

BAB 3

PELAKSANAAN KERJA MAGANG

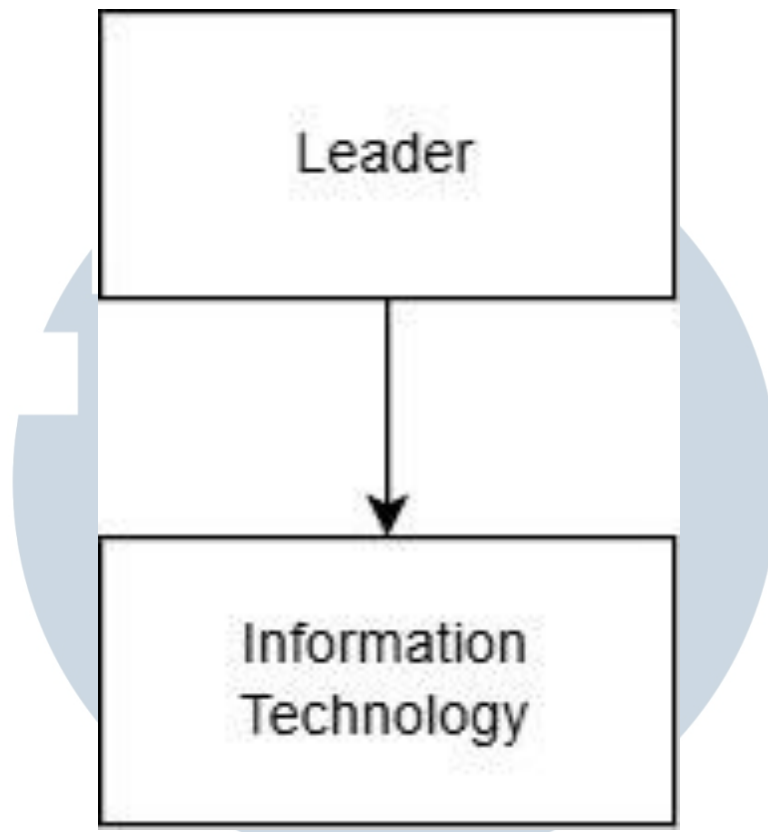
Pelaksanaan Kerja Magang merupakan inti dari kegiatan yang dilakukan penulis selama kurun waktu Enam Bulan di PT. Hanielshen. Pembahasan dalam bab ini difokuskan pada realisasi kerja, tanggung jawab yang diemban, serta implementasi ilmu pengetahuan yang diperoleh di perkuliahan ke dalam praktek kerja nyata.

Penjelasan disusun secara sistematis meliputi gambaran umum pekerjaan, kedudukan praktikan dalam struktur organisasi, hingga rincian tugas harian yang diselesaikan selama periode magang berlangsung. Tujuannya adalah untuk memberikan gambaran yang jelas mengenai kontribusi dan pengalaman teknis yang didapatkan penulis di dunia kerja profesional.

3.1 Kedudukan dan Koordinasi

Intership di PT Hanielsen ditempatkan pada divisi *Information Technology* yang bertugas untuk melakukan proses perancangan dan pengembangan sistem perusahaan, khususnya sistem absensi dan penjadwalan karyawan berbasis MySQL. Dalam proses pelaksanaan magang, penulis bekerja dalam satu tim kecil yang terdiri dari beberapa posisi yang saling berkoordinasi.





Gambar 3.1. Struktur Divisi IT PT Hanielsen

Tim ini berada di bawah pengawasan langsung oleh Bapak Hendi pangudi selaku *leader* dari perusahaan Koordinasi dilakukan secara berkala melalui pertemuan mingguan untuk membahas progres, hambatan, dan rencana pekerjaan berikutnya. Selama kegiatan magang, seluruh proses koordinasi dan pengembangan sistem dilakukan menggunakan beberapa platform kerja seperti *WhatsApp*, *Google Meet* untuk memastikan setiap progres dapat dipantau dengan baik. Dokumentasi teknis serta penyimpanan kode dilakukan melalui *GitHub* untuk kejelasan struktur revisi dan mempermudah kolaborasi antar anggota tim .

3.2 Tugas dan Uraian Kerja Magang

Tugas yang di lakukan pada pelaksanaan magang adalah sebagai berikut dapat di lihat pada table 3.1

