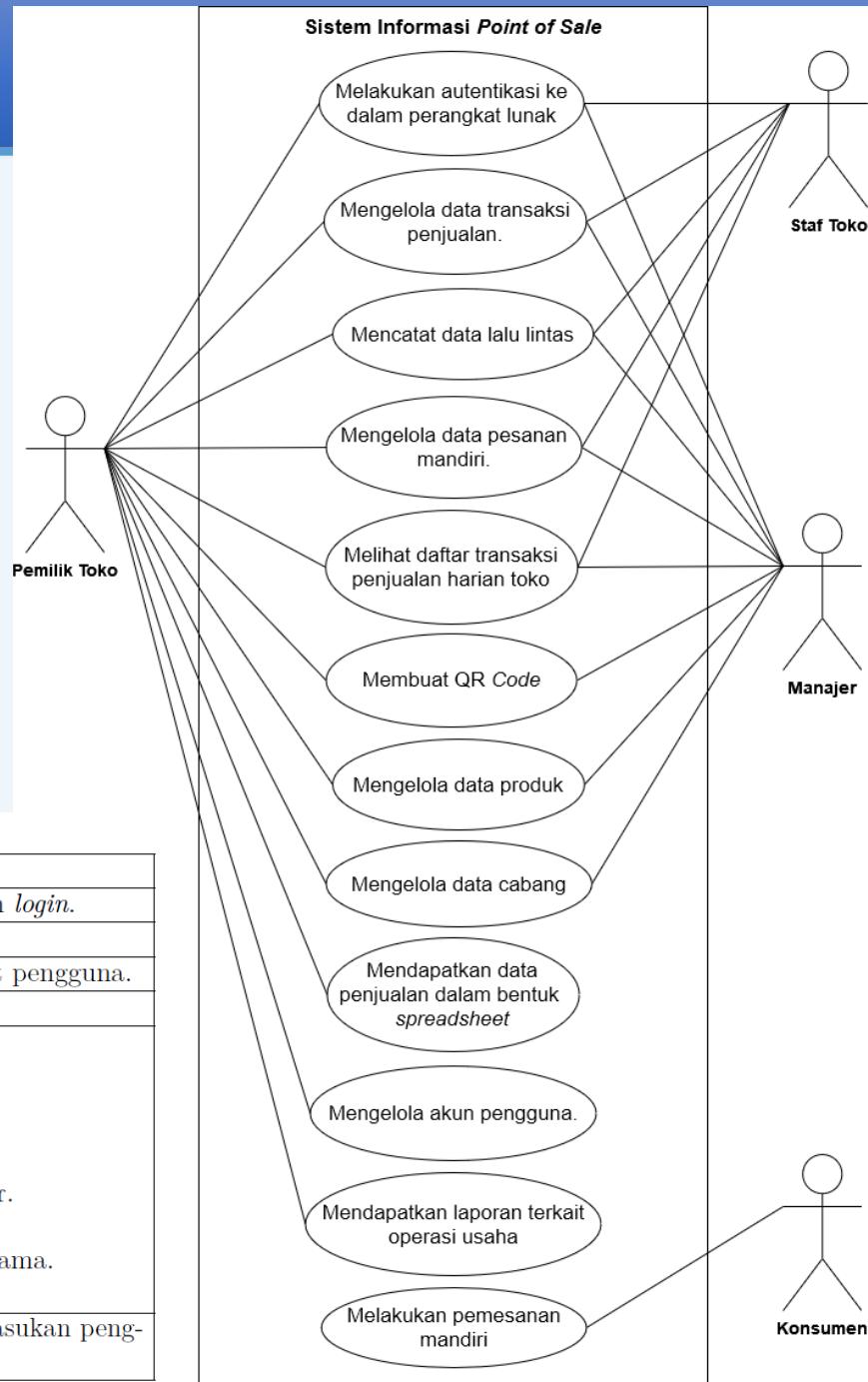
### Spesifikasi Fungsional - Konsumen

(khusus makan di tempat)

- 4.1. Melakukan pemesanan mandiri menggunakan perangkat lunak *point of sale*

# 5. Pembangunan PL

## Analisis kebutuhan PL



Tabel 3.1: Tabel skenario melakukan *login* pada *website*.

