

**1. Identitas Peserta Didik**

- a. Nama : Siti Wahyuni Faujiah
- b. Kelas : XII PPLG 4
- c. NISN : 2324102136
- d. HP : 08995809779
- e. Link Repo Github : [https://github.com/zyaahhh/Arunalaco\\_siti](https://github.com/zyaahhh/Arunalaco_siti)

**2. Nama Aplikasi yang Dibuat**

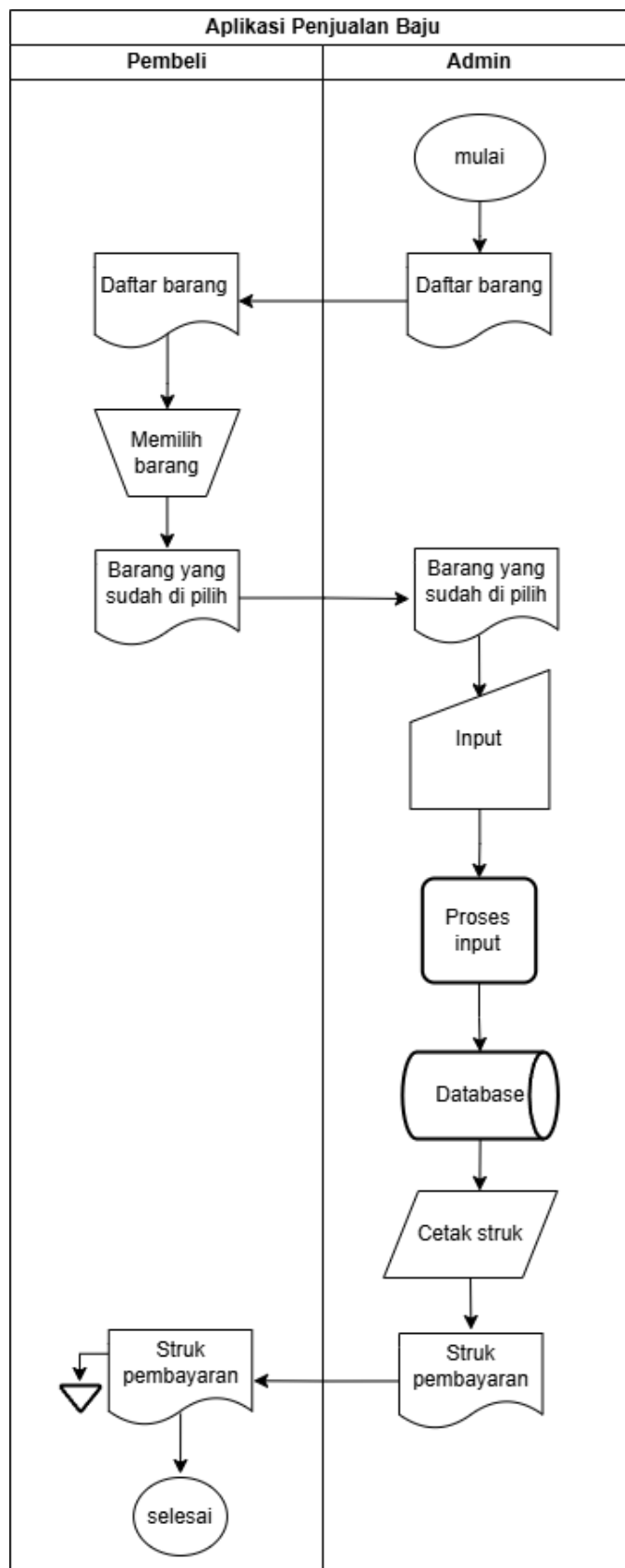
Aplikasi Penjualan Toko Baju Arunala & Co.

**3. Penjelasan / Deskripsi Aplikasi**

Aplikasi ini dibuat untuk mengelola pencatatan transaksi atau penjualan yang terjadi di sebuah toko baju, terdapat beberapa fitur yang bisa digunakan oleh admin dan pengguna aplikasi seperti pengelolaan data barang, data penjualan, detail penjualan, dashboard untuk melihat data yang diperlukan dan juga termasuk melihat stok barang yang ada. Selain itu ada fitur diskon untuk pembelian barang (jika ada) dan pencetakan struk penjualan.