Tabel 3.1. Tugas yang di lakukan selama pelaksanaan kerja magang

3

Minggu ke -	Perkerjaan yang di lakukan
1	Perkenalan dengan <i>supervisor</i> dan melakukan penjelasan <i>project</i> yang ingin dilakukan.
2	Melakukan <i>meeting</i> dengan team dan <i>supervisor</i> untuk membahas ERD (<i>Entity Relationship Diagram</i>) awal.
3	Mempelajari dasar dasar <i>back-end</i> , <i>Instalasi</i> dan konfigurasi <i>environment</i> pengembangan <i>back-end</i> (Bun, Node.js, MySQL).
4	Belajar membuat koneksi antara <i>back-end</i> dengan <i>database</i> MySQL dan mempelajari struktur folder <i>back-end</i> .
5	Membuat <i>database</i> sederhana untuk latihan awal dan melakukan <i>meeting</i> dengan <i>supervisor</i> untuk <i>Review</i> dan revisi ERD serta membuat tabel awal di MySQL sesuai hasil ERD yang telah disetujui.
6	Merancang struktur <i>database</i> akhir untuk aplikasi absensi dan penjadwalan serta membuat tabel-tabel utama seperti <i>users</i> , <i>roles</i> , <i>attendance</i> , <i>schedule</i> dan <i>Shift</i> .
7	Membuat API endpoint dasar untuk fitur utama aplikasi CRUD (<i>Create</i> , <i>Read</i> , <i>Update</i> , <i>Delete</i>).
8	Menghubungkan relasi antar tabel di <i>database</i> (<i>foreign key</i> dan <i>constraint</i>).Serta membuat fitur absensi dan penjadwalan menggunakan <i>query SQL</i> .
9	Mengembangkan sebagian <i>back-end</i> untuk versi <i>mobile</i> (struktur serupa dengan <i>desktop</i>). Dan melakukan <i>Review</i> hasil pekerjaan <i>back-end</i> dekstop dan <i>mobile</i> dengan <i>supervisor</i> .
10	Melakukan koreksi dari hasil rapat dengan <i>supervisor</i> dan juga membuat bagian <i>back-end</i> untuk <i>login</i> di <i>mobile</i> .
11	Membuat bagian <i>back-end</i> di <i>mobile</i> agar dapat <i>upload</i> foto saat absensi .
12	Membuat <i>database</i> baru secara online agar saat melakukan <i>fetch</i> lebih mudah
13	Membenarkan <i>debugging</i> pada aplikasi <i>mobile</i> di bagian <i>Profile</i>
14	Melakukan <i>debugging</i> secara keseluruhan pada aplikasi <i>desktop</i> dan <i>mobile</i>

- Minggu pertama, melakukan perkenalan dengan *supervisor* dan melakukan penjelasan *project*.

- Minggu kedua, melakukan *meeting* dengan team dan *supervisor* untuk membahas ERD (*Entity Relationship Diagram*) awal.
- Minggu ketiga, mempelajari dasar dasar *back-end*, *Instalasi* dan konfigurasi environment pengembangan *back-end* (Bun, Node.js, MySQL).
- Minggu Keempat, memulai untuk belajar membuat koneksi antara *back-end* dengan *database* MySQL dan mempelajari struktur folder *back-end*.
- Minggu kelima membuat *database* sederhana untuk latihan awal dan melakukan *meeting* dengan *supervisor* untuk *Review* dan revisi ERD serta membuat tabel awal di MySQL sesuai hasil ERD yang telah disetujui.
- Minggu keenam merancang struktur *database* akhir untuk aplikasi absensi dan penjadwalan serta membuat tabel-tabel utama seperti *users*, *roles*, *attendance*, *schedule* dan *Shift*.
- Minggu ketujuh membuat API endpoint dasar untuk fitur utama aplikasi CRUD (*Create, Read, Update, Delete*).
- Minggu kedelapan menghubungkan relasi antar tabel di *database* (*foreign key* dan *constraint*). Serta membuat fitur absensi dan penjadwalan menggunakan *query* SQL.
- Minggu kesembilan mengembangkan sebagian *back-end* untuk versi *mobile* (struktur serupa dengan *desktop*). Dan melakukan *Review* hasil pekerjaan *back-end desktop* dan *mobile* dengan *supervisor*.
- Minggu kesepuluh melakukan koreksi dari hasil rapat dengan *supervisor* dan juga membuat bagian *back-end* untuk *login* di *mobile*.
- Minggu kesebelas Membuat bagian *back-end* di *mobile* agar dapat *upload* foto saat absensi.
- Minggu keduabelas Membuat *database* baru secara *online* agar saat melakukan *fetch* lebih mudah.
- Minggu ketigabelas Membenarkan *debugging* pada aplikasi *mobile* di bagian *Profile*.
- Minggu keempatbelas Melakukan *debugging* secara keseluruhan pada aplikasi *desktop* dan *mobile*.

3.2.1 Analisis Kebutuhan Pengguna (User Requirement)

Analisis kebutuhan pengguna dilakukan untuk mengidentifikasi fungsi-fungsi sistem yang dibutuhkan oleh pengguna berdasarkan proses bisnis di PT Hanielshen. Kebutuhan pengguna yang dianalisis dan dikembangkan dalam sistem ini dibatasi sesuai dengan ruang lingkup kerja magang, yaitu sistem absensi dan penjadwalan karyawan.

Berdasarkan hasil analisis, pengguna sistem dibagi menjadi dua peran utama, yaitu Admin dan Karyawan, dengan kebutuhan sebagai berikut:

Kebutuhan Pengguna Admin:

1. Admin dapat melakukan *login* ke dalam sistem.
2. Admin dapat mengelola data karyawan.
3. Admin dapat mengelola jadwal kerja karyawan.
4. Admin dapat melihat dan mengelola data absensi karyawan.

