

## BAB 3

### PELAKSANAAN KERJA MAGANG

#### 3.1 Kedudukan dan Organisasi

Kedudukan pada pelaksanaan kerja magang MBKM di PT Tangara Mitrakom adalah sebagai *Programmer* yang berada di bawah naungan departemen *product development*. Pada posisi ini, mahasiswa dituntut untuk kreatif, kritis, dan memiliki pemahaman HTML, CSS, JavaScript dan PHP serta kemampuan *problem solving* dalam melakukan pembuatan dan pengembangan website.

Koordinasi dalam melaksanakan tugas kegiatan magang yang dibimbing oleh Bapak Ahmad Dadan Hudaya selaku karyawan tetap divisi *programmer* pada PT. Tangara Mitrakom. Bentuk dari koordinasi dan bimbingan itu adalah pemberian pemahaman mengenai *programmer* pada dunia kerja serta kritik dan saran atas tugas yang dilakukan selama pelaksanaan kerja magang MBKM. Tim *Programmer Intern* terdiri dari 2 orang dan satu supervisor yang bekerja secara berkolaborasi dan saling memenuhi tugas yang diberikan oleh supervisor.

Pada masa praktik kerja magang berlangsung, kegiatan magang ditunjang dengan perangkat lunak seperti Figma untuk melakukan desain *user interface* agar lebih terlihat menarik dan mudah dipahami oleh *end-user*, dan Visual Studio Code untuk menulis codingan dan penyesuaian detail pada HTML, CSS, JS, dan PHP pada hasil implementasi website oleh tim *front end developer* agar lebih sesuai dengan desain yang dilakukan di Figma.

#### 3.2 Tugas yang Dilakukan

Selama pelaksanaan kerja magang MBKM di PT. Tangara Mitrakom, tugas yang dilakukan adalah membuat sebuah *dashboard* aplikasi untuk melakukan monitoring dan analisis perangkat IoT. Pembuatan aplikasi ini dibuat berdasarkan kebutuhan PD (*Product Development*).

Selama pengerjaan tugas, terbagi menjadi beberapa proses yaitu *database management system*, *server control and settings*, *user interface design*, *front-end development*, dan *back-end development*. Setiap proses pembuatan aplikasi dilakukam secara bertahap dan di kerjakan berdasarkan kesepakatan antar peserta magang dengan supervisi.

### 3.3 Uraian Pelaksanaan Magang

Uraian pelaksanaan magang MBKM sebagai *Programmer Intern* di PT. Tangara Mitrakom terbagi menjadi perencanaan, pelaksanaan, peninjauan, perbaikan, dan penelitian. Pelaksanaan kerja magang diuraikan seperti pada Tabel 3.1.

Tabel 3.1. Timeline Kegiatan Magang

Minggu Ke -	Pekerjaan yang dilakukan
1	Melakukan briefing bersama tim <i>product development</i> terkait aplikasi yang akan di <i>develop</i> .
2 - 3	Mempelajari dan memahami struktur basis data yang dibutuhkan dalam pembuatan aplikasi.
4 - 6	Melakukan konfigurasi file dari server ke lokal (atau sebaliknya).
7 - 8	Melakukan design UI ( <i>User Interface</i> ) pada aplikasi Figma
9 - 10	Membuat tampilan UI website berdasarkan design pada Figma.
11 - 13	Menghubungkan <i>back-end</i> website dengan tampilan UI.
14 - 15	Melakukan <i>quality assurance</i> dan <i>quality control</i> pada website yang telah dibuat.

Pada minggu pertama dari kegiatan magang MBKM di PT Tangara Mitrakom, peserta magang melakukan briefing dengan departemen *workshop*, departemen *operational*, dan departemen *Product Development and IT* terkait IoT (*Internet of Things*). Setelah itu, peserta magang akan berdiskusi dengan supervisor untuk tugas yang akan dikerjakan.

Pada minggu kedua dan ketiga, peserta magang diberikan sebuah file PDF yang berisi rincian terkait aplikasi yang harus dibuat dan tautan untuk mengakses database pada server PT. Tangara Mitrakom. Peserta magang diperbolehkan untuk melakukan manajemen terkait struktur database pada server tersebut.

Pada minggu keempat hingga keenam, peserta magang di minta untuk mengunduh sebuah software bernama WinSCP untuk melakukan konfigurasi file project pada server ke dalam file lokal. Sistem operasi yang digunakan dalam server adalah CentOS, sehingga konfigurasi dilakukan menggunakan sebuah terminal pada WinSCP.

Pada minggu ketujuh dan kedelapan, melakukan pembuatan dan revisi terkait design UI sederhana pada aplikasi Figma. Design ini akan menjadi acuan

dalam pengembangan website dari sisi *front-end*.

Pada minggu kesembilan dan kesepuluh, memulai pembuatan website dari sisi *front-end* berdasarkan design yang telah dibuat melalui aplikasi Figma pada minggu ketujuh dan kedepalan. Dalam pembuatan website ini menggunakan bahasa pemrograman seperti HTML, CSS dan Javascript.

Pada minggu kesebelas hingga ketigabelas, memulai pembuata website dari sisi *back-end* dan juga menghubungkan proses *back-end* dengan *front-end* sehingga website dapat berjalan sesuai kebutuhan dan kriteria yang telah didiskusikan di awal.

Pada minggu keempatbelas dan kelimabelas, difokuskan untuk melakukan *quality assurance* dan *quality control* pada hasil pembuatan website dari kedua sisi *front-end* dan *back-end*. Dilakukannya hal ini agar hasil pembuatan website dapat berfungsi secara keseluruhan dan sesuai dengan kriteria yang telah ditentukan pada awal perancangan.

### **3.3.1 Persiapan**

Tahap awal dari kegiatan magang MBKM sebagai *internship programmer* di PT. Tangara Mitrakom adalah persiapan. Pada tahap ini, supervisor mendapatkan informasi mengenai permintaan untuk membuat sebuah aplikasi, yang kemudian disampaikan kepada para peserta magang. Pemberian informasi mengenai aplikasi ini dilakukan secara langsung.

