

# Test Case Scenarios

---

## Positive Testing

### test\_can\_create\_karyawan

Skenario: Pengguna membuat data karyawan baru.

Langkah:

- 1. Nonaktifkan middleware CSRF.
- 2. Kirim permintaan POST untuk membuat karyawan dengan data `nama`, `jabatan`, dan `gaji`.
- 3. Verifikasi bahwa respons menghasilkan status 302 (berhasil dan dialihkan).
- 4. Pastikan bahwa satu data karyawan telah dibuat di database.

### test\_can\_read\_karyawan

Skenario: Pengguna membaca data karyawan yang ada.

Langkah:

- 1. Nonaktifkan middleware CSRF.
- 2. Buat user menggunakan factory dan autentikasi sebagai user tersebut.
- 3. Buat data karyawan menggunakan factory.
- 4. Kirim permintaan GET untuk membaca data karyawan.
- 5. Verifikasi bahwa respons menampilkan `nama`, `jabatan`, dan `gaji` karyawan.

### test\_can\_update\_karyawan

Skenario: Pengguna memperbarui data karyawan yang ada.

Langkah:

- 1. Nonaktifkan middleware CSRF.
- 2. Buat data karyawan awal menggunakan model Eloquent.
- 3. Buat user dan autentikasi sebagai user tersebut.
- 4. Kirim permintaan PUT untuk memperbarui data karyawan dengan ID yang spesifik.
- 5. Verifikasi bahwa tidak ada error dan dialihkan setelah pembaruan berhasil.
- 6. Muat ulang model dari database.
- 7. Verifikasi bahwa data telah diperbarui dengan benar.

### test\_can\_delete\_karyawan

Skenario: Pengguna menghapus data karyawan yang ada.

Langkah:

- 1. Nonaktifkan middleware CSRF.
- 2. Buat data karyawan awal menggunakan model Eloquent.
- 3. Buat user dan autentikasi sebagai user tersebut.
- 4. Kirim permintaan DELETE untuk menghapus data karyawan dengan ID yang spesifik.
- 5. Verifikasi bahwa tidak ada error dan dialihkan setelah penghapusan berhasil.

## Negative Testing

### test\_cannot\_create\_karyawan\_without\_required\_fields

Skenario: Pengguna tidak dapat membuat karyawan tanpa mengisi field yang diperlukan.

Langkah:

- 1. Nonaktifkan middleware CSRF.
- 2. Kirim permintaan POST untuk membuat karyawan tanpa data.
- 3. Verifikasi bahwa respons menghasilkan status 302 karena validasi gagal dan dialihkan kembali.
- 4. Pastikan tidak ada data yang disimpan dalam database.

### test\_cannot\_read\_nonexistent\_karyawan

Skenario: Pengguna tidak dapat membaca data karyawan yang tidak ada.

Langkah:

- 1. Buat user menggunakan factory dan autentikasi sebagai user tersebut.
- 2. Kirim permintaan GET untuk membaca karyawan dengan ID yang tidak ada.
- 3. Verifikasi bahwa respons menghasilkan status 404 karena karyawan tidak ditemukan.

### test\_cannot\_update\_karyawan\_without\_required\_fields

Skenario: Pengguna tidak dapat memperbarui data karyawan tanpa mengisi field yang diperlukan.

Langkah:

- 1. Nonaktifkan middleware CSRF.
- 2. Buat data awal karyawan menggunakan factory.
- 3. Buat user dan autentikasi sebagai user tersebut.
- 4. Kirim permintaan PUT untuk memperbarui karyawan tanpa data.

- 5. Verifikasi bahwa respons menghasilkan status 302 karena validasi gagal dan dialihkan kembali.
- 6. Pastikan data karyawan tidak berubah.

### **test\_cannot\_delete\_nonexistent\_karyawan**

Skenario: Pengguna tidak dapat menghapus data karyawan yang tidak ada.

Langkah:

- 1. Buat user menggunakan factory dan autentikasi sebagai user tersebut.
- 2. Kirim permintaan DELETE untuk menghapus karyawan dengan ID yang tidak ada.
- 3. Verifikasi bahwa respons menghasilkan status 404 karena karyawan tidak ditemukan.