<b>Skenario</b>	Melakukan <i>login</i> pada <i>website</i> .
<b>Deskripsi</b>	Pada skenario aktor melakukan autentikasi pada halaman <i>login</i> .
<b>Aktor</b>	Pemilik Toko
<b>Pra-Kondisi</b>	Pengguna sudah membuka halaman <i>login</i> pada perangkat pengguna.
<b>Pos-Kondisi</b>	Aktor terautentikasi dan masuk ke halaman utama.
<b>Skenario normal</b>	<ol style="list-style-type: none"> <li>1. Aktor masuk ke halaman <i>login</i>.</li> <li>2. Perangkat lunak menampilkan halaman <i>login</i>.</li> <li>3. Aktor memasukan <i>email</i> dan kata sandi.</li> <li>4. Aktor menekan tombol <i>login</i>.</li> <li>5. Perangkat lunak menerima masukan data dari aktor.</li> <li>6. Perangkat lunak mengautentikasi data tersebut.</li> <li>7. Perangkat lunak mengarahkan aktor ke halaman utama.</li> </ol>
<b>Eksepsi</b>	Perangkat lunak mengembalikan pesan <i>error</i> pada saat masukan pengguna tidak berhasil diautentikasi.

## 5. Pembangunan PL

### Analisis Kebutuhan Basis Data (Entity Relationship Diagram)

1. *Users*: Entitas ini berisi data-data pengguna yang berada dalam kegiatan operasi usaha yang memiliki atribut sebagai berikut.

- 1.1. *User\_id*, sebuah *primary key* untuk entitas *users* yang bertujuan untuk memberikan identitas unik untuk pihak pengguna yang ada.
- 1.2. *Name*, atribut berisi nama pengguna.
- 1.3. *Password*, atribut berisi kata sandi pengguna.
- 1.4. *Access Privilege*, atribut berisi informasi mengenai hak akses setiap pengguna.
- 1.5. *Status*, atribut yang berisi informasi mengenai status akun aktif atau tidak aktif.
- 1.6. *Branch\_id*, sebuah *foreign key* dari entitas *Branches* yang bertujuan untuk menginformasikan di cabang mana setiap pengguna beroperasi.

## 5. Pembangunan PL

### Analisis Kebutuhan Basis Data

2. *Orders*: Entitas ini berisi data-data dari pesanan dibuat baik oleh konsumen mandiri atau dicatat oleh pihak staf toko.

Atribut-atribut dari entitas ini adalah sebagai berikut,

- 2.1. *Order\_id*, sebuah *primary key* untuk entitas pesanan bertujuan untuk memberikan identitas unik untuk setiap pesanan yang ada.
- 2.2. *User\_id*, sebuah *foreign key* dari entitas *Users*.
- 2.3. *Order\_date\_at*, atribut berisi tanggal pesanan dicatat.
- 2.4. *Order\_time\_at*, atribut berisi jam dan waktu pesanan dicatat.
- 2.5. *Payment\_method*, atribut berisi metode pembayaran yang dipilih pada saat pencatatan dilakukan.
- 2.6. *Transaction\_value*, atribut berisi total nilai transaksi dari pesanan yang diterima.

## 5. Pembangunan PL

### Analisis Kebutuhan Basis Data

- 2.7. *Order\_type*, atribut ini berisi tipe dari transaksi penjualan yang dicatat.
- 2.8. *Branch\_id*, sebuah *foreign key* dari entitas *Branches*.
- 2.9. *User\_id*, sebuah *foreign key* dari entitas *Users*.

3. *Sales*: Entitas ini berisi data-data setiap penjualan yang terjadi dan data dicatat per produk yang dipesan.

Atribut-atribut dari entitas ini adalah sebagai berikut,

- 3.1. *Sale\_id*, sebuah *primary key* untuk entitas penjualan bertujuan untuk memberikan identitas unik untuk setiap penjualan yang terjadi.
- 3.2. *Product\_id*, sebuah *foreign key* dari entitas *products*.
- 3.3. *Total\_price*, sebuah atribut yang dihasilkan dari perhitungan komputasi antara harga per produk dan jumlah produk yang dibeli.

## 5. Pembangunan PL

### Analisis Kebutuhan Basis Data

- 3.4. *Amount*, sebuah atribut yang mencatat jumlah produk yang dibeli.
  - 3.5. *Sale\_status*, sebuah atribut yang mencatat status akhir dari penjualan yang dilakukan.
4. *Products*: Entitas ini berisi data-data mengenai produk yang dijual pada setiap cabang yang terdaftar.

Atribut-atribut dari entitas ini adalah sebagai berikut,

- 4.1. *Product\_id*, sebuah *primary key* untuk entitas produk bertujuan untuk memberikan identitas unik untuk setiap produk yang ada.
- 4.2. *Product\_name*, atribut untuk mencatat nama-nama produk yang tersedia.
- 4.3. *Price*, atribut untuk mencatat harga setiap produk.

## 5. Pembangunan PL

### Analisis Kebutuhan Basis Data

- 4.4. Status, atribut yang berisi informasi mengenai status ketersediaan produk.
- 4.5. Branch\_id, sebuah *foreign key* dari entitas *branches*.