Pada tahap persiapan ini, dilakukan diskusi kepada para peserta magang dalam aplikasi yang akan dibuatkan, juga fitur apa saja yang akan mejadi kriteria utama hingga mengatur batas waktu dalam pembuatan aplikasi tersebut. Dalam diskusi, para peserta magang diberikan kesempatan dalam berpendapat terkait aplikasi yang akan dibuat.

Supervisor diberikan informasi oleh *product manager* bahwa aplikasi ini sangat dibutuhkan oleh perusahaan dalam melakukan monitoring dan analisis perangkat IoT (*Internet of Things*).

### **3.3.2 Perancangan**

Setelah tahap persiapan dilakukan, dilanjutkan dengan tahap perancangan yang mana pada tahap ini peserta magang MBKM melakukan proses pengerjaan atas aplikasi yang telah diberikan dan direncanakan sebelumnya.

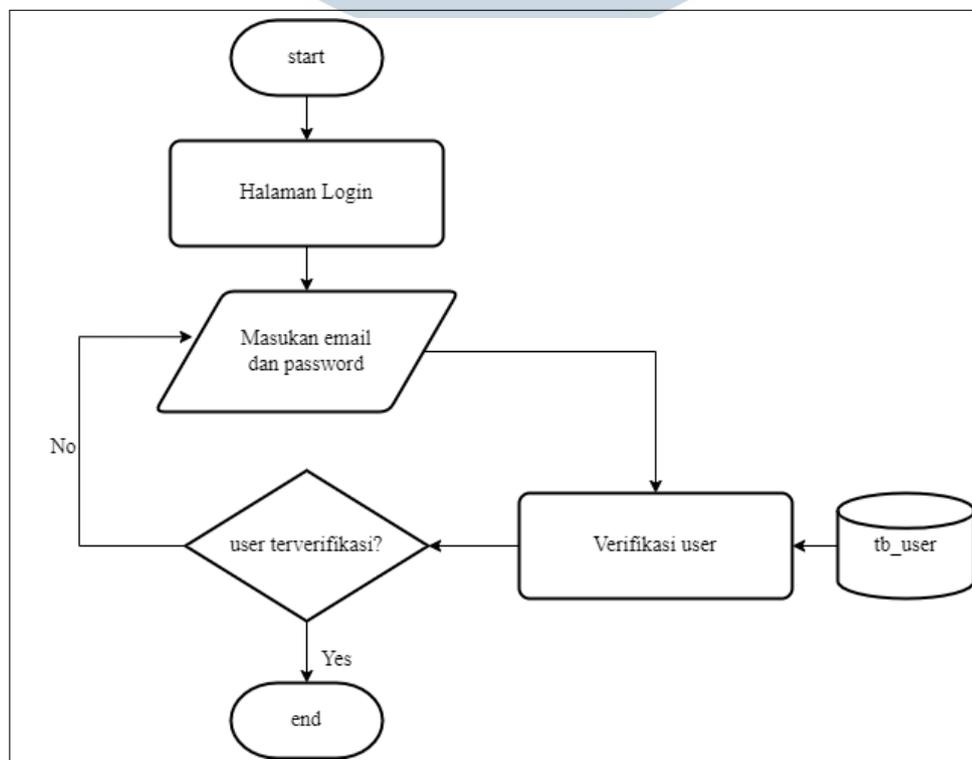
## A. Flowchart Website Sistem Manajemen Kehadiran

Flowchart atau bagan alur adalah diagram yang menampilkan langkah-langkah dan keputusan untuk melakukan sebuah proses dari suatu program. Setiap langkah digambarkan dalam bentuk diagram dan dihubungkan dengan garis atau arah panah [6].

Dalam melakukan perancangan suatu perangkat lunak atau website, terdapat beberapa alur yang harus diperhatikan. Alur atau langkah tersebut dibuat dalam suatu *flowchart* yang mana merupakan sebuah diagram yang menggambarkan alur atau algoritma yang ada pada suatu sistem [7]. Berikut *flowchart* dari aplikasi *dashboard* IoT PT. Tangara Mitrakom yang telah di rancang.

### A.1 Flowchart Login User

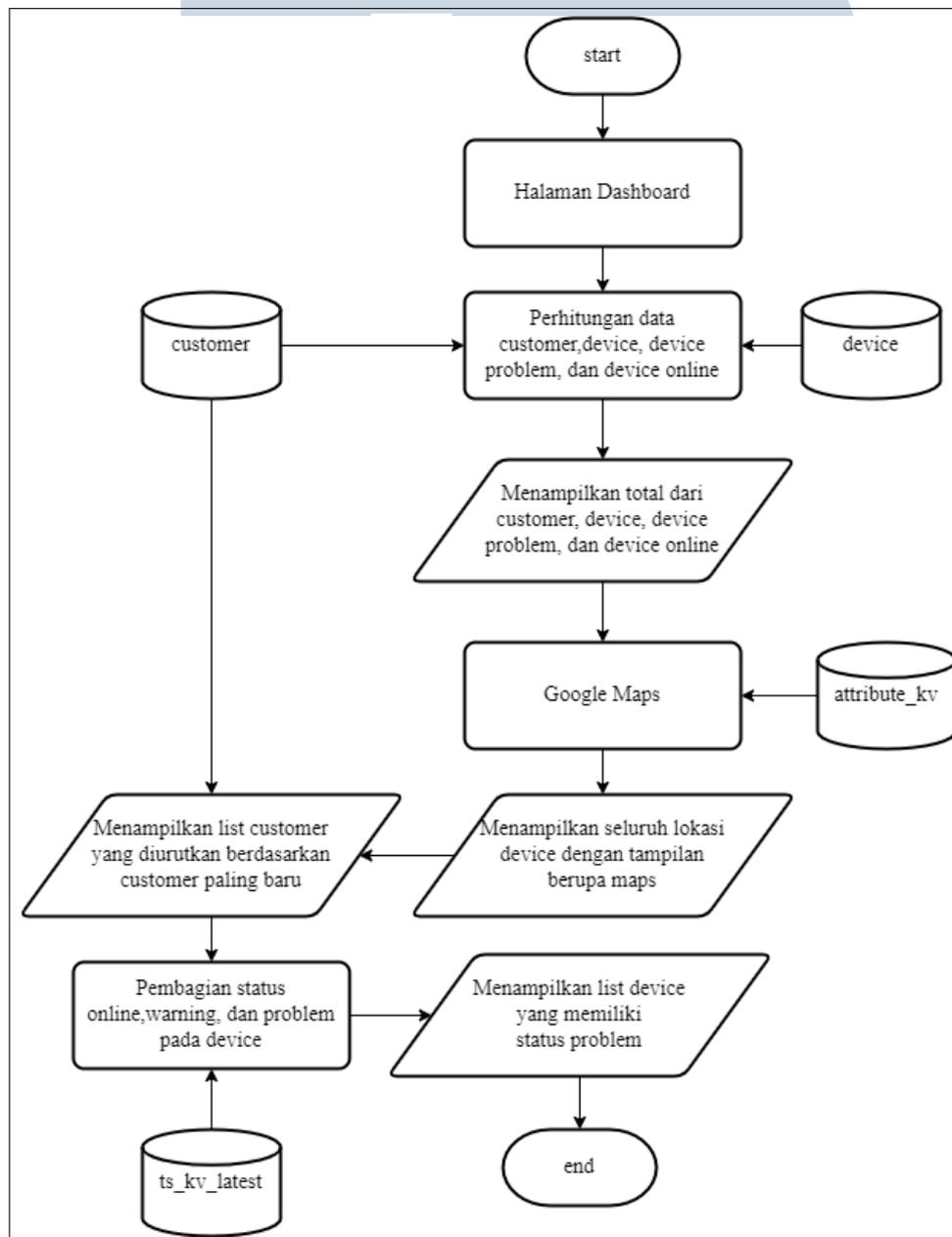
Halaman login merupakan halaman awal dalam aplikasi sistem manajemen kehadiran. Pada Gambar 3.1 terdapat alur dalam proses verifikasi data *user*. Verifikasi data *user* dilakukan berdasarkan *database* pada perusahaan.



