

TEST CASE

1. Test Case 1: Login Pengguna

- **Tujuan Pengujian** : Untuk memastikan user/pengguna dapat login/masuk ke dalam aplikasi dengan menggunakan akun yang valid.
- **Langkah Pengujian** :
 - Buka aplikasi FoodFast.
 - Masukkan email/username dan password yang valid.
 - Klik tombol "Login/Masuk".
- **Data yang Diperlukan** : email/username(user@mail.com), password(password123).
- **Hasil yang Diharapkan** : pengguna berhasil login dan langsung diarahkan ke halaman atau menu utama dari aplikasi.

2. Test Case 2: Pencarian Restoran

- **Tujuan Pengujian** : Untuk memastikan fitur pencarian restoran berfungsi serta menampilkan hasil yang pencarian yang sesuai.
- **Langkah Pengujian** :
 - Masuk ke aplikasi,
 - Cari dan tekan kolom pencarian atau tombol "Cari Restoran".
 - Masukkan kata kunci pencarian/kategori restoran (contoh: "Bakwan Kawi" / "Jogja") di kolom pencarian
 - Tekan tombol "Cari".
- **Data yang Diperlukan**: Kata kunci pencarian: Bakwan Kawi
- **Hasil yang Diharapkan**: Menampilkan daftar restoran sesuai dengan kata kunci yang digunakan.

3. Test Case 3: Menambahkan Item ke Keranjang

- **Tujuan Pengujian**: Untuk memastikan bahwa user/pengguna dapat menambahkan item ke keranjang belanja.
- **Langkah Pengujian**:
 - Masuk ke aplikasi.
 - Pilih salah satu restoran dari hasil pencarian.
 - Pilih satu atau beberapa item makanan yang ditampilkan.
 - Tekan tombol "Tambahkan ke Keranjang" pada item yang diinginkan.

- **Data yang Diperlukan :** Makanan(Bakwan Kawi), Jumlah(3).
- **Hasil yang Diharapkan :** Data makanan berhasil ditambahkan kedalam keranjang beserta jumlah dan total harga yang sesuai.

4. Test Case 4: Checkout Pesanan

- **Tujuan Pengujian:** Untuk memastikan user/pengguna dapat melakukan checkout dan pembayaran pesanan.
- **Langkah Pengujian:**
 1. Masuk ke aplikasi.
 2. Pilih item makanan untuk ditambahkan ke dalam keranjang.
 3. Tekan tombol "Checkout".
 4. Pilih metode pembayaran (contoh: e-wallet).
 5. Tekan tombol "Bayar/Konfirmasi Pesanan".
- **Data yang Diperlukan:** Metode pembayaran: e-wallet(gopay, shopeepay).
- **Hasil yang Diharapkan:** Pesanan berhasil diproses dan pengguna diarahkan ke halaman konfirmasi pesanan yang baru saja dibuat.

5. Test Case 5: Notifikasi Untuk Pesanan Berhasil

- **Tujuan Pengujian:** Untuk memastikan user/pengguna menerima notifikasi pemberitahuan untuk pesanan yang telah berhasil dibuat.
- **Langkah Pengujian:**
 1. Masuk ke aplikasi.
 2. Lakukan proses pemesanan hingga checkout.
 3. Tunggu agar sistem memproses pesanan.
 4. Periksa apakah notifikasi muncul pada layar aplikasi atau melalui push notification.
- **Data yang Diperlukan:** Metode pembayaran: e-wallet.
- **Hasil yang Diharapkan:** Pengguna menerima notifikasi bahwa pesanan telah berhasil dibuat.

6. Test Case 6: Pembayaran Tidak Berhasil (Saldo Tidak Mencukupi)

- **Tujuan Pengujian:** Untuk memastikan pembayaran gagal dengan notifikasi yang sesuai jika saldo e-wallet tidak mencukupi.
- **Langkah Pengujian:**
 - Masuk ke aplikasi.

