**BRD**

**1. Tujuan Proyek**

Objektif:

Aplikasi ini bertujuan untuk mempermudah admin dalam mengelola data barang yang ada di toko elektronik dan memudahkan karyawan untuk melakukan pencatatan transaksi penjualan. Aplikasi ini juga memungkinkan pemantauan stok barang secara real-time, mempercepat proses input data barang baru, serta pengelolaan pemasok dan transaksi dengan mudah.

**2. Fitur Utama**

Untuk Admin

- Pengelolaan Barang: Menambah, mengubah, dan menghapus data barang.

- Pengelolaan Pemasok: Menambah, mengedit, dan menghapus pemasok.

- Transaksi Penjualan: Mencatat transaksi penjualan barang dan memperbarui stok barang secara otomatis.

- Laporan Penjualan: Menyediakan laporan penjualan untuk analisis kinerja toko.

- Pengelolaan Pengguna: Menambah, mengubah, dan menghapus pengguna dengan role yang sesuai.

Untuk Karyawan

- Melihat Data Barang: Mengakses data barang yang tersedia di toko (stok, harga, kategori).

- Mencatat Transaksi Penjualan: Melakukan transaksi penjualan barang kepada pelanggan dan mengupdate stok.

**3. Persyaratan Fungsional**

- Sistem Login

Pengguna (admin dan karyawan) akan melakukan login untuk mengakses aplikasi, dengan hak akses yang berbeda berdasarkan peran mereka.

- Pengelolaan Data Barang

Admin dapat menambah, mengubah, dan menghapus data barang, termasuk nama barang, harga, stok, kategori, dan pemasok.

- Pengelolaan Transaksi Penjualan

Karyawan akan mencatat transaksi penjualan barang, dengan otomatis memperbarui jumlah stok barang yang terjual.

- Laporan Penjualan

Admin dapat mengakses laporan penjualan untuk memantau kinerja toko dan melakukan analisis.

- Keamanan Akses Berdasarkan Peran

Admin memiliki kontrol penuh terhadap aplikasi, sementara karyawan hanya dapat mengakses dan melakukan tindakan terbatas seperti melihat barang dan mencatat transaksi penjualan.

**4. Persyaratan Non-Fungsional**

- Kegunaan

Antarmuka pengguna harus mudah digunakan oleh admin dan karyawan untuk melakukan tugas-tugas yang diperlukan, seperti pencatatan transaksi, melihat laporan, dan mengelola data barang.

- Keamanan

Aplikasi harus aman dengan sistem login yang mengharuskan pengguna untuk memasukkan kredensial yang valid. Selain itu, hak akses dibatasi sesuai dengan peran pengguna (admin dan karyawan).

**5. Model, Migrasi, Seeder, dan Resource yang Perlu Dibuat**

**Barang (Products)**

Model: Product

- Menyimpan informasi terkait barang (nama, harga, stok, kategori, pemasok).

Migration:

Struktur tabel `products` yang akan dibuat pada database:

- `id`: bigint UNSIGNED (Primary Key)

- `name`: varchar(255)

- `price`: decimal

- `stock`: int

- `category`: varchar(255)

- `supplier\_id`: bigint (Foreign Key)

- `created\_at`: timestamp

- `updated\_at`: timestamp

**Pemasok (Suppliers)**

Model: Supplier

- Menyimpan informasi terkait pemasok barang.

Migration:

Struktur tabel `suppliers`:

- `id`: bigint UNSIGNED (Primary Key)

- `name`: varchar(255)

- `contact\_info`: varchar(255)

- `created\_at`: timestamp

- `updated\_at`: timestamp

**Transaksi Penjualan (Sales Transactions)**

Model: Transaction

- Menyimpan informasi transaksi penjualan barang.

Migration:

Struktur tabel `transactions`:

- `id`: bigint UNSIGNED (Primary Key)

- `product\_id`: bigint (Foreign Key)

- `quantity`: int

- `total\_price`: decimal

- `transaction\_date`: timestamp

- `user\_id`: bigint (Foreign Key) (Karyawan yang mencatat transaksi)

**Pengguna (Users)**

Model: `User`

- Menyimpan informasi pengguna (admin dan karyawan).

Migration:

Struktur tabel `users`:

- `id`: bigint UNSIGNED (Primary Key)

- `name`: varchar(255)

- `email`: varchar(255)

- `password`: varchar(255)

- `role`: varchar(255)

- `created\_at`: timestamp

- `updated\_at`: timestamp

**Permissions (Hak Akses)**

Model: `Permission`

- Mengelola hak akses yang diizinkan untuk pengguna (admin dan karyawan).

Seeder:

Seeder akan digunakan untuk menambahkan permissions yang diperlukan, misalnya:

- `view\_products`: Mengizinkan karyawan melihat barang.

- `manage\_products`: Mengizinkan admin mengelola barang.

- `view\_transactions`: Mengizinkan karyawan melihat transaksi.