Gambar 3.1. Flowchart Halaman Login

## A.2 Flowchart Halaman Dashboard

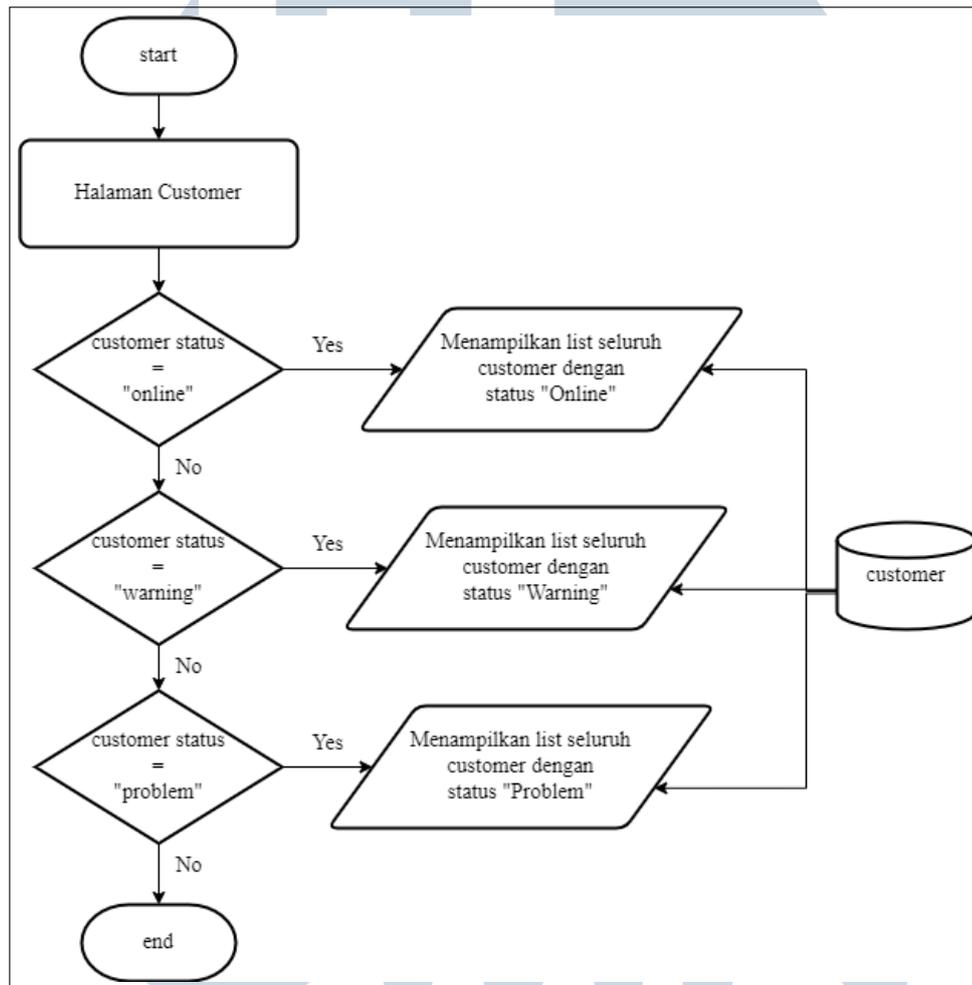
Halaman dashboard merupakan halaman selanjutnya setelah *user* berhasil melalui verifikasi. Pada Gambar 3.2 merupakan *flowchart* untuk halaman dashboard. Pada halaman ini *user* dapat melihat jumlah dari *customer*, *device*, *device problem* dan *device online*. Pada halaman ini juga *user* dapat melihat lokasi dari perangkat IoT yang terinstalasi.