Kebutuhan Pengguna Karyawan:

1. Karyawan dapat melakukan *login* ke aplikasi.
2. Karyawan dapat melakukan absensi masuk dan keluar.
3. Karyawan dapat mengunggah foto sebagai bukti kehadiran.
4. Karyawan dapat melihat jadwal kerja yang telah ditentukan.

3.2.2 Metode Pengembangan Sistem (SDLC)

Metode pengembangan sistem yang digunakan dalam pelaksanaan kerja magang ini adalah *System Development Life Cycle* (SDLC) dengan model *Waterfall*. Model *Waterfall* dipilih karena memiliki tahapan yang terstruktur dan sistematis, sehingga sesuai untuk pengembangan sistem absensi dan penjadwalan karyawan di PT Hanielshen.

Tahapan SDLC yang dilakukan dalam pengembangan sistem ini adalah sebagai berikut:

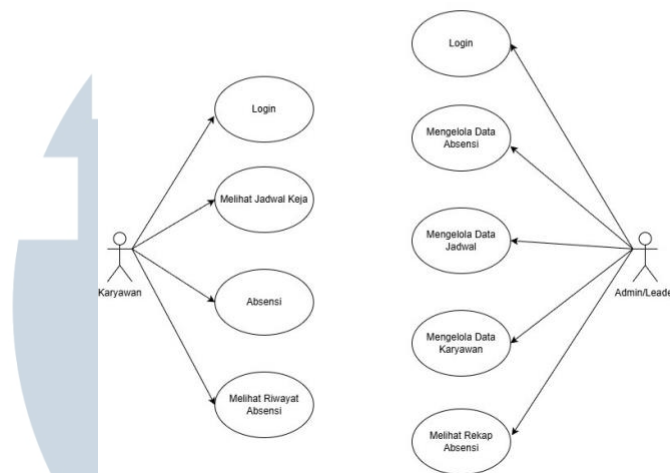
1. **Analisis Kebutuhan** : Pada tahap ini dilakukan analisis kebutuhan sistem dan pengguna dengan mempelajari proses bisnis perusahaan serta melakukan diskusi dengan pihak terkait untuk menentukan fitur yang akan dikembangkan.
2. **Perancangan Sistem** : Tahap perancangan sistem meliputi pembuatan *use case diagram*, *flowchart* sistem, serta perancangan struktur *database* yang digunakan sebagai dasar pengembangan sistem.
3. **Implementasi Sistem** : Pada tahap ini dilakukan pengembangan *back-end* sistem dan *database* menggunakan teknologi Node.js dan MySQL sesuai dengan hasil perancangan sistem.
4. **Pengujian Sistem** : Pengujian sistem dilakukan untuk memastikan seluruh fitur berjalan sesuai dengan kebutuhan yang telah ditentukan dan meminimalkan kesalahan pada sistem.
5. **Pemeliharaan Sistem** : Tahap pemeliharaan dilakukan dengan melakukan perbaikan *bug* serta penyesuaian sistem berdasarkan masukan dari pengguna.

3.2.3 Perancangan Sistem

A *Use case Diagram*

Web application hanielshen yang di buat memiliki *Use case diagram* agar mempermudah dan memberikan pemahaman dalam menggunakannya.