**4. Analisis Kebutuhan Aplikasi (Sesuaikan dengan aplikasi yang akan dibuat)**

## a. Gambar Flowmap



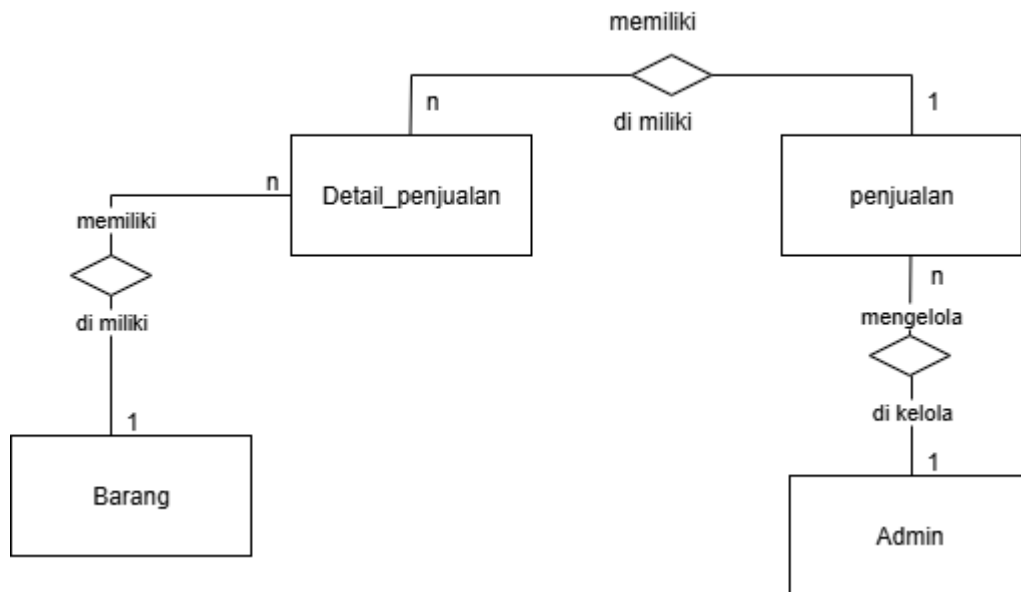
## b. Tabel Kebutuhan Fungsional

Kode	Kebutuhan fungsional	Deskripsi
001	Login	Digunakan untuk memasukkan username dan password pengguna.
002	Dashboard	Halaman utama pengguna setelah berhasil masuk
003	Mengelola data barang	Digunakan untuk menambah, mengubah, dan menghapus data barang yang tersedia.
004	Mengelola data user	Digunakan admin untuk mengelola data user.
005	Mengelola data penjualan	Digunakan untuk mencatat setiap transaksi penjualan barang yang dilakukan oleh pembeli dan dilayani oleh admin. Setiap transaksi memiliki tanggal, total harga, dan diskon (jika ada diskon).
006	Mengelola data detail penjualan	Digunakan untuk mencatat rincian barang-barang yang termasuk dalam satu transaksi penjualan. Setiap transaksi bisa terdiri dari beberapa barang.
007	Cetak struk	Digunakan untuk menghasilkan bukti pembelian atau nota dari transaksi yang sudah dilakukan. Struk berisi informasi transaksi secara lengkap.
008	Log out	Mengakhiri sesi pengguna dan mengarahkan pengguna ke halaman login