Gambar 3.2. Flowchart Halaman Dashboard

### A.3 Flowchart Halaman Customer

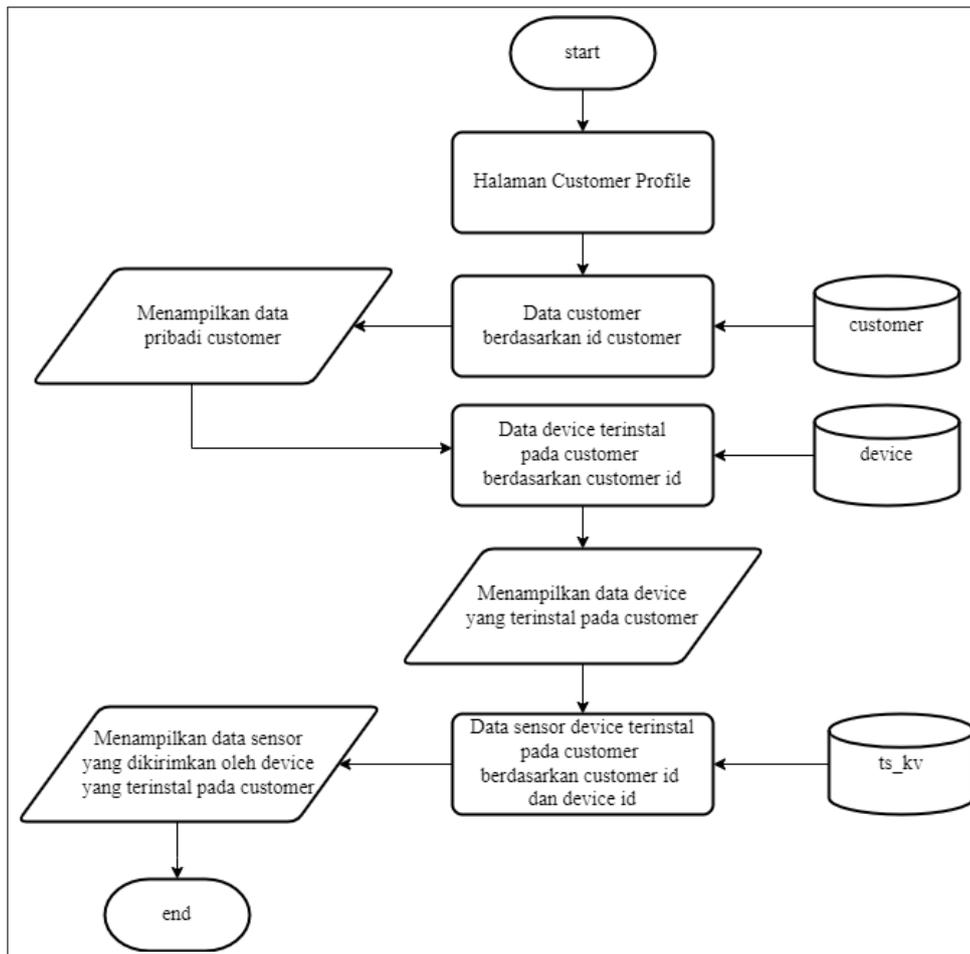
Pada Gambar 3.3 merupakan *flowchart* untuk halaman customer. Pada halaman ini *user* dapat melihat list dari *customer* berdasarkan status. Status terdiri dari 3, yaitu : *online*, *warning*, *problem*.



Gambar 3.3. *Flowchart* Halaman Customer

### A.4 Flowchart Halaman Customer Profile

Pada Gambar 3.4 merupakan *flowchart* untuk halaman customer profile. Pada halaman ini *user* dapat melihat keterangan atau detail dari *customer* dan perangkat IoT yang telah terpasang pada *customer*.

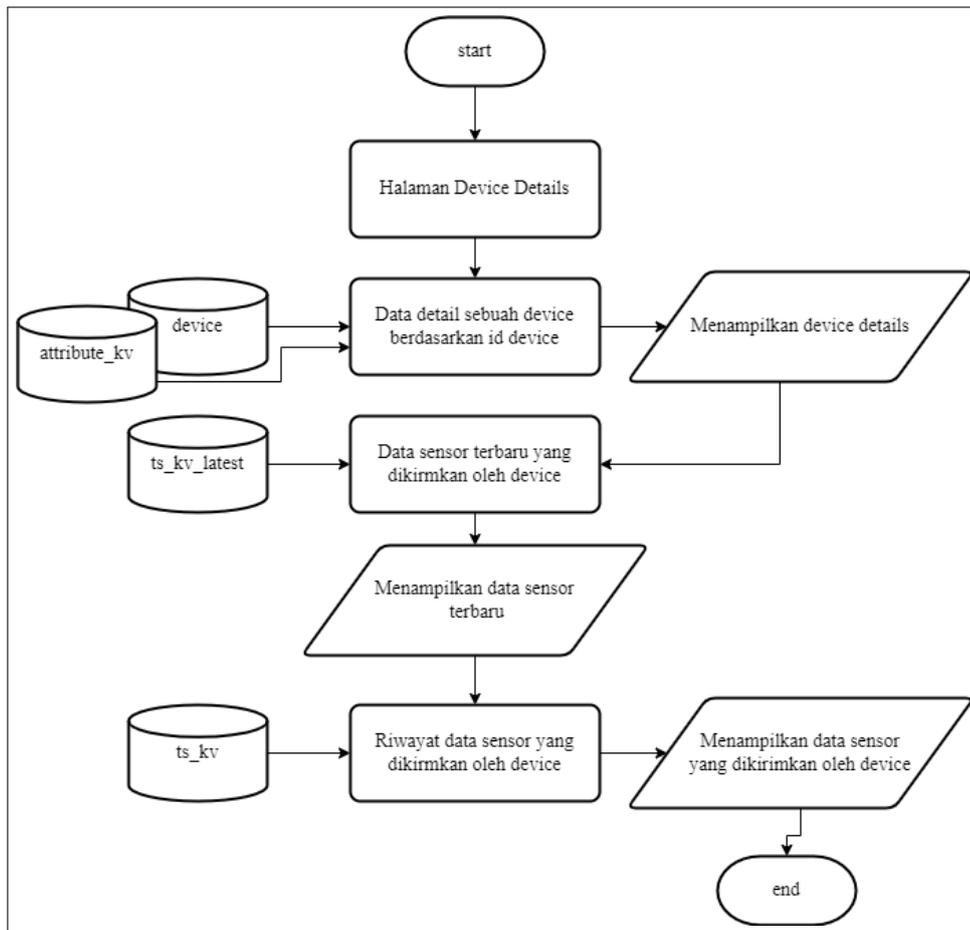


Gambar 3.4. *Flowchart* Halaman Customer

### A.5 Flowchart Halaman Device Details

Pada Gambar 3.5 merupakan *flowchart* untuk halaman *device details*. Pada halaman ini *user* dapat melihat keterangan atau detail dari perangkat IoT. Pada halaman ini juga, *customer* dapat melihat data sensor yang dikirimkan pada server.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



