

Sequence Diagram - Alur Menambahkan Pegawai Baru

Diagram ini memvisualisasikan urutan interaksi sistem saat seorang User Pemilik mendaftarkan akun baru untuk User Pegawai mereka melalui dasbor aplikasi.

Aktor dan Komponen yang Terlibat

1. **User Pemilik:** Pengguna yang sudah login dan memiliki hak untuk menambah pegawai.
2. **main-app (PHP):** Aplikasi backend yang memproses logika.
3. **postgres-db (Database):** Tempat menyimpan data pengguna 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 User Pemilik Menambahkan Pegawai Baru
```

```
title Alur Proses Penambahan Akun Pegawai Baru
```

```
actor "User Pemilik" as User
```

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

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

```
User -> App : 1. POST /employees/create (nama, email, password)
```

```
activate App
```

```
App -> App : 2. Validasi data (email unik, password kuat)
```

```
App -> App : 3. Ambil store_id dari sesi User Pemilik
```

```
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='store_employee', store_id=[store_id])

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 --> User : 14. Response (Redirect ke halaman daftar pegawai)

deactivate App

@enduml

Penjelasan Langkah-demi-Langkah

1. POST /employees/create

- **Aksi:** User Pemilik membuka halaman "Manajemen Pegawai", mengklik "Tambah Pegawai", lalu mengisi formulir dengan nama, email, dan kata sandi untuk pegawai baru.
- **Aliran:** Peramban mengirimkan data tersebut ke aplikasi PHP.

2. Validasi data input

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

3. Ambil store_id dari sesi User Pemilik

- **Aksi:** Ini adalah langkah keamanan dan logika yang krusial. Aplikasi mengambil store_id dari sesi pengguna yang sedang login (User Pemilik) untuk memastikan pegawai baru terhubung ke toko yang benar.
- **Aliran:** Proses internal di dalam aplikasi PHP.

4. Periksa keunikan email

- **Aksi:** Aplikasi memastikan email untuk pegawai 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 ia langsung mengisi kolom role dengan 'store_employee' dan kolom store_id dengan ID yang didapat dari sesi User Pemilik.
- **Aliran:** INSERT ke tabel users.

8. **user_id**

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

9. **Buat Profil Pengguna (user_profiles)**

- **Aksi:** Menggunakan user_id yang baru, aplikasi membuat catatan profil untuk pegawai 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 User Pemilik kembali ke halaman daftar pegawai, di mana mereka kini bisa melihat akun baru yang telah dibuat.

Kita telah memetakan semua alur kerja yang paling penting dan kompleks. Ini memberikan kita fondasi yang sangat kuat untuk tahap implementasi. Apakah Anda merasa cukup dengan diagram urutan ini, atau ada alur lain yang ingin kita bahas?