

## 2. Exploratory Testing

### A. Bagaimana cara melakukan exploratory testing?

melakukan **Exploratory Testing** dengan pendekatan **Session-Based Testing**, yaitu:

#### a. Mempersiapkan Scope Testing:

- Memahami fungsi umum fitur berdasarkan UI/UX yang terlihat.
- Menentukan area mana yang ingin diuji terlebih dahulu (misal: form, tombol aksi, validasi input, flow submit).

#### b. Membuat Charter Testing:

- Menentukan tujuan kecil dalam waktu tertentu, misal: "Menguji form input laporan apakah menerima data tidak valid".

#### c. Melakukan Testing Secara Bertahap dan Bebas:

- Berinteraksi langsung dengan fitur tanpa mengikuti dokumentasi.
- Mencoba berbagai skenario normal, edge case, dan skenario error.
- Mencatat semua perilaku sistem baik yang sesuai ekspektasi maupun yang mencurigakan.

#### d. Mencatat Temuan (Bugs, Error, Behavior Aneh)

- Langsung mendokumentasikan temuan bug beserta langkah reproduksi, screenshot atau screen recording.
- Mengkategorikan temuan berdasarkan severity (Critical, Major, Minor).

#### e. Evaluasi dan Memberikan Feedback

- Memberikan rekomendasi perbaikan terhadap usability, bug, atau risiko error dari sisi user experience.

### B. Hal yang akan diperiksa saat melakukan Exploratory testing

#### a. Validasi Form dan Field Input

- Memastikan field wajib diisi (required) berfungsi.
- Mengetes behavior input field terhadap data tidak valid (angka di teks, simbol aneh, input kosong).

#### b. Behavior Action Button

- Memastikan tombol seperti Submit, Cancel, Upload berfungsi sesuai ekspektasi.
- Mencoba klik tombol dalam kondisi form kosong, valid, invalid, atau klik ganda.

#### c. Error Handling dan Pesan Notifikasi

- Menilai apakah error muncul saat seharusnya.
- Memastikan pesan error informatif dan tidak membingungkan user.
- Cek behavior aplikasi saat jaringan lambat, file upload error, atau server error.