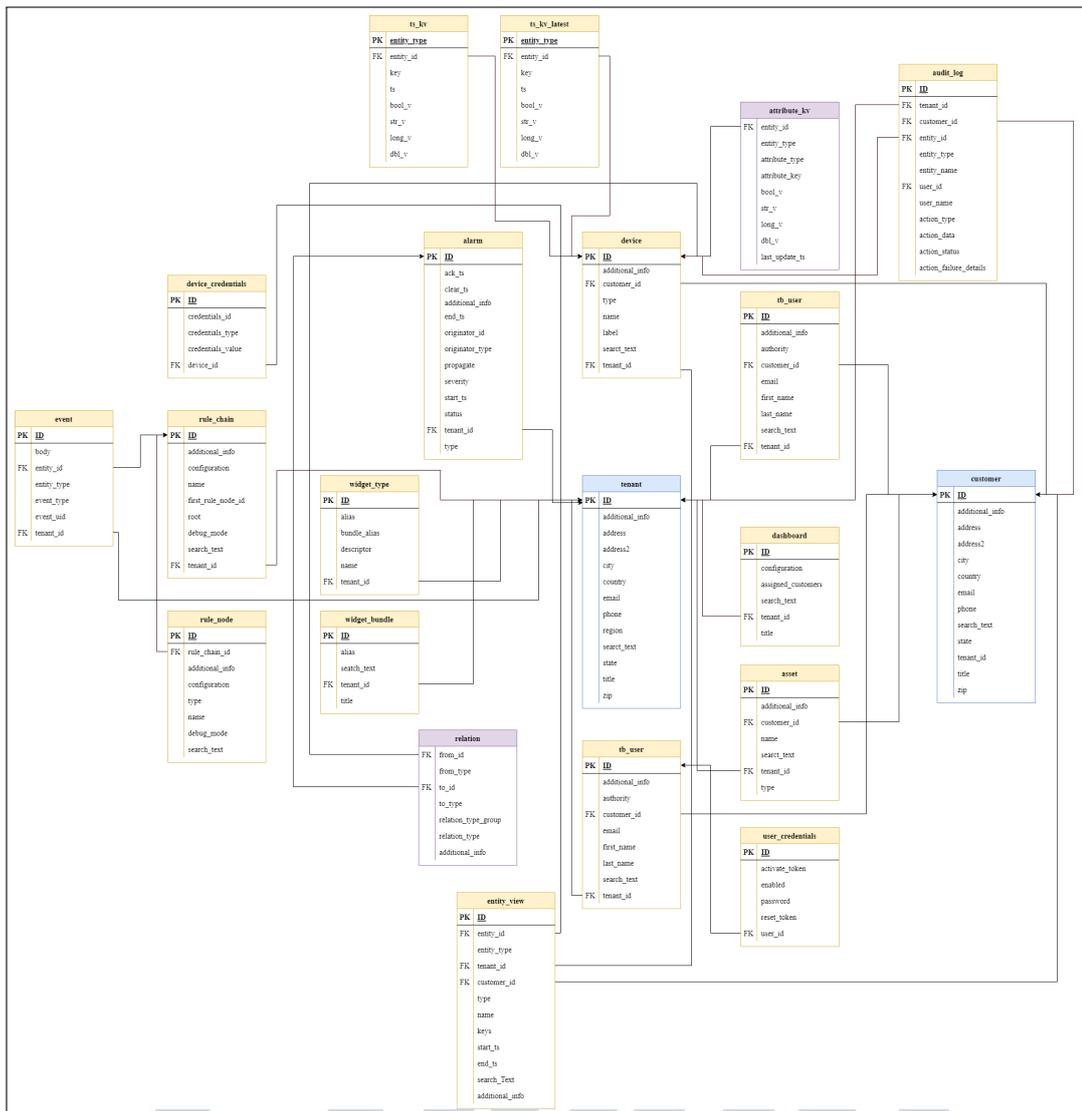
Gambar 3.5. Flowchart Halaman Device Details

## B. Pembuatan Database Schema

Skema database dianggap sebagai cetak biru database yang menjelaskan bagaimana data dapat berhubungan dengan tabel lain atau model data lainnya [8]. Pembuatan skema database menggunakan website online bernama draw.io.

Pada aplikasi monitoring dan analisis perangkat IoT terdapat dua puluh dua tabel, yaitu tabel tenant, customer, dashboard, asset, tb\_user, user\_credentials, widget\_type, widget\_bundle, rule\_chain, rule\_node, event, admin\_settings, component\_descriptor, device, ts\_kv, ts\_kv\_latest, device\_credentials, attribute\_kv, alarm, entity\_view, relation, dan audit\_logs.

Data pada tabel tb\_user digunakan untuk menyimpan informasi yang dimasukkan *user* pada halaman *login*. Pada Gambar 3.6 merupakan skema database yang terdiri dari relasi antar tabel yang telah dibuat.