- Lakukan pemesanan hingga tahap pembayaran.
- Pilih “E-Wallet” dengan saldo yang tidak mencukupi.
- Masukkan PIN pembayaran.
- Tekan tombol “Bayar”.
- **Data yang Diperlukan:** Saldo e-wallet (Rp 1000), total pesanan (Rp 25000).
- **Hasil yang Diharapkan:** Muncul pesan/notifikasi kesalahan: "Saldo tidak mencukupi. Silakan pilih metode pembayaran lain".

7. Test Case 7: Pembatasan Jumlah Maximal Item Pesanan

- **Tujuan Pengujian:** Untuk memastikan bahwa aplikasi/sistem dapat membatasi jumlah maksimal pesanan tiap item.
- **Langkah Pengujian:**
 - Masuk ke aplikasi.
 - Tambahkan makanan ke keranjang dengan jumlah melebihi batas maksimal (contoh: Gado-gado : 100).
 - Tekan tombol "Tambah ke Keranjang"
- **Data yang Diperlukan:** Makanan: Gado-gado, Jumlah: 100.
- **Hasil yang Diharapkan:** Aplikasi menampilkan pesan/pop up error bahwa jumlah pesanan melebihi batas maksimal yang dibolehkan dalam satu aktivitas pesanan.

ANALISIS POTENSI BUG

1. Bug 1: User Gagal Login

- **Penyebab Bug:** Kesalahan validasi pada database ataupun pada server.
- **Dampak pada Pengguna:**
 - Pengguna tidak dapat masuk ke aplikasi.
 - Pengguna tidak dapat melakukan pemesanan makanan.
- **Saran Perbaikan:**
 - Periksa pesan error yang ada pada log server.
 - Periksa source code validasi untuk login pengguna berjalan dengan urutan yang benar dan dapat berkomunikasi dengan database.
 - Uji coba ulang pada skenario login pengguna.
 - Buat beberapa metode untuk login sebagai cadangan agar aplikasi tetap dapat digunakan.

2. Bug 2: Item Tidak Masuk ke Keranjang atau Item di Keranjang Hilang

- **Penyebab Bug:**
 - Ketidaksinkronan antara tampilan (front-end) dan proses (back-end) ketika menambahkan item baru ke dalam keranjang belanja.
 - Masalah penyimpanan lokal (session/cache) yang tidak terupdate secara real-time maupun penyimpanan yang bersifat tidak permanen (persisten).
- **Dampak pada Pengguna:**
 - Pengguna tidak dapat melakukan checkout pesanan.
 - Pengguna harus mengulang dari awal untuk menambahkan item pesanan yang diinginkan.
- **Saran Perbaikan:**
 - Perbaiki mekanisme state penyimpanan untuk keranjang dan memberikan validasi bahwa item berhasil ditambahkan.
 - Menambahkan notifikasi atau pesan pop up pada aplikasi sebagai feedback visual pada pengguna disetiap aksi atau aktivitas menambahkan item di keranjang.
 - Menerapkan sinkronisasi penyimpanan otomatis agar data diperbarui secara real-time.

3. Bug 3: Notifikasi Pesanan Berhasil Tidak Muncul

- **Penyebab Bug:**

- Terjadi masalah pada sistem notifikasi maupun terjadinya delay komunikasi antara aplikasi dan server.
- Terjadi kesalahan pada proses update status pesanan.

- **Dampak pada Pengguna:**

- Pengguna tidak mengetahui pesanan berhasil dibuat atau tidak.
- Pengguna ragu dengan status pesanan yang dibuat sehingga dapat menyebabkan pemesanan yang berulang-ulang.

- **Saran Perbaikan:**

- Menerapkan mekanisme pengulangan (retry) untuk pesan notifikasi.
- Penambahan feedback ke pengguna berupa tampilan status pesanan di aplikasi untuk memberikan informasi yang akurat.
- Menambahkan alternatif notifikasi lain, seperti email otomatis.