Huawei-Technical-Test-I-Documentation\_Fairuz Azka Maulana

Fairuz Azka Maulana

Date: September 2025

# 1. Overview

Dokumen ini berisi dokumentasi setup dan konfigurasi untuk Huawei Technical Test I. Tugas dibagi menjadi tiga bagian utama:  
1. Backend API (Node.js/Express)  
2. Automation (Cron Job & Data Cleansing)  
3. SQL Data Processing  
  
Setiap bagian akan dijelaskan dengan detail mengenai tools yang digunakan, fungsi utama, langkah instalasi, cara menjalankan, serta metode pengujian.

# 2. Backend API

Tools yang digunakan: Node.js, Express

Fungsi API:

- POST /form → menyimpan data dari frontend ke dalam array di server.  
- GET /form → mengambil semua data yang tersimpan.

## Langkah Setup:

1. Pastikan Node.js dan npm sudah terinstall di sistem.  
2. Masuk ke folder `backend`.  
3. Jalankan perintah berikut:  
 cd backend  
 npm install  
 node server.js  
4. Server akan berjalan pada http://localhost:3000.

## Pengujian API:

Pengujian dilakukan menggunakan Postman:  
- POST Request: kirim data JSON ke endpoint /form, data akan disimpan ke array.  
- GET Request: akses endpoint /form, server akan mengembalikan semua data dalam bentuk JSON array.

# 3. Automation (Cron Job & Data Cleansing)

Tools yang digunakan: Python 3, Cron (Linux/Unix)

## Deskripsi Script:

- collect.py → Mengumpulkan data simulasi setiap 08:00, 12:00, dan 15:00 WIB, disimpan ke dalam folder /home/cron/ dengan format cron\_{tanggal}\_{jam}.csv.  
- cleanup.py → Menghapus file di folder /home/cron/ yang berumur lebih dari 30 hari.

## Langkah Setup:

1. Pastikan Python 3 sudah terinstall.  
2. Simpan script collect.py dan cleanup.py di folder automation.  
3. Buat folder /home/cron jika belum ada.  
4. Ubah hak akses folder agar script bisa menulis file.

## Konfigurasi Cron:

Tambahkan konfigurasi berikut di crontab (crontab -e):  
  
0 8,12,15 \* \* \* /usr/bin/python3 /path/to/automation/collect.py  
0 0 \* \* \* /usr/bin/python3 /path/to/automation/cleanup.py  
  
Baris pertama: menjalankan collect.py pada jam 08:00, 12:00, dan 15:00.  
Baris kedua: menjalankan cleanup.py setiap tengah malam.

# 4. SQL Data Processing

Tools yang digunakan: MySQL / PostgreSQL

## Struktur Tabel:

Tabel employee memiliki kolom:  
- Name  
- Position  
- JoinDate  
- ReleaseDate  
- YearOfExperience  
- Salary

## Query yang dijalankan:

1. Insert data Albert sebagai Engineer:  
 INSERT INTO employee (Name, Position, JoinDate, YearOfExperience, Salary)  
 VALUES ('Albert', 'Engineer', '2024-01-24', 2.5, 50);  
  
2. Update salary Engineer menjadi 85:  
 UPDATE employee SET Salary = 85 WHERE Position = 'Engineer';  
  
3. Hitung total salary tahun 2021:  
 SELECT SUM(Salary) AS total\_salary\_2021 FROM employee  
 WHERE YEAR(JoinDate) <= 2021 AND (ReleaseDate IS NULL OR YEAR(ReleaseDate) >= 2021);  
  
4. Menampilkan 3 karyawan dengan Years of Experience terbanyak:  
 SELECT Name, Position, YearOfExperience FROM employee  
 ORDER BY YearOfExperience DESC LIMIT 3;  
  
5. Subquery Engineer dengan pengalaman ≤ 3 tahun:  
 SELECT \* FROM employee WHERE Position = 'Engineer' AND YearOfExperience <= 3;

# 5. Conclusion

Semua bagian dari Huawei Technical Test I telah berhasil diimplementasikan:  
- Backend API berjalan dengan baik menggunakan Node.js dan Express.  
- Automation script collect.py dan cleanup.py dapat dijalankan secara manual maupun otomatis dengan cron.  
- Query SQL menghasilkan output sesuai dengan requirement.