Gambar 3.6. Database Schema Monitoring dan Analisis Perangkat IoT

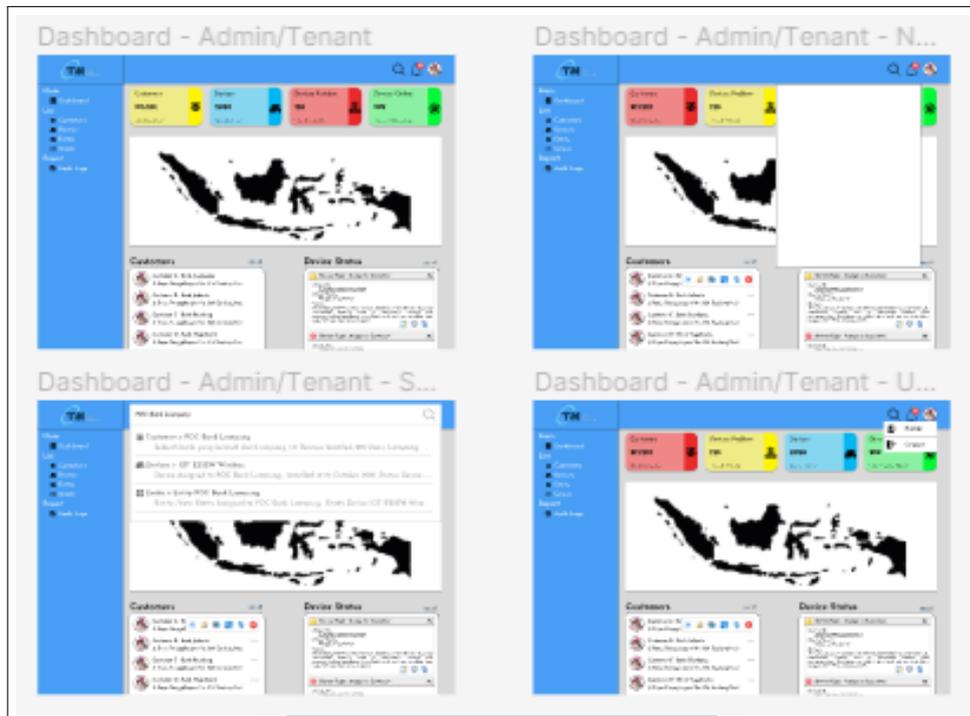
### C. User Interface Design Aplikasi Monitoring dan Analisis Perangkat IoT

Perancangan *user interface* sangatlah penting dalam sebuah pembangunan website, hal ini dikarenakan dengan desain yang baik yang mana memenuhi kebutuhan pengguna dapat membuat pengguna merasa nyaman saat mengunjungi sebuah website [9].

Perancangan *user interface* sangat membantu dalam proses pembuatan website dari sisi *front-end*, sehingga implementasi sebuah design kedalam codingan akan jauh lebih mudah dengan adanya design *user interface* yang telah dibuat.

Berikut ini adalah *design user interface* dari halaman monitoring dan

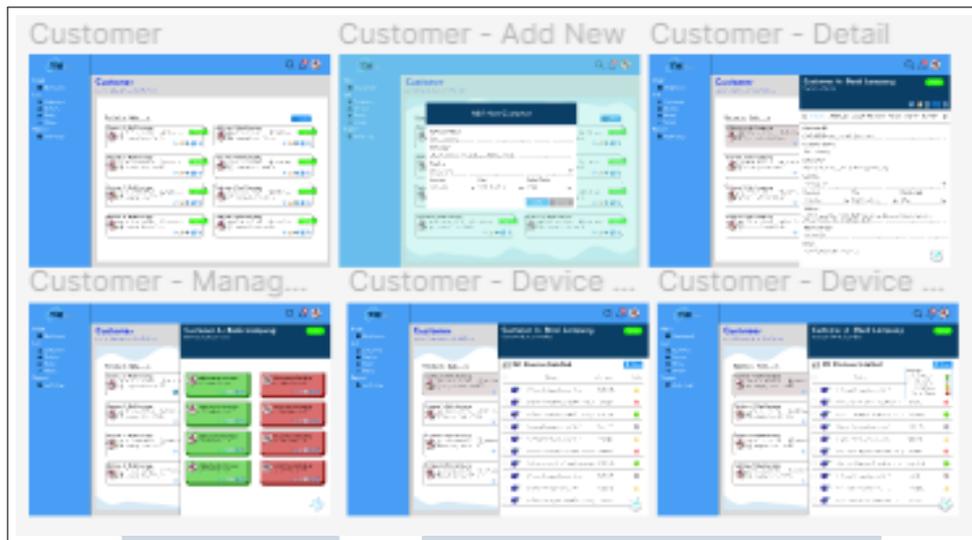
analisis perangkat IoT pada PT. Tangara Mitrakom.



Gambar 3.7. User Interface Halaman Dashboard

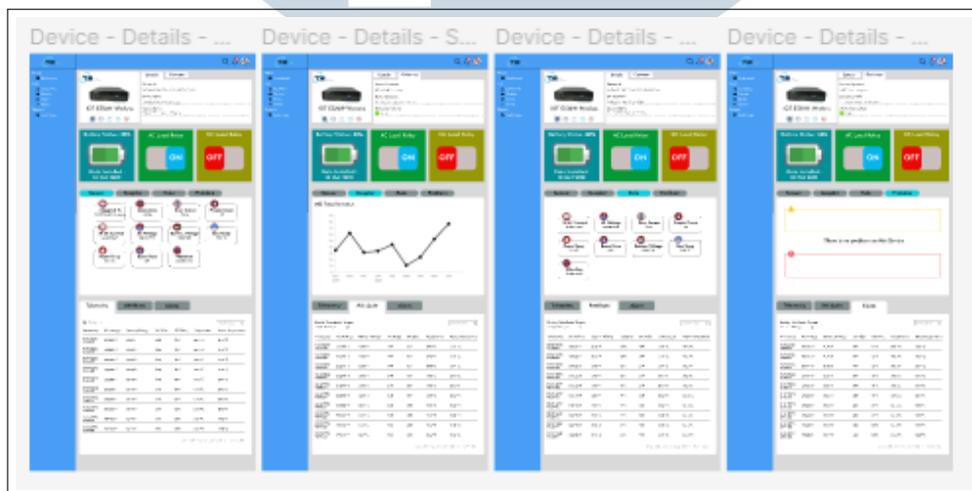
Gambar 3.7 merupakan halaman dimana *user* dapat mengetahui informasi terkait jumlah dari *customer*, *device*, *device problem* dan *device online*. Pada halaman ini juga terdapat sebuah *maps* sehingga *user* dapat mengetahui lokasi dari perangkat IoT. Pada bagian terakhir halaman *user* dapat mengetahui status perangkat yang bermasalah dan list *customer*.

U M N  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



Gambar 3.8. *User Interface* Halaman Customer

Gambar 3.9 merupakan halaman dimana *user* dapat mengetahui informasi terkait status, user, perangkat IoT yang terinstal, hingga detail dari setiap *customer*.



Gambar 3.9. *User Interface* Halaman Device Details

Gambar 3.8 merupakan halaman dimana *user* dapat mengetahui informasi terkait perangkat IoT yang terkoneksi. Pada halaman ini terdapat sebuah data mengenai nilai sensor yang dikirimkan perangkat IoT pada server.

