

CORE ACTIVITIES IN OTOKU PROTOTYPE

1. Activity: Emergency Assist (Bantuan Darurat)

“Ketika pengguna mengalami kendala kendaraan di jalan dan membutuhkan bantuan segera.”

Tasks:

1. Menentukan lokasi darurat pengguna
2. Memilih jenis masalah (ban bocor, mesin mati, aki drop, dll.)
3. Memanggil mekanik darurat terdekat
4. Mengonfirmasi permintaan bantuan

Actions:

- Pengguna membuka fitur Emergency Assist
- Sistem otomatis mengambil lokasi GPS pengguna
- Pengguna memilih jenis masalah → “Ban bocor”, “Mesin tidak menyala”, “Aki mati”
- Pengguna menekan tombol “Panggil Mekanik Darurat”
- Pengguna melihat estimasi respon: jarak mekanik, ETA, biaya estimasi

Operations:

- Aplikasi mengakses modul GPS Location Fetching
- Sistem melakukan pencarian mekanik dalam radius tertentu
- Sistem mengirim notifikasi darurat ke mekanik terdekat
- Mekanik menerima tugas prioritas (emergency flag = ON)
- Server mencatat permintaan darurat & status real-time

2. Activity: Tracking Mekanik (Real-time Mechanic Tracking)

“Mengikuti perjalanan mekanik menuju lokasi pengguna.”

Tasks:

1. Membuka halaman tracking
2. Melihat posisi mekanik di peta
3. Memahami estimasi waktu kedatangan (ETA)
4. Berkomunikasi dengan mekanik

Actions:

- Pengguna menekan tombol “Lacak Mekanik”
- Aplikasi menampilkan peta real-time
- Pengguna melihat ikon posisi mekanik
- Pengguna melihat progres perjalanan & ETA
- Pengguna dapat chat atau telepon mekanik

Operations:

- Sistem memproses data GPS mekanik
- Backend mengupdate lokasi setiap 5 detik
- Peta dirender menggunakan Google Maps API
- Sistem mengkalkulasi ETA berdasarkan jarak & kondisi lalu lintas

3. Activity: Smart Maintenance Service (Perawatan Kendaraan Pintar)

“Mengelola jadwal servis rutin, permintaan servis, dan riwayat servis kendaraan.”

Tasks:

1. Membuka menu Smart Maintenance
2. Melihat status kesehatan kendaraan
3. Menerima pengingat servis otomatis
4. Melakukan pemesanan servis rutin
5. Melihat riwayat perawatan

Actions:

- Pengguna membuka tab “Smart Maintenance”
- Pengguna melihat rekomendasi: ganti oli, rem, tune-up
- Pengguna menerima notifikasi:
- “Oli terakhir diganti 3.000 km yang lalu. Waktunya servis!”
- Pengguna memilih jadwal servis → tanggal, jam, mekanik
- Pengguna melihat histori servis: tanggal, biaya, mekanik

Operations:

- Sistem membaca data input pengguna (kilometer / waktu)
- Algoritma memproses rekomendasi servis
- Database menyimpan riwayat servis
- Sistem membuat jadwal & menugaskan mekanik
- Notifikasi push dikirim otomatis

4. Activity: Pemesanan Mekanik Biasa (Regular Mechanic Request)

“Pengguna memesan jasa mekanik non-darurat.”

Tasks:

1. Memilih jenis layanan
2. Memilih lokasi servis (rumah / jalan / kantor)
3. Memilih mekanik atau bengkel terdekat
4. Mengonfirmasi pemesanan

Actions:

- Pengguna memilih “Servis Umum”
- Pengguna memasukkan lokasi atau memilih GPS
- Pengguna memilih mekanik berdasarkan rating
- Pengguna melihat detail biaya
- Pengguna menekan “Pesan Sekarang”

Operations:

- Sistem membuat request baru
- Backend menghubungkan pengguna dengan mekanik
- Data pemesanan disimpan ke server
- Mekanik menerima notifikasi pekerjaan baru

5. Activity: Otoku Rewards (Sistem Poin & Gamifikasi)

“Pengguna memperoleh poin dari penggunaan aplikasi dan menukarkannya dengan benefit.”

Tasks:

1. Melihat saldo poin
2. Mengumpulkan poin dari aktivitas: servis, pemesanan, maintenance
3. Menukarkan poin dengan reward
4. Melihat riwayat transaksi reward

Actions:

- Pengguna membuka halaman Rewards
- Pengguna melihat poin: “320 Points Available”
- Pengguna melihat daftar reward: potongan harga, voucher layanan
- Pengguna memilih reward → “Tukar 300 poin untuk diskon Rp 20.000”
- Sistem menampilkan kode voucher

Operations:

- Sistem menambahkan poin setiap transaksi berhasil
- Database memvalidasi poin & reward
- Sistem membuat & mengirimkan voucher digital
- Riwayat reward disimpan ke modul log aktivitas

6. Activity: Pembayaran & Checkout

“Melakukan pembayaran setelah servis selesai.”

Tasks:

1. Melihat total biaya
2. Memilih metode pembayaran
3. Melakukan pembayaran

4. Mendapatkan invoice

Actions:

- Pengguna membuka halaman invoice
- Memilih metode pembayaran: e-wallet, QRIS, kartu
- Melakukan pembayaran
- Men-download invoice

Operations:

- Sistem menghitung biaya
- Backend memproses pembayaran via Payment Gateway
- Status pembayaran diverifikasi
- Invoice disimpan & dikirim via email

7. Activity: Rating & Feedback

“Pengguna menilai pengalaman setelah servis.”

Tasks:

1. Memberikan rating mekanik
2. Memberikan ulasan
3. Mengirim feedback ke aplikasi

Actions:

- Pengguna membuka halaman “Ulasan”
- Memberikan rating bintang 5
- Menulis ulasan: “Mekanik cepat dan ramah”
- Menekan “Kirim”

Operations:

- Sistem menyimpan rating ke database
- Review ditautkan ke profil mekanik
- Rating dihitung untuk ranking mekanik