**USECASE DIAGRAM ABSENSI DAN PENJADWALAN PT
HANIELSHEN**



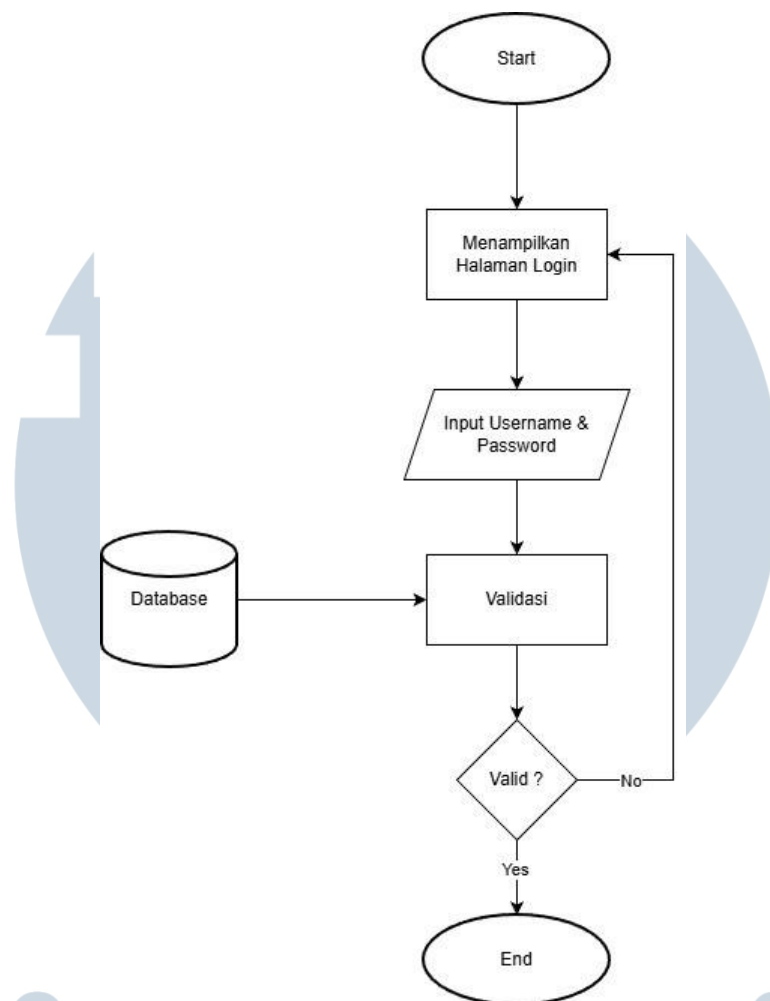
Gambar 3.2. Use case Diagram

Gambar 3.2 di atas merupakan *Use case diagram* bagi pengguna *desktop* dan *mobile*. Dari *Use case* tersebut memberitahu bahwa terdapat 2 pengguna yaitu karyawan dan juga admin. Admin dapat menggunakan aplikasi seperti *Login*, mengelola data absensi, mengelola data jadwal, mengelola data karyawan dan melihat rekap absensi. Sementara para pekerja dapat menggunakan aplikasi seperti *login*, melihat jadwal kerja, absensi dan melihat riwayat absensi.

B Flowchart

Web application ini memiliki *flowchart* di setiap fitur pada dan terdapat sistemnya masing - masing. Setiap sistem memiliki *CRUD* (*Create, Read, Update, Delete*) kecuali halaman *Login Desktop* .

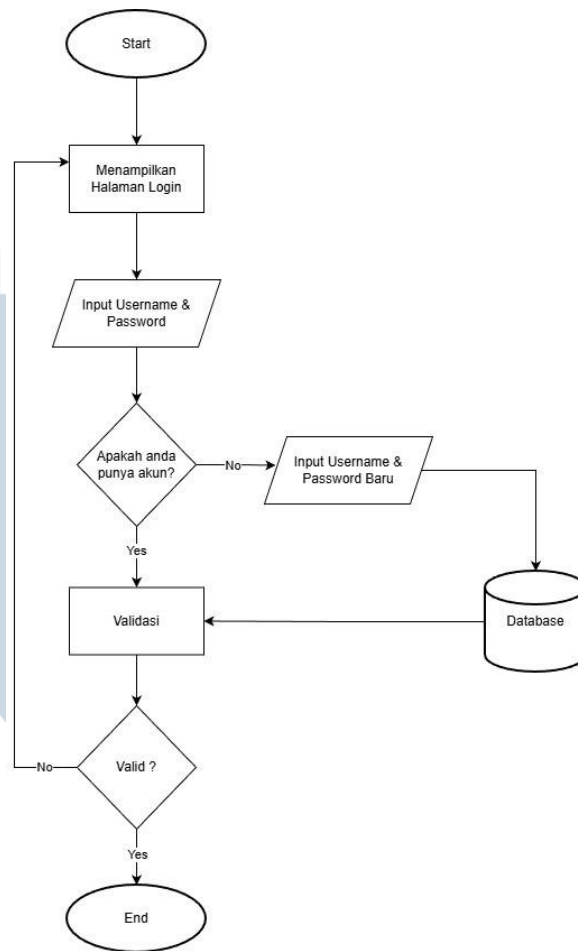
1. Flowchart Login Desktop



Gambar 3.3. *Flowchart Login Desktop*

pada gambar 3.3 di atas merupakan *flowchart login* yang berfungsi untuk masuk ke dalam *web application desktop* hanielshen dengan cara memasukan *username* dan *password* yang tersimpan di dalam *database*. Setelah melakukan *login* maka tampilan akan berpindah ke halaman *dashboard*.