5. *Branches*: Entitas ini berisi data-data mengenai informasi cabang yang ada, mencakup nama dan lokasi cabang tersebut.

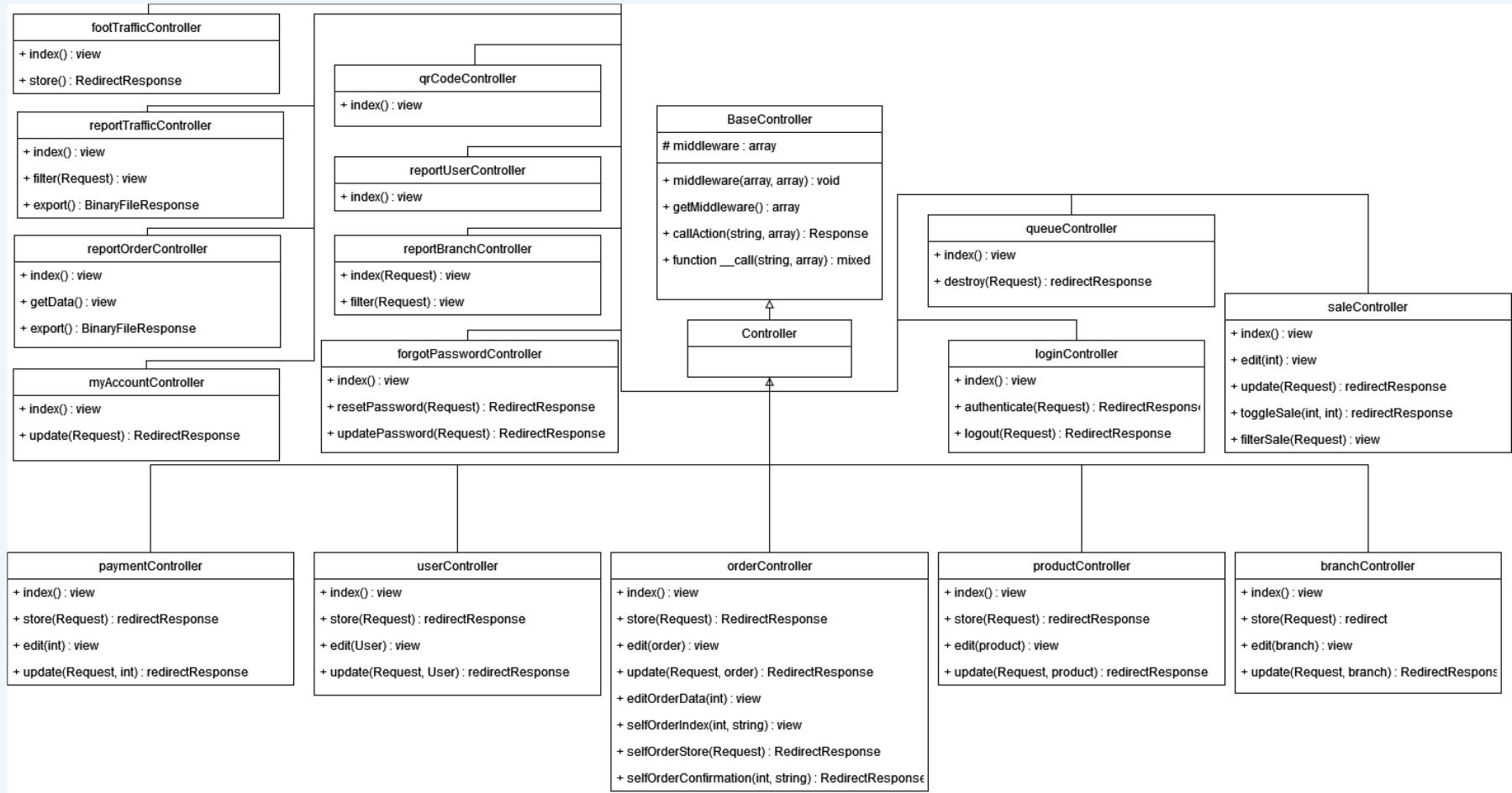
Atribut-atribut dari entitas ini adalah sebagai berikut,

- 5.1. Branch\_id, sebuah *primary key* untuk entitas cabang bertujuan untuk memberikan identitas unik untuk setiap cabang yang ada.
- 5.2. Branch\_name, atribut untuk mencatat nama setiap cabang.
- 5.3. *Location*, atribut untuk mencatat lokasi setiap cabang.
- 5.4. *Active*, atribut berisi mengenai status aktif atau tidaknya sebuah cabang.
- 5.5. *Size*, atribut ini berisi luas dari sebuah toko cabang dalam satuan meter persegi.

