

Testing Model

1. Unit Testing (Preprocessing & Model)

- **Valid Input**

- Masukkan contoh data normal:

{ "Age": 30, "Weight": 70, "SMOKE": "no", "MTRANS": "Walking", ... }

Pastikan model mengembalikan label yang masuk akal (misal “Normal Weight”).

- **Invalid Input**

- Coba Age = -5 atau Weight = 500

- Coba SMOKE = "maybe" atau MTRANS = "Horse"

Model/API harus menolak (HTTP 422) atau memberikan error terkontrol.

- **Missing Values**

- Hapus FCVC atau MTRANS dari payload

- Pipeline dengan SimpleImputer harus mengisi dengan median/mode, bukan crash.

2. Integration Testing (API Endpoint)

- Kirim request JSON lengkap ke /predict dan verifikasi:

- Status code 200

- Respons JSON berisi kunci "prediction"

- Kirim JSON rusak (typo nama field, format string pada angka) dan cek status code 422.

3. End-to-End Testing (Frontend→Backend)

- Jalankan Streamlit, isi form, klik “Predict”

- Bandingkan hasil di UI dengan panggilan manual menggunakan curl atau Postman.

- **Manipulasi Fitur** (contoh):

- **Age:** 1 dan 100

- **Weight:** 20 dan 200

- **SMOKE:** “yes” vs “no”

- **MTRANS:** “Walking” → “Automobile”

Amati perubahan prediksi—harus konsisten dengan logika model.

4. Load & Stress Testing

- Simulasikan ratusan hingga ribuan request per detik menggunakan tool seperti Locust atau JMeter.

- Catat latency, throughput, dan error rate. Pastikan API tetap responsif.

5. Security Testing

- Coba injeksi SQL di field string, misal

{ "SMOKE": "'; DROP TABLE users;--", ... }

Input harus divalidasi/di-sanitize dan ditolak, tanpa mengungkapkan detail server.

6. Monitoring & Alerting

- Log setiap input & output prediksi untuk audit.
- Setup alert jika:
 - Error rate > 5%
 - Latency request > 200 ms rata-rata
- Gunakan dashboard (Grafana, Kibana) untuk visualisasi metrik.

Model Drift vs Data Drift

1. Data Drift

Distribusi input berubah dari kondisi pelatihan:

- *Contoh:* Awalnya 70 % pasien MTRANS = “Motorbike”, kemudian kampanye sepeda menaikkan proporsi “Bike” menjadi 50 %.
- *Akibat:* Model belum pernah “melihat” distribusi baru ini → performa prediksi menurun.

2. Model Drift

Hubungan fitur → target berubah meski distribusi input sama:

- *Contoh:* SMOKE dulu berkorelasi dengan obesitas I karena perokok cenderung gaya hidup sedentari; setelah program berhenti merokok, korelasi itu hilang.
- *Akibat:* Model menempatkan bobot berlebihan pada SMOKE, sehingga prediksi menjadi keliru.

Langkah Pencegahan:

- **Monitoring:** Pantau metrik (akurasi, distribusi fitur) per minggu/bulan.
- **Deteksi Otomatis:** Gunakan Evidently atau Alibi Detect untuk drift detection.
- **Retraining Berkala:** Jadwalkan retrain setiap kuartal atau saat drift terdeteksi.
- **Continuous Learning:** Integrasikan data baru dan validasi silang (cross-validation) terus-menerus.