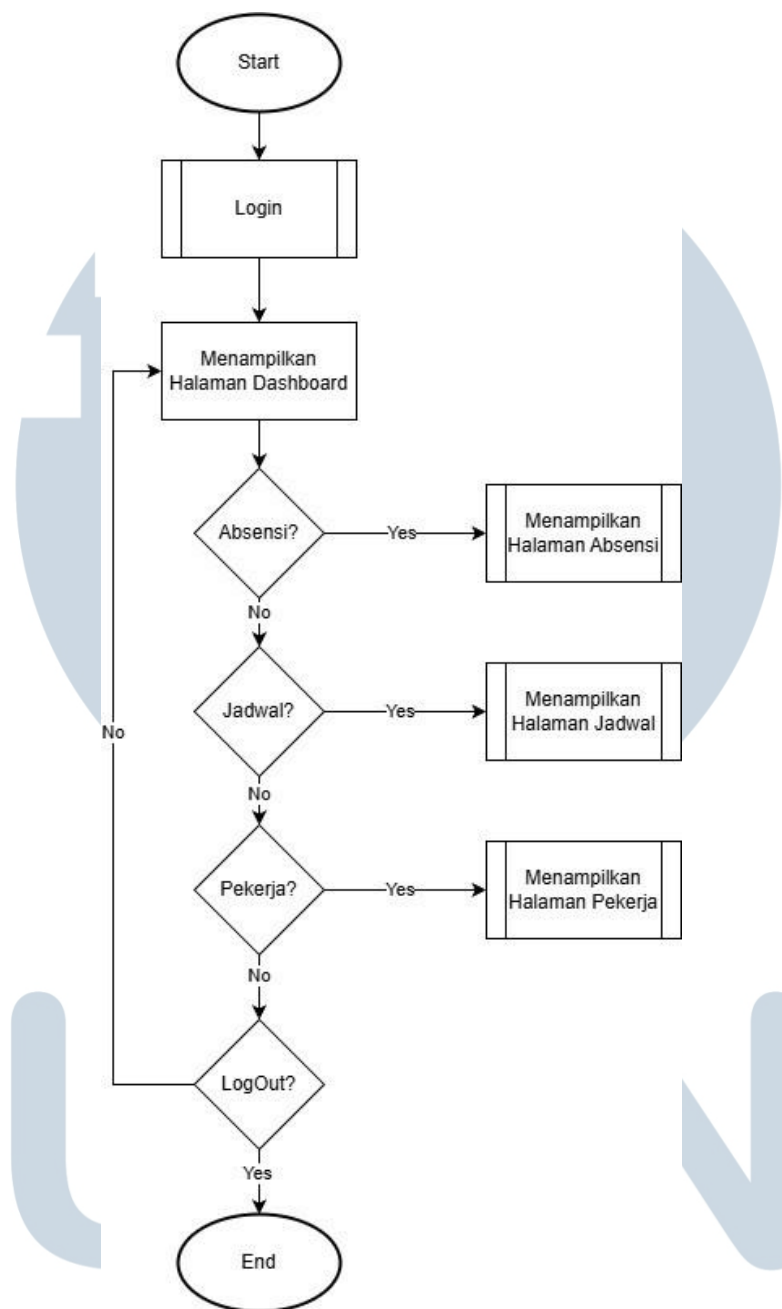
2. *Flowchart Login Mobile*



Gambar 3.4. *Flowchart Login Mobile*

pada gambar 3.4 di atas merupakan *flowchart login* untuk masuk ke dalam *web application* hanielshen versi *mobile* dengan memasukkan *username* dan *password* yang tersimpan di dalam *database*. Jika tidak memiliki *username* dan *password* anda dapat membuat *username* dan *password* yang baru agar dapat di simpan di dalam *database* dan bisa masuk ke dalam *web application* hanielshen versi *mobile*.

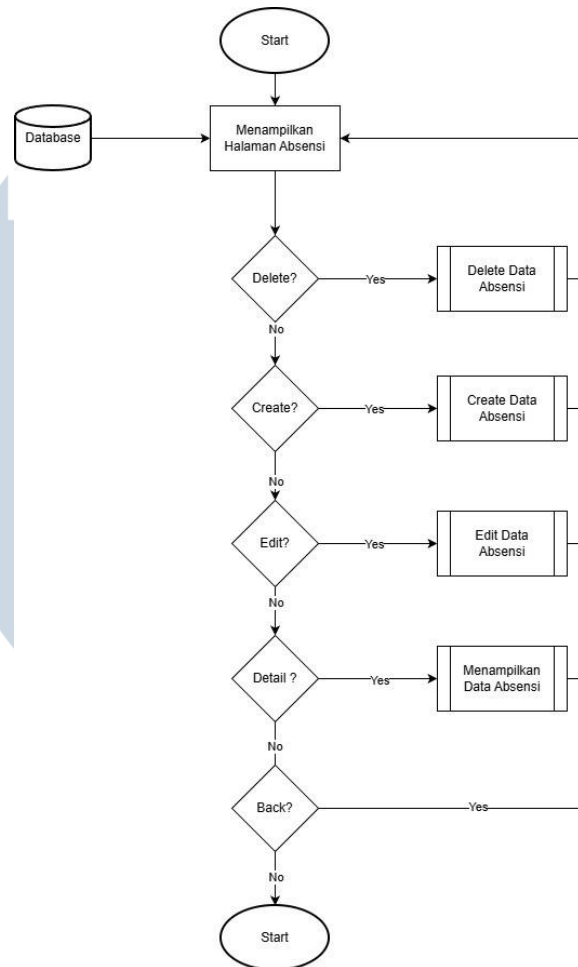
3. *Flowchart Dashboard Desktop*



Gambar 3.5. *Flowchart Dashboard Desktop*

pada gambar 3.5 di atas merupakan *flowchart* halaman *dashboard desktop*. Halaman *dashboard* menampilkan konten atau fitur yang dapat di tampilkan di *web application* hanielshen seperti absensi karyawan. Menu absensi menampilkan halaman absensi yang berisi data data dari absesnsi pekerja, lalu menu untuk Jadwal untuk menampilkan jadwal dari para pekerja dan yang terakhir menu untuk pekerja untuk menampilkan para pekerja.