# 5. Pembangunan PL

## Perancangan PL

### Diagram Kelas Controller



# 5. Pembangunan PL

## Perancangan PL

### Diagram Kelas Model

branch	order	product
# table : string  # primaryKey : string  + timestamps : boolean  # fillable : array  # attributes : array	# table : string  # primaryKey : string  + timestamps : boolean	# table : string  # primaryKey : string  + timestamps : boolean
+ createOrder() : collection  + haveUser() : collection  + usePayment() : collection	+ sales() : collection  + createdBranch() : collection  + registerUser() : collection	+ containSale() : collection  + manageUser() : collection
payment	orders_sales	queue_list
# table : string  # primaryKey : string  + timestamps : boolean  + useBranch() : collection	# table : string  + timestamps : boolean  # fillable : array	# table : string  # primaryKey : string  + timestamps : boolean
branch_payment	sale	User
# table : string  + timestamps : boolean  # fillable : array	# table : string  # primaryKey : string  + timestamps : boolean	# table : string  # primaryKey : string  + timestamps : boolean  # fillable : array  # hidden : array  # casts : array
+ orders() : collection  + containProduct() : collection	+ orders() : collection  + containProduct() : collection	+ registerOrder() : collection  + haveBranch() : collection  + manageProduct() : collection
traffic		
# table : string  # primaryKey : string  # attributes : array  + timestamps : boolean		
+ branch() : collection		

# 5. Pembangunan PL

## Perancangan PL

### Diagram Kelas Component

addBranch	addProduct
+ __construct() : void + render() : view	+ __construct() : void + render() : view
addPayment	addStaff
+ __construct() : void + render() : view	+ __construct() : void + render() : view

## 5. Pembangunan PL

### Perancangan Basis Data

#### 2. Tabel *branch\_payment*

Nama Kolom	Tipe Data	PK / FK	Nullable
branch_id	bigint(20) unsigned	FK	False
payment_id	bigint(20) unsigned	FK	False

#### 3. Tabel *orders*

Nama Kolom	Tipe Data	PK / FK	Nullable
order_id	bigint(20) unsigned	PK	False
payment_method	varchar(255)	-	False
order_date_at	date	-	False
order_time_at	time	-	False
transaction_value	bigint(20)	-	False
order_type	varchar(8)	-	False
branch_id	bigint(20) unsigned	FK	False
user_id	bigint(20) unsigned	FK	False

# 5. Pembangunan PL

## Perancangan Antar Muka



### Order



Waffle



Cake



Coffee



IC

Ice Tea

#### Daftar Pesanan

Waffle (2x)	Rp 30.000
Coffee	Rp 10.000

Metode Pembayaran

BAYAR (Rp 44.000)

## 5. Pembangunan PL

### Perancangan Antar Muka

#### Product

Cabang Sukajadi

Tambah

Nama	Harga	Cabang	Active
Waffle	Rp 15.000	Bandung	true
Coffee	Rp 10.000	Jakarta	false
Ice Tea	Rp 8.000	Medan	true



## 5. Pembangunan PL

### Perancangan Antar Muka

#### Staf

Tambah

Username	Nama	Password	Hak Akses	Cabang
admin	John	*****	Semua Akses	Sukajadi
cashier	Budi	*****	Kasir	Pasteur

## 5. Pembangunan PL

### Perancangan Antar Muka

#### Sales

Order ID	Revenue	Date	Status	Cabang
1	Rp 70.000	10 Jan'20	Success	Sukajadi
2	Rp 60.000	5 Jan'20	Fail	Pasteur