### 3.3.3 Peninjauan

#### A. Peninjauan hasil kerja oleh tim product development

Setelah proses pengerjaan selesai dilakukan sesuai dengan batas waktu, selanjutnya terdapat proses peninjauan di mana pada proses ini, tim *product department* mengulas hasil pekerjaan yang telah dilakukan oleh peserta magang.

Proses peninjauan ini para peserta magang diminta untuk mempresentasikan hasil pengerjaan yang telah dibuat kepada tim *product department* secara langsung.

Pada tahap ini, tim *product department* dan peserta magang MBKM saling memberikan masukan dan arahan hingga tercapailah sebuah kesimpulan/solusi atas permasalahan yang akan menuntun pada tahap selanjutnya yaitu perbaikan untuk mencapai hasil yang maksimal.

#### B. Perbaikan website monitoring dan analisis perangkat IoT berdasarkan peninjauan tim product development

Pada proses ini para peserta magang melakukan perombakan pada hasil kerja sebelumnya baik dalam skala kecil maupun skala besar. Kemudian, setelah berbagai proses perbaikan telah dipenuhi sesuai dengan masukan dan arahan yang didapatkan, para peserta magang dapat mengkonfirmasi hasil akhir pengerjaan dengan mengkomunikasikannya bersama tim *product development*. Lalu setelah mendapatkan kesepakatan mengenai ketetapan hasil akhir. Peserta magang kembali mengerjakan perbaikan yang telah disepakati bersama baik dari sisi *design* hingga pengembangan website.

#### C. QA/QC pada aplikasi monitoring dan analisis perangkat IoT PT. Tangara Mitrakom

Selanjutnya dilakukan QA/QC pada website hasil implementasi terakhir. *Quality assurance* adalah proses yang dilakukan untuk memutuskan apakah suatu *software* ataupun *website* memenuhi syarat yang ditentukan dan dilakukan secara sistematis. Sedangkan, *quality control* adalah tindakan pengecekan tiap adanya kekurangan pada hasil implementasi *software* atau website [10],

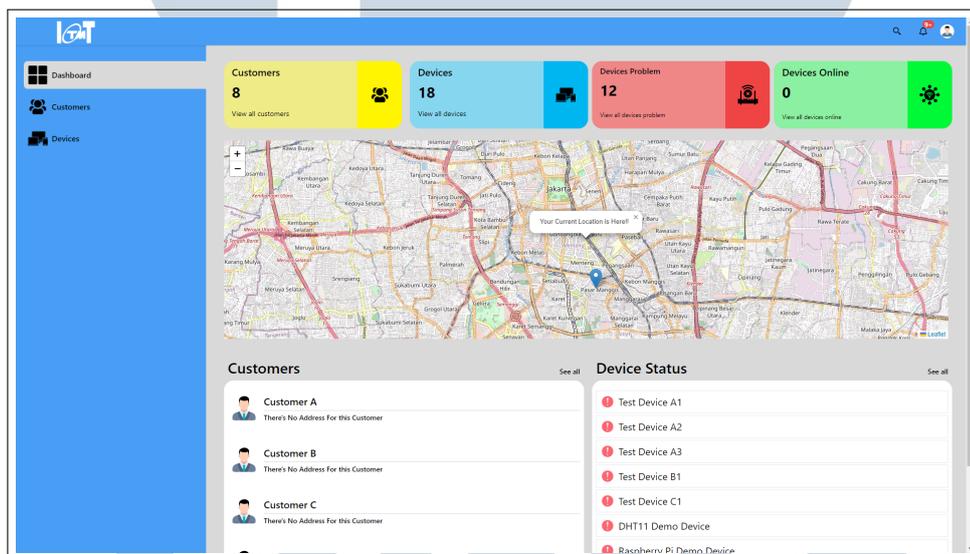
Tujuan dari *quality assurance* dan *quality control* dalam pengembangan *software* atau *website* adalah untuk memastikan bahwa implementasi yang dihasilkan memenuhi spesifikasi dan persyaratan kualitas [11].

Saat melakukan proses peninjauan terhadap hasil kerja dari peserta magang. Hanya terdapat sedikit ketidak sesuaian seperti bentuk tulisan yang kurang sesuai pada design, kebutuhan data table pada setiap halaman, *responsive* dari beberapa halaman, hingga penambahan halaman untuk *customer profile*.

### 3.3.4 Hasil Implementasi

Pada hasil implementasi aplikasi monitoring dan analisis perangkat IoT PT. Tangara Mitrakom menggunakan sebuah bahasa pemrograman PHP dan PostgreSQL, juga menggunakan bahasa pemrograman HTML, CSS, dan Javascript untuk pengembangan website dari sisi *front-end*.

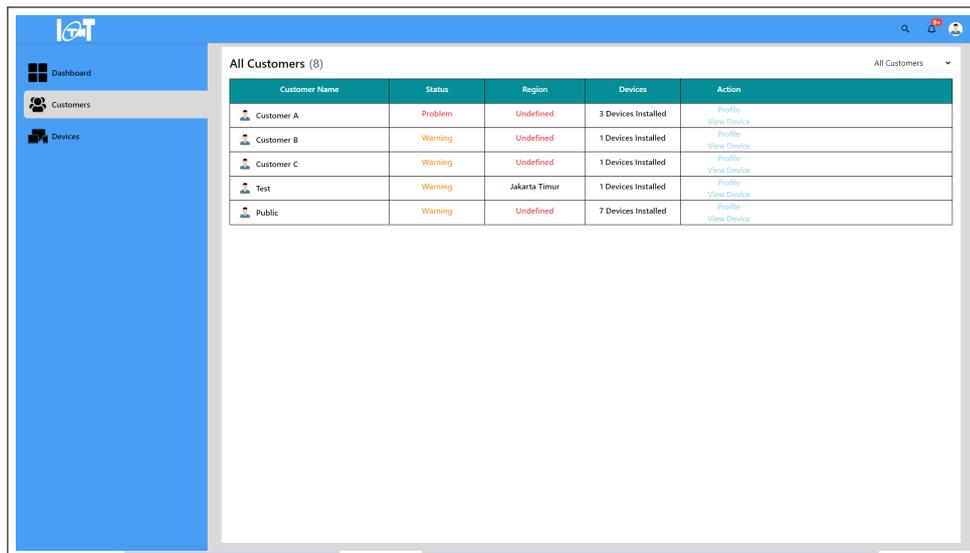
Berikut ini adalah hasil implementasi dari sistem manajemen absensi PT. Tangara Mitrakom.