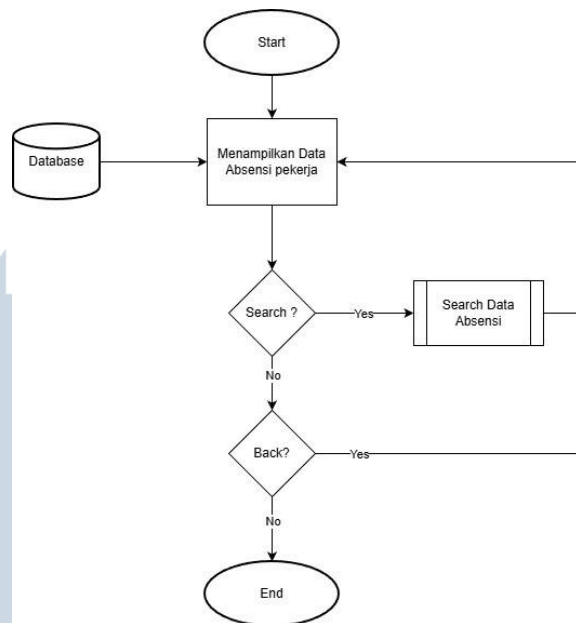
4. *Flowchart Halaman Absensi*



Gambar 3.6. *Flowchart Halaman Absensi*

Gambar 3.6 di atas merupakan *flowchart* halaman absensi. Halaman absensi menampilkan beberapa fitur seperti menambah data, mengedit data dan juga melihat *detail* data.

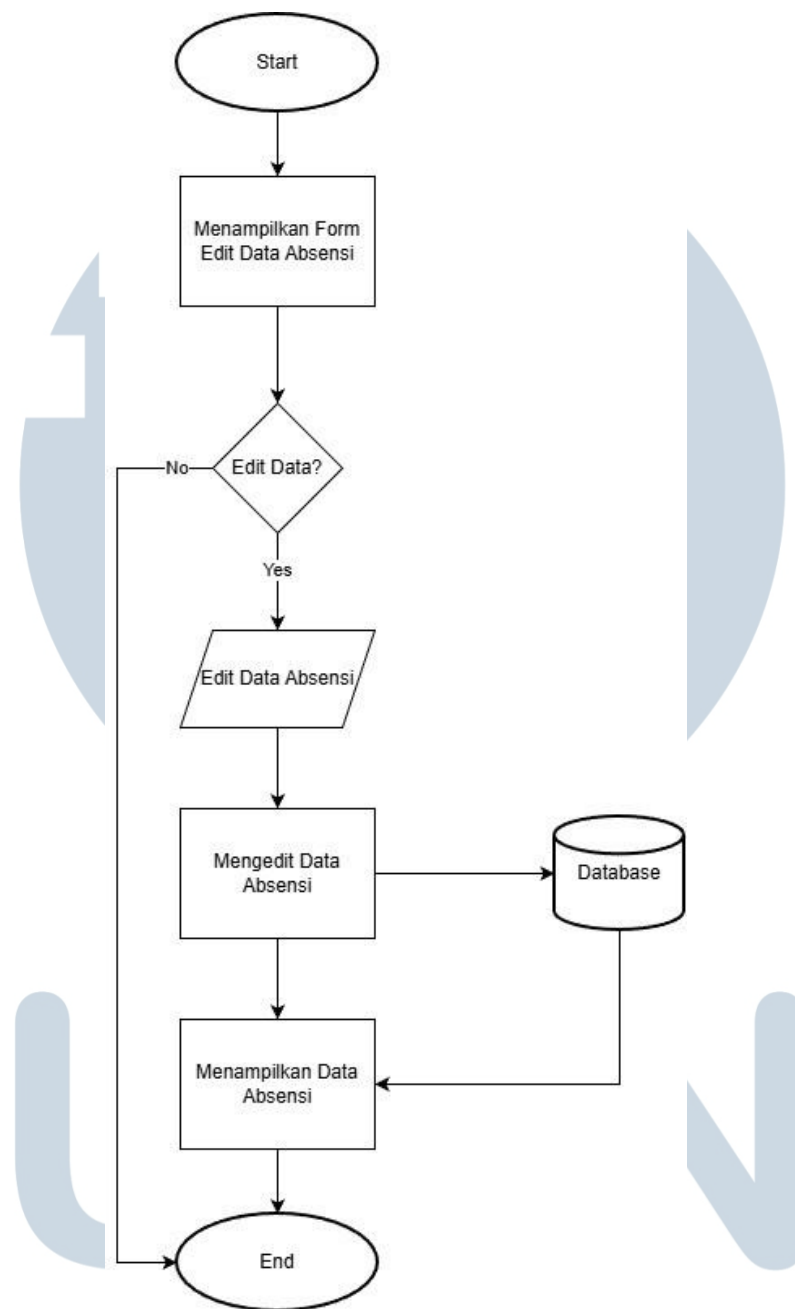
5. *Flowchart Detail Data Absensi*



Gambar 3.7. *Flowchart Detail Data Absensi*

Pada gambar 3.7 di atas merupakan *flowchart detail* pada halaman absensi. halaman *detail* memiliki fungsi untuk menampilkan data absensi secara lengkap dan untuk menampilkan informasi dari absensi dengan maksimal.