## 5. Pembangunan PL

### Perancangan Antar Muka

foot traffic

Person	1	^	v
Car	1	^	v
Motorcycle	1	^	v
Submit			



## 5. Pembangunan PL

### Perancangan Antar Muka



#### QR Code

**Path**

**Branch**

Bandung

**Table**

**Size**

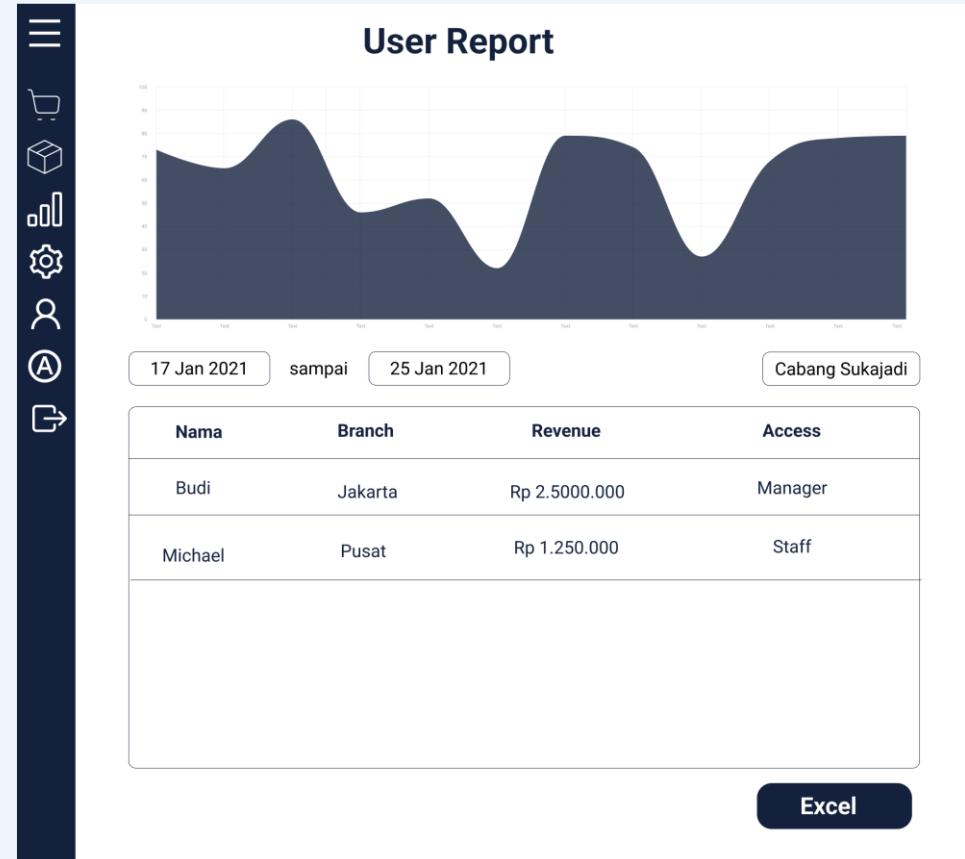
500 px

Generate



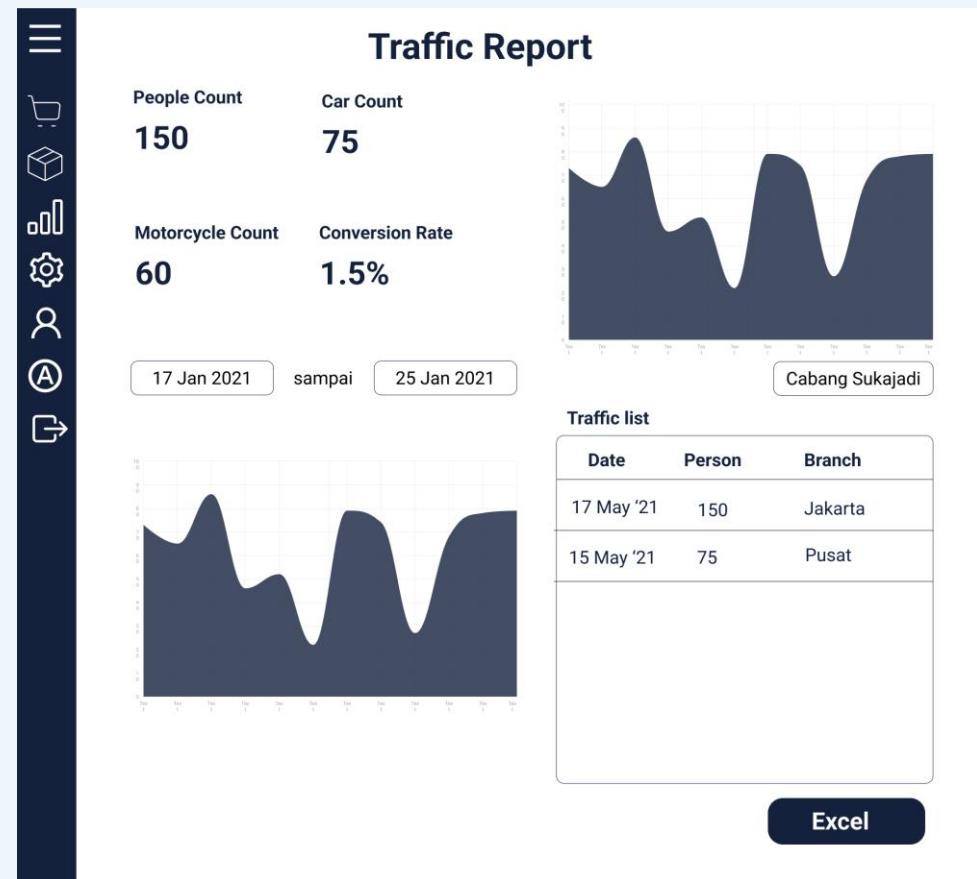
# 5. Pembangunan PL

## Perancangan Antar Muka



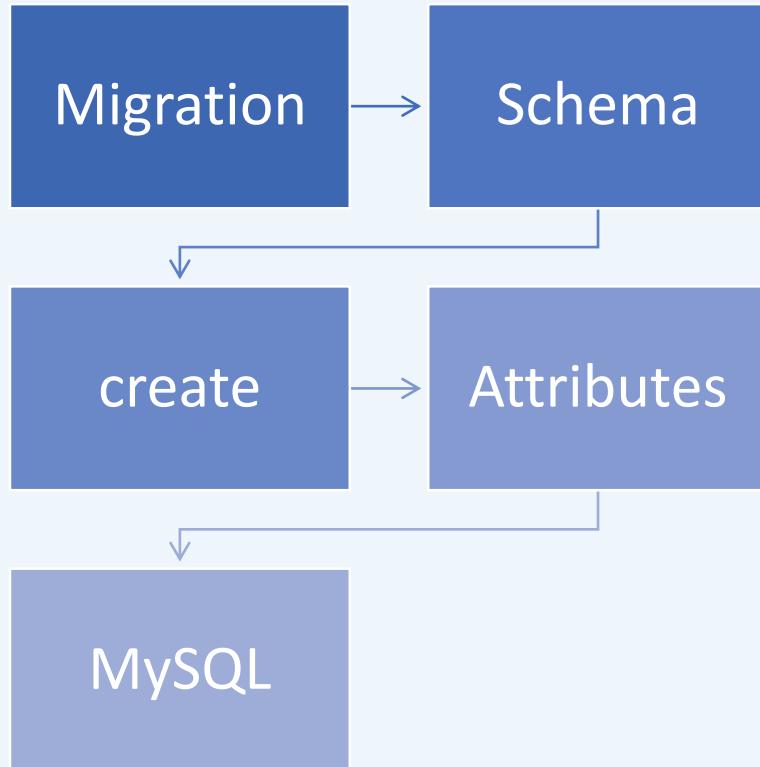