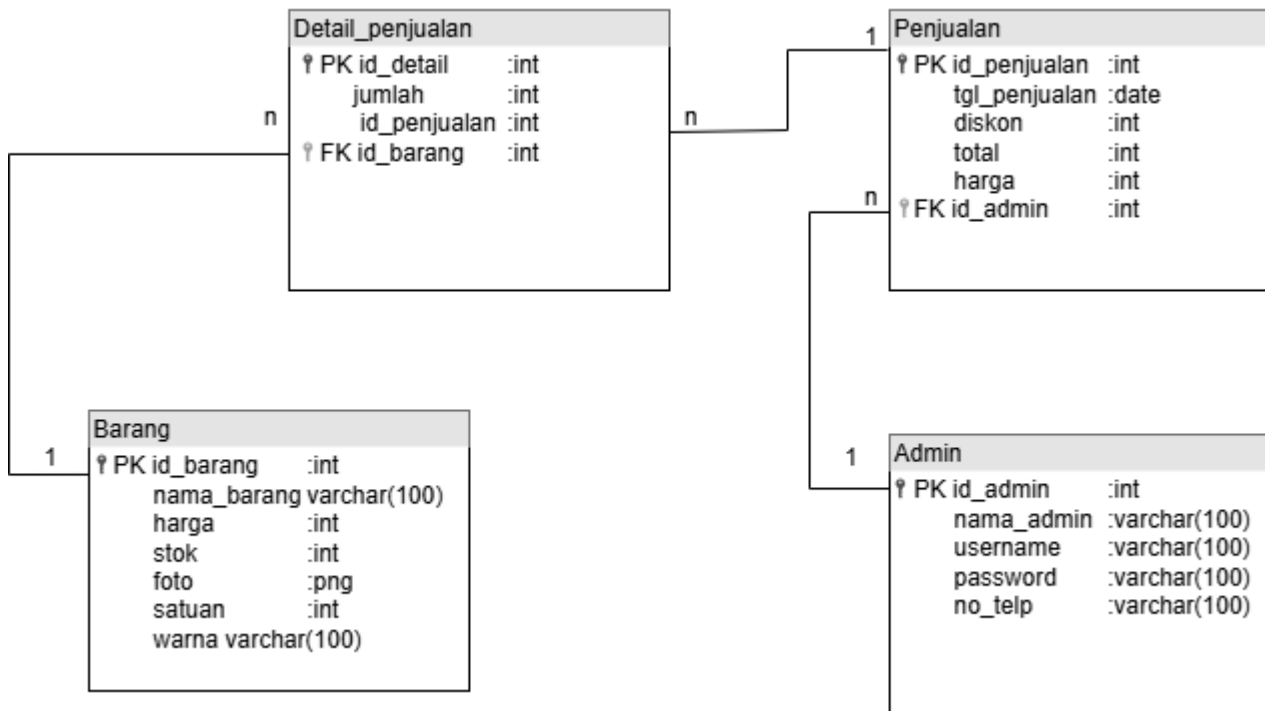
## c. Gambar Usecase



## d. Gambar ERD



## e. Gambar Skema Database



## f. Kamus data

Entity	Atribut
Barang	{id_barang, nama_barang, harga, stok, foto, satuan, warna}
Penjualan	{id_penjualan, tgl_penjualan, total, diskon, harga, id_admin}
Admin	{id_admin, nama_admin, username, password, no_telp }
Detail_penjualan	{id_detail, jumlah, id_penjualan, id_barang}

## 5. Screenshoot Data yang dibuat di VS Code

## a. File .Env

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=arunala_co_siti
DB_USERNAME=root
DB_PASSWORD=
```

## b. File Model setiap tabel

```
class Admin extends Authenticatable
{
    use Notifiable;

    protected $fillable = [
        'nama_admin',
        'username',
        'password',
        'no_telp',
    ];

    // kalau kamu gak pakai "remember_token", tambahkan ini
    public $timestamps = true;

    protected $hidden = [
        'password',
    ];

    public function penjualan()
    {
        return $this->hasMany(Penjualan::class, 'id_admin');
    }
}
```

```
use Illuminate\Database\Eloquent\Model;

class Barang extends Model
{
    protected $fillable = [
        'foto',
        'nama_barang',
        'harga',
        'stok',
        'satuan',
        'warna',
    ];

    public function detail_penjualan()
    {
        return $this->hasMany(DetailPenjualan::class, 'id_barang');
    }
}
```

```
class DetailPenjualan extends Model
{
    use HasFactory;

    protected $table = 'detail_penjualans';
    protected $fillable = [
        'id_penjualan',
        'id_barang',
        'jumlah',
    ];

    public function penjualan()
    {
        return $this->belongsTo(Penjualan::class, 'id_penjualan');
    }

    public function barang()
    {
        return $this->belongsTo(Barang::class, 'id_barang');
    }
}
```