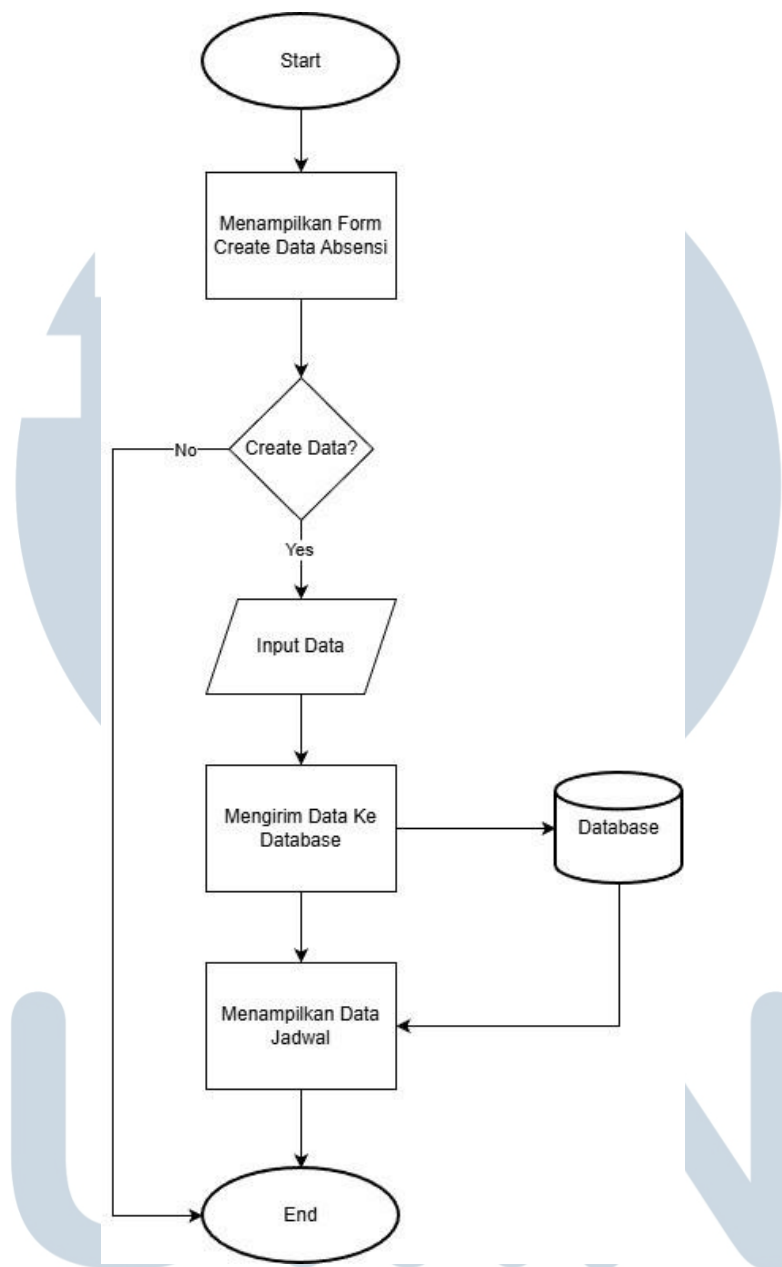
6. *Flowchart Edit Data Absensi*



Gambar 3.8. *Flowchart Edit Data Absensi*

Pada gambar 3.8 di atas merupakan *flowchart* untuk mengedit data jadwal di halaman absensi. Saat ingin melakukan pengeditan data absensi akan ada *form* yang muncul yang berisikan data yang sudah ada di *database*..