# 5. Pembangunan PL

## Perancangan Antar Muka



## 5. Pembangunan PL

### Implementasi Basis Data (Laravel)

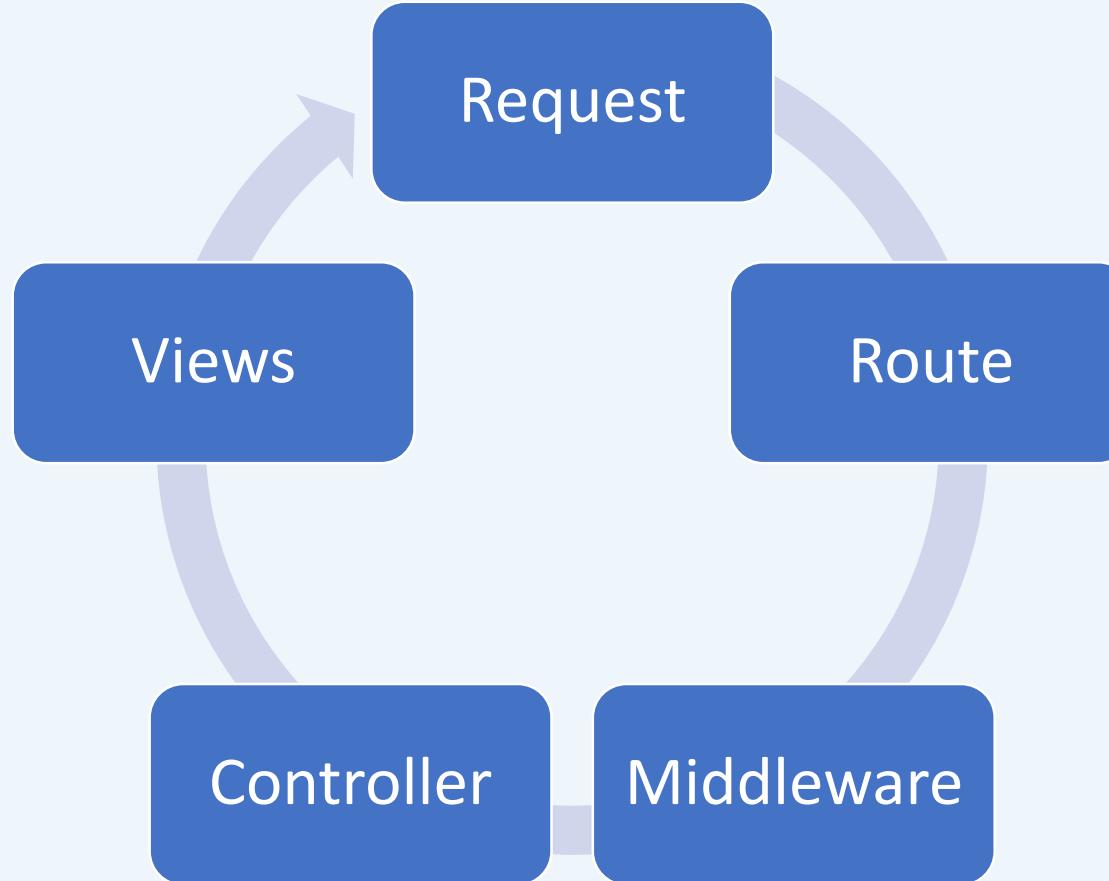


```
Schema::create('sales', function (Blueprint $table)  
{  
    $table->id('sale_id');  
    $table->integer('amount',false);  
    $table->string('sale_status', 64);  
    $table->bigInteger('total_price');  
});
```