Gambar 3.10. Halaman Dashboard

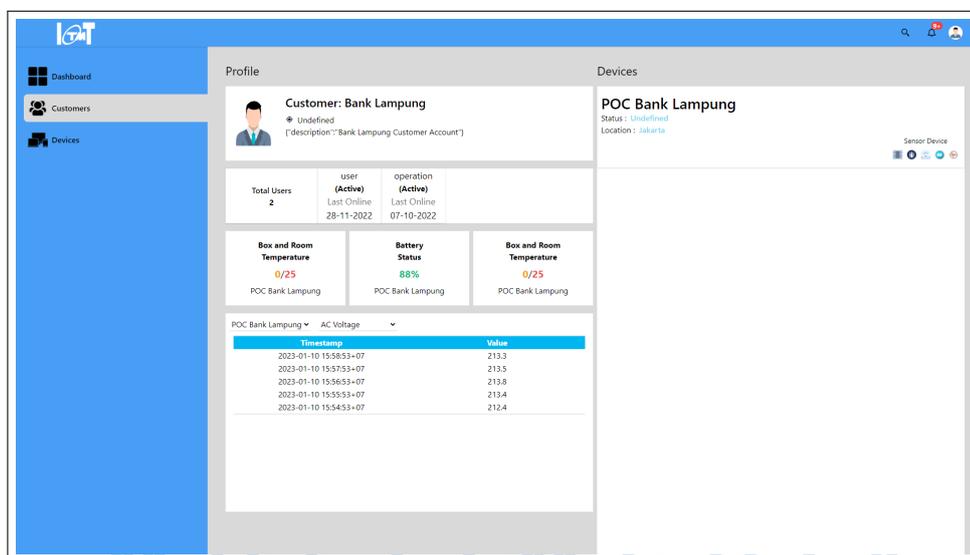
Gambar 3.10 merupakan hasil implementasi pada halaman *customer*. Konsep kinerja *dashboard* merupakan suatu model aplikasi sistem informasi yang disediakan bagi para manajer untuk menyajikan informasi kualitas kinerja, dari suatu perusahaan atau lembaga organisasi, dashboard telah banyak diadopsi oleh perusahaan atau bisnis [12].

Pada halaman *dashboard* terdiri dari jumlah *customer*, *device*, *device online* dan *device problem*, sebuah map untuk mengetahui lokasi dari perangkat IoT, dan list dari *customer* dan *device status*.



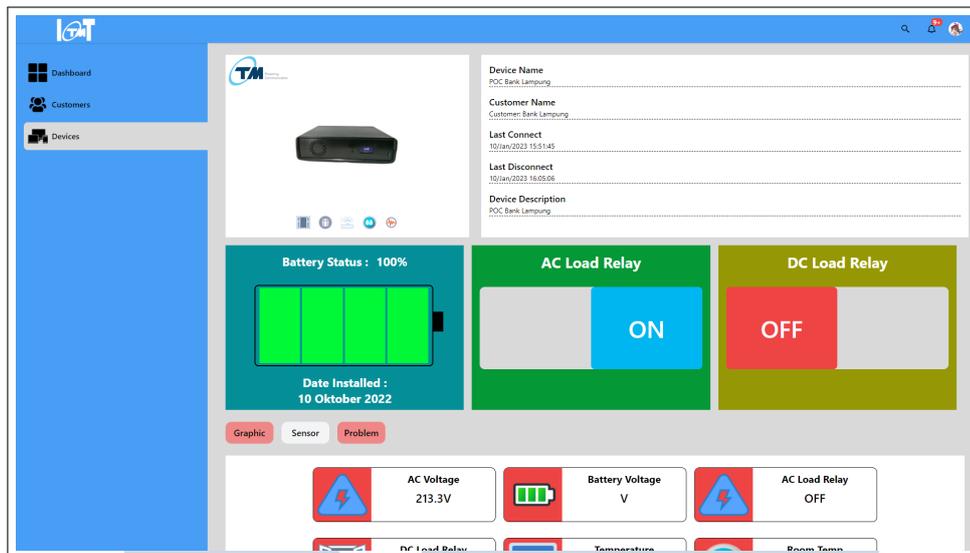
Gambar 3.11. Halaman Customer

Gambar 3.11 merupakan hasil implementasi halaman *customer*. Pada halaman ini *user* dapat melihat list dari *customer*. Pada halaman ini *user* dapat mengetahui status dari *customer* tersebut, *user* juga dapat melihat jumlah perangkat IoT yang telah terinstal pada setiap *customer*.



Gambar 3.12. Halaman Customer Profile

Gambar 3.12 merupakan hasil implementasi halaman *customer profile*. Pada halaman ini *user* dapat melihat detail tentang setiap *customer*. Pada halaman ini *user* dapat mengetahui informasi tentang perangkat IoT yang terinstal pada *customer* dan nilai sensor yang dikirimkan oleh perangkat IoT.



Gambar 3.13. *Halaman Device Details*

Gambar 3.13 merupakan hasil implementasi halaman *device details*. Pada halaman ini *user* dapat melihat detail pada perangkat IoT dan nilai sensor yang dikirimkan oleh perangkat IoT.

### 3.4 Kendala dan Solusi yang Ditemukan

#### 3.4.1 Kendala yang Dihadapi

Kendala yang dialami selama melakukan pengembangan aplikasi *monitoring* dan analisis perangkat IoT PT. Tangara Mitrakom adalah adanya serangan sebuah virus Ransomware pada server. Virus ini menyebabkan hilangnya seluruh *database* pada server. Hal ini menjadi kendala utama dalam proses pembuatan aplikasi.

#### 3.4.2 Solusi atas Kendala

Solusi atas kendala yang ditemukan adalah melakukan instalasi ulang terhadap server dan memasang sebuah Anti-Virus sehingga server dapat terhindar dari serangan Ransomware kembali dan melakukan *restore* ulang terhadap *database* yang di lakukan oleh tim *Workshop*.