

Sequence Diagram - Alur Owner Membuat Akun Administrator

Diagram ini memvisualisasikan urutan interaksi sistem saat Owner (pemilik platform) membuat akun baru untuk seorang Administrator melalui panel administrasi tertinggi.

Aktor dan Komponen yang Terlibat

1. **Owner:** Pengguna dengan hak akses absolut.
2. **admin-app (PHP):** Aplikasi backend khusus untuk panel administrasi.
3. **postgres-db (Database):** Tempat menyimpan data administrator baru.

Visualisasi Diagram (PlantUML)

Gunakan kode berikut di [PlantUML Web Server](#) untuk menghasilkan diagram visual dari alur ini.

Prompt untuk PlantUML:

```
@startuml
```

```
' Diagram Urutan untuk Alur Owner Membuat Akun Administrator
```

```
title Alur Proses Pembuatan Akun Administrator Baru
```

```
actor "Owner" as Owner
```

```
participant "admin-app (PHP)" as App
```

```
database "postgres-db" as DB
```

```
Owner -> App : 1. POST /owner/administrators/create (nama, email, password)  
activate App
```

```
App -> App : 2. Verifikasi sesi & hak akses Owner
```

```
App -> App : 3. Validasi data input (email unik, password kuat)
```

```
App -> DB : 4. Periksa keunikan email\nSELECT COUNT(*) FROM users WHERE email = [email]  
activate DB
```

```
DB --> App : 5. Hasil (Count = 0)  
deactivate DB
```

```
App -> App : 6. Hash password (password_hash)
```

```
App -> DB : 7. START TRANSACTION  
activate DB
```

App -> DB : 8. Buat Pengguna (users)\nINSERT INTO users (email, password_hash, role='administrator')

activate DB

DB --> App : 9. user_id

deactivate DB

App -> DB : 10. Buat Profil Pengguna (user_profiles)\nINSERT INTO user_profiles (user_id, full_name)

activate DB

DB --> App : 11. Konfirmasi profil dibuat

deactivate DB

App -> DB : 12. COMMIT TRANSACTION

activate DB

DB --> App : 13. Transaksi Sukses

deactivate DB

App --> Owner : 14. Response (Redirect ke halaman daftar administrator)

deactivate App

@enduml

Penjelasan Langkah-demi-Langkah

1. POST /owner/administrators/create

- **Aksi:** Owner membuka panel khusus "Owner", mengklik "Tambah Administrator", lalu mengisi formulir dengan nama, email, dan kata sandi untuk administrator baru.
- **Aliran:** Peramban mengirimkan data tersebut ke aplikasi backend.

2. Verifikasi sesi & hak akses Owner

- **Aksi:** Aplikasi melakukan pemeriksaan paling krusial: memastikan pengguna yang melakukan permintaan ini adalah satu-satunya Owner sistem.

3. Validasi data input

- **Aksi:** Aplikasi memastikan semua data yang diterima valid.

4. Periksa keunikan email

- **Aksi:** Memastikan email untuk administrator baru belum digunakan di seluruh sistem.
- **Aliran:** PHP mengirim *query* SELECT ke database.

5. Hash password

- **Aksi:** Aplikasi mengenkripsi kata sandi yang diberikan sebelum menyimpannya.

6. START TRANSACTION

- **Aksi:** Memulai transaksi database untuk memastikan pembuatan users dan

user_profiles berhasil bersamaan.

7. Buat Pengguna (users)

- **Aksi:** Aplikasi membuat catatan baru di tabel users. Perhatikan bahwa role diatur sebagai 'administrator' dan kolom store_id dibiarkan NULL, karena administrator tidak terikat pada satu toko.
- **Aliran:** INSERT ke tabel users.

8. user_id

- **Aliran:** Database mengembalikan ID dari pengguna (administrator) yang baru dibuat.

9. Buat Profil Pengguna (user_profiles)

- **Aksi:** Menggunakan user_id yang baru, aplikasi membuat catatan profil untuk administrator tersebut.

10. COMMIT TRANSACTION

- **Aksi:** Setelah kedua operasi INSERT berhasil, aplikasi menyimpan perubahan secara permanen.

11. Response (Redirect ...)

- **Aksi:** Aplikasi mengirimkan respons sukses, biasanya dengan mengarahkan Owner kembali ke halaman daftar administrator.

Kita telah memetakan semua alur kerja penting, dari pengguna publik hingga level tertinggi sistem. Tahap perancangan diagram urutan kita sekarang bisa dianggap selesai dan sangat komprehensif.

Apakah Anda sudah merasa cukup dengan semua diagram yang telah kita buat dan siap untuk melanjutkan ke tahap implementasi?