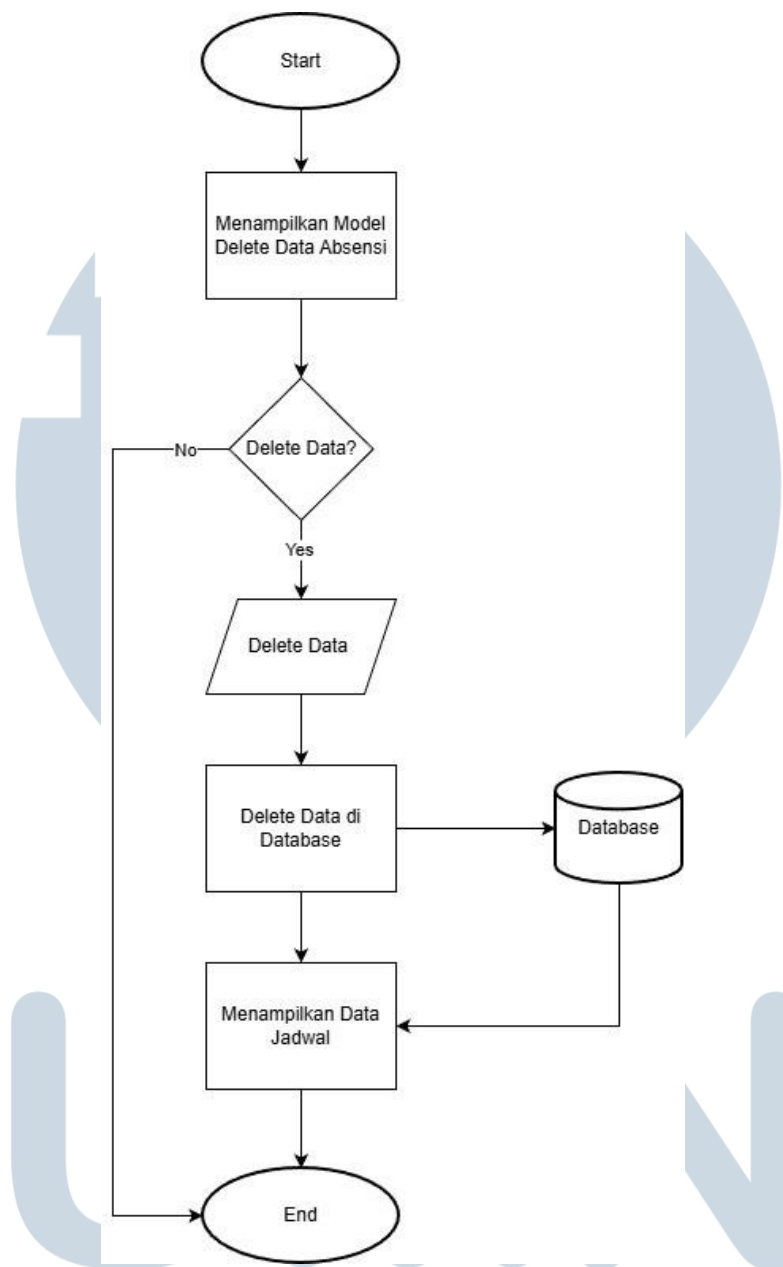
7. *Flowchart Create Data Absensi*



Gambar 3.9. *Flowchart Create Data Absensi*

Pada Gambar 3.9 di atas merupakan *flowchart* untuk menambahkan data atau membuat data baru pada halaman absensi. Saat menambahkan data akan muncul *form* yang menampilkan data absensi yang perlu di isi sesuai dengan jam masuk dan keluar dari perusahaan.

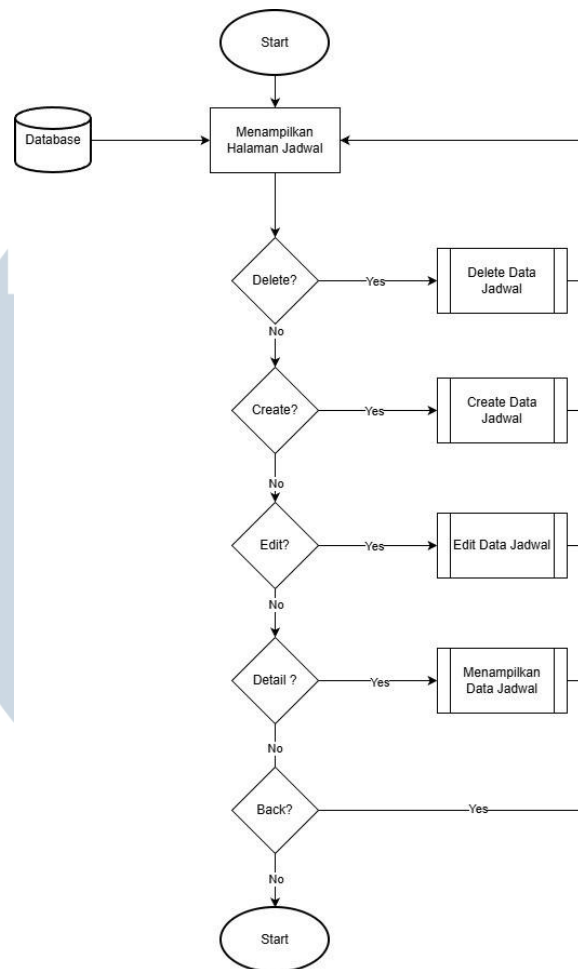
8. *Flowchart Delete Data Absensi*



Gambar 3.10. *Flowchart Delete Data Absensi*

pada gambar 3.10 di atas adalah *flowchart* untuk menghapus data pada halaman absensi. Penghapusan data pada halaman ini memiliki beberapa proses seperti gambar yang ada di 3.10. Saat penghapusan data absensi dilakukan maka data yang di hapus akan terhapus juga di dalam *database* sehingga tampilan pada halaman absensi akan berubah sesuai dengan isi dari *database*

9. *Flowchart* Halaman Jadwal