```

class Penjualan extends Model
{
    use HasFactory;

    protected $fillable = [
        'tgl_penjualan',
        'diskon',
        'total',
        'harga',
        'id_admin',
    ];

    public function detail_penjualan()
    {
        return $this->hasMany(DetailPenjualan::class, 'id_penjualan');
    }

    public function admin()
    {
        return $this->belongsTo(Admin::class, 'id_admin');
    }
}

```

c. File Migrations setiap tabel

```

return new class extends Migration
{
    public function up(): void
    {
        Schema::create('admins', function (Blueprint $table) {
            $table->id();
            $table->string('nama_admin');
            $table->string('username')->unique();
            $table->string('password');
            $table->string('no_telp')->nullable();
            $table->timestamps();
        });
    }

    public function down(): void
    {
        Schema::dropIfExists('admins');
    }
};

```

```

    */
    public function up(): void
    {
        Schema::create('barangs', function (Blueprint $table) {
            $table->id();
            $table->string('foto')->default('nophoto.jpg');
            $table->string('nama_barang', 100);
            $table->integer('harga');
            $table->integer('stok');
            $table->string('satuan', 50);
            $table->string('warna', 50); // <-- ini penting! ha
            $table->timestamps();
        });
    }

    public function down(): void
    {
        Schema::dropIfExists('barangs');
    }
};

```

```

return new class extends Migration {
    public function up(): void
    {
        Schema::create('detail_penjualans', function (Blueprint $table) {
            $table->id();
            $table->foreignId('id_penjualan')->constrained('penjualans')
                ->onDelete('cascade');
            $table->foreignId('id_barang')->constrained('barangs')
                ->onDelete('cascade');
            $table->integer('jumlah');
            $table->timestamps();
        });
    }

    public function down(): void
    {
        Schema::dropIfExists('detail_penjualans');
    }
};

```

```

public function up(): void
{
    Schema::create('penjualans', function (Blueprint $table) {
        $table->id();
        $table->date('tgl_penjualan');
        $table->integer('diskon');
        $table->integer('total');
        $table->integer('harga');
        $table->integer('id_admin');
        $table->timestamps();
    });
}

/**
 * Reverse the migrations.
 */
public function down(): void
{
    Schema::dropIfExists('penjualans');
}
};

```

d. File Seeders setiap tabel

```

use Illuminate\Database\Seeder;
use Illuminate\Support\Facades\DB;

class BarangSeeder extends Seeder
{
    public function run(): void
    {
        DB::table('barangs')->insert([
            'foto'      => 'nophoto.jpg',
            'nama_barang' => 'Rok',
            'harga'     => 1000000, // 
            'stok'      => 100,
            'satuan'    => 1,
            'warna'     => 'merah',
        ]);
    }
}

```

```

use Illuminate\Database\Seeder;
use App\Models\Penjualan;

class PenjualanSeeder extends Seeder
{
    public function run()
    {
        Penjualan::create([
            'tgl_penjualan' => now(),
            'diskon' => 10,
            'total' => 50000,
            'harga' => 55000,
            'id_admin' => 1, |
        ]);
    }
}

```

```

use Illuminate\Database\Seeder;
use App\Models\Admin; // pastikan model Admin sudah ada

class AdminSeeder extends Seeder
{
    /**
     * Jalankan seeder untuk tabel admins.
     */
    public function run(): void
    {
        Admin::firstOrCreate(
            ['username' => 'sityy'], // field unik
            [
                'nama_admin' => 'Siti',
                'password' => bcrypt('221207'),
                'no_telp' => '08995809779',
            ]
        );
    }
}

```

```
use Illuminate\Database\Seeder;
use App\Models\DetailPenjualan;

class DetailPenjualanSeeder extends Seeder
{
    /**
     * Run the database seeds.
     */
    public function run(): void
    {
        DetailPenjualan::create([
            'id_penjualan' => 1,
            'id_barang' => 1,
            'jumlah' => 50,
        ]);
    }
}
```