**Kode: method up() dari kelas migration traffic**

## 5. Pembangunan PL

### Implementasi Controller



Proses request pengguna hingga tampil view secara umum.

# 5. Pembangunan PL

## Implementasi Basis Data

Kode 5.7: CreateSalesTable.php

```
1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 class CreateSalesTable extends Migration
8 {
9     /**
10      * Run the migrations.
11      *
12      * @return void
13      */
14     public function up()
15     {
16         Schema::create('sales', function (Blueprint $table) {
17             $table->id('sale_id');
18             $table->integer('amount', false);
19             $table->string('sale_status', 64);
20             $table->bigInteger('total_price');
21         });
22     }
23
24     /**
25      * Reverse the migrations.
26      *
27      * @return void
28      */
29     public function down()
30     {
31
32     }
33 }
```

## 5. Pembangunan PL

### Implementasi Antarmuka

- Menggunakan *Blade* (*templating engine* dari *framework* Laravel).
- Direktori *views* memuat 4 sub direktori dan 19 *file blade.php* pada direktori utama.
- Pembangunan antarmuka menggunakan *framework* CSS yaitu Tailwind CSS.

## 5. Pembangunan PL

### Demo Program:

<https://drive.google.com/file/d/1nAv7DRfjxFyS0VxCRISiLrTn-flgxT5B/view?usp=sharing>

## 5. Pembangunan PL

### Pengujian Program:

44 pengujian fungsional

- Pengujian dengan mencoba fitur-fitur yang dibuat dengan memberi masukan yang ideal atau sesuai permintaan input box.

8 pengujian eksperimental

- Pengujian dengan mencoba fitur-fitur yang dibuat dengan memberi masukan yang tidak ideal atau tidak sesuai permintaan input box.
- Pengujian dengan mencoba melanggar peraturan keamanan yang sudah perangkat lunak buat.
- Pengujian eksperimental menghasilkan 1 kasus dengan status tidak tepat.
- Pada halaman order jika DOM input pada HTML diedit dan dikirim ke server, nilai editan tersebut tercatat di basis data.

## 5. Pembangunan PL

### Pengujian Program:

UAT (User Acceptance Test) terhadap konsumen umum.

UAT yang ditujukan kepada konsumen umum bertujuan untuk mencari feedback dari perspektif konsumen terhadap fitur pemesanan mandiri.

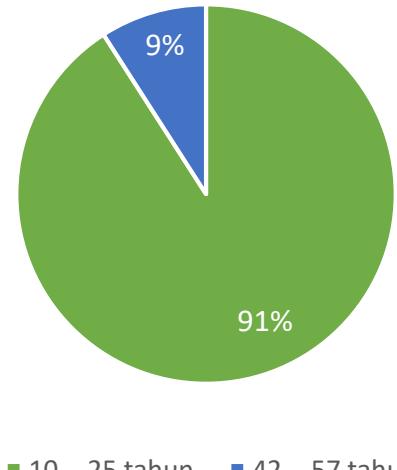
- Survei yang disebar mendapatkan 11 responden dari pengguna yang merupakan konsumen pada umumnya.
- Sebagian besar pengisi survei masuk ke kategori generasi Z
- Sebagian besar pengguna pernah bertransaksi di usaha sejenis kedai kopi modern, toko roti atau restoran sejenisnya.
- Mayoritas pengguna tertarik menggunakan fitur pesan mandiri ke depannya.