- `manage\_transactions`: Mengizinkan admin mencatat transaksi.

**6. Analisis Permissions untuk Karyawan dan Admin**

Pada aplikasi manajemen barang ini, permissions diperlukan untuk membatasi akses berdasarkan peran pengguna. Admin memiliki akses penuh ke seluruh sistem, sementara karyawan hanya memiliki akses terbatas.

* **Permissions untuk Karyawan**

- `view\_products`: Mengizinkan karyawan melihat data barang yang tersedia di toko.

- `manage\_sales`: Mengizinkan karyawan untuk mencatat transaksi penjualan barang.

* **Permissions untuk Admin**

- `view\_products`: Mengizinkan admin melihat data barang.

- `manage\_products`: Mengizinkan admin menambah, mengubah, dan menghapus data barang.

- `view\_transactions`: Mengizinkan admin melihat transaksi penjualan.

- `manage\_transactions`: Mengizinkan admin mengelola transaksi penjualan.

- `manage\_users`: Mengizinkan admin mengelola pengguna (admin dan karyawan).

**7. Mengapa Migration Permissions Tidak Dibuat?**

Karena saat membuat proyek baru dengan perintah `composer create-project --prefer-dist raugadh/fila-starter .`, migrasi untuk tabel permissions sudah disediakan oleh paket \*\*spatie/laravel-permission\*\*. Paket ini secara otomatis mengatur tabel dan kolom yang diperlukan untuk permissions dan roles, sehingga tidak perlu membuat migrasi permissions secara manual.

**Resources untuk Aplikasi Toko Elektronik:**

1. **ProductResource**
   * **Deskripsi**: Menyediakan akses untuk data barang (produk) di toko elektronik, memungkinkan admin untuk melihat, menambah, mengedit, dan menghapus barang.
   * **Endpoint**: /api/products
     + **GET**: Mengambil daftar semua barang.
     + **POST**: Menambahkan barang baru.
     + **PUT/PATCH**: Memperbarui data barang yang ada.
     + **DELETE**: Menghapus barang.
2. **CategoryResource**
   * **Deskripsi**: Menyediakan akses untuk kategori barang yang tersedia di toko elektronik.
   * **Endpoint**: /api/categories
     + **GET**: Mengambil daftar kategori barang.
     + **POST**: Menambahkan kategori baru.
     + **PUT/PATCH**: Memperbarui kategori barang.
     + **DELETE**: Menghapus kategori barang.
3. **TransactionResource**
   * **Deskripsi**: Menyediakan akses untuk data transaksi penjualan barang, memungkinkan admin untuk melihat dan mengelola transaksi yang terjadi.
   * **Endpoint**: /api/transactions
     + **GET**: Mengambil daftar transaksi.
     + **POST**: Mencatat transaksi baru.
     + **PUT/PATCH**: Memperbarui transaksi yang ada.
     + **DELETE**: Menghapus transaksi.
4. **UserResource**
   * **Deskripsi**: Menyediakan akses untuk data pengguna yang terdaftar di aplikasi (misalnya admin dan karyawan).
   * **Endpoint**: /api/users
     + **GET**: Mengambil daftar pengguna.
     + **POST**: Menambahkan pengguna baru (admin atau karyawan).
     + **PUT/PATCH**: Memperbarui data pengguna.
     + **DELETE**: Menghapus pengguna.
5. **InventoryResource**
   * **Deskripsi**: Menyediakan akses untuk data inventaris barang, memudahkan admin untuk melihat stok barang yang tersedia di toko.
   * **Endpoint**: /api/inventory
     + **GET**: Mengambil daftar inventaris barang.
     + **POST**: Menambahkan barang ke inventaris.
     + **PUT/PATCH**: Memperbarui stok barang.
     + **DELETE**: Menghapus barang dari inventaris.
6. **PermissionResource**
   * **Deskripsi**: Menyediakan akses untuk mengelola hak akses (permissions) untuk admin dan karyawan dalam aplikasi.
   * **Endpoint**: /api/permissions
     + **GET**: Mengambil daftar permissions.
     + **POST**: Menambahkan permission baru.
     + **PUT/PATCH**: Memperbarui permission yang ada.
     + **DELETE**: Menghapus permission.
7. **RoleResource**
   * **Deskripsi**: Menyediakan akses untuk mengelola role (peran) pengguna, seperti admin dan karyawan, dalam aplikasi.
   * **Endpoint**: /api/roles
     + **GET**: Mengambil daftar role yang ada.
     + **POST**: Menambahkan role baru.
     + **PUT/PATCH**: Memperbarui role yang ada.
     + **DELETE**: Menghapus role.

Dokumen ini mengatur kebutuhan fungsional dan non-fungsional dari aplikasi manajemen data barang di toko elektronik, serta struktur database dan hak akses pengguna. Dengan demikian, aplikasi ini dapat memberikan solusi efektif dalam pengelolaan barang dan transaksi di toko elektronik.