## 5. Pembangunan PL

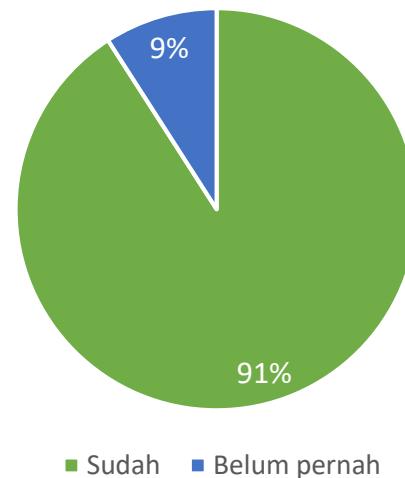
### Pengujian Program:

user acceptance test terhadap konsumen umum.

Kategori usia pengisi survei



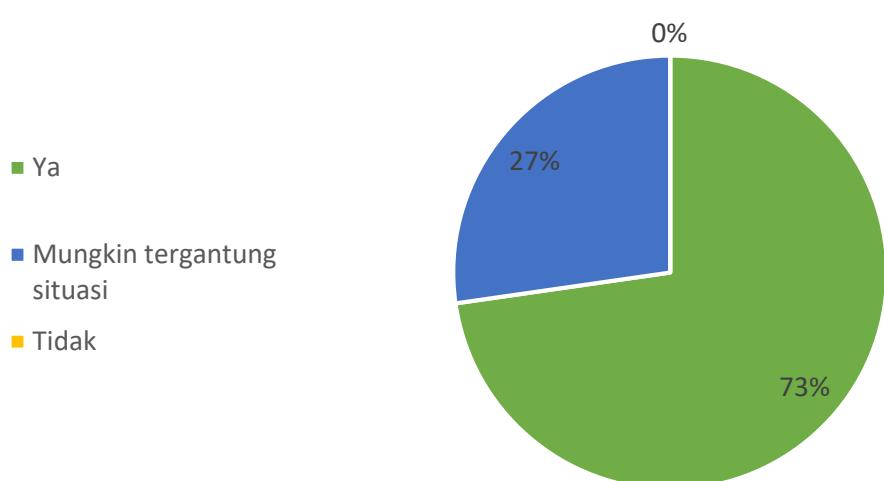
Pengguna yang bertransaksi di kedai kopi modern, toko roti atau restoran



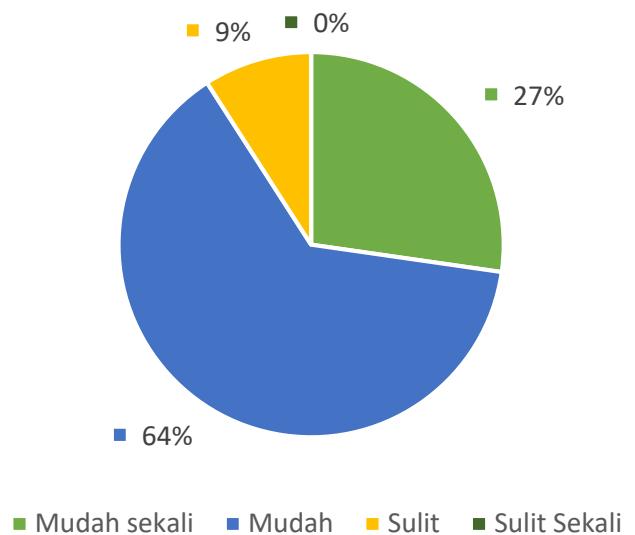
## 5. Pembangunan PL

**Pengujian Program:**  
user acceptance test  
terhadap konsumen umum.

Pengguna akan menggunakan pemesanan mandiri atau tidak



Pendapat pengguna saat melakukan pemesanan mandiri



## 5. Pembangunan PL

### Pengujian Program:

UAT (User Acceptance Test) terhadap pelaku usaha

- Halaman order: memenuhi kebutuhan tetapi ada kekurangan pada tampilan antarmuka.
- Halaman queue: memenuhi kebutuhan tetapi desain antarmuka perlu diperjelas bagian tombol order id karena ambigu desain tombolnya.
- Halaman queue: terdapat bug pada modal window konfirmasi ulang.
- Fitur traffic count: sedikit berlebihan untuk pencatatan lalu lintas mobil atau motor belum diperlukan.
- Fitur lainnya sudah memenuhi kebutuhan pelaku usaha.



- [informatika.unpar.ac.id](http://informatika.unpar.ac.id)
- ✉ [informatika@unpar.ac.id](mailto:informatika@unpar.ac.id)
- ❑ [facebook.com/if.unpar](https://facebook.com/if.unpar)
- 🐦 [@if\\_unpar](https://twitter.com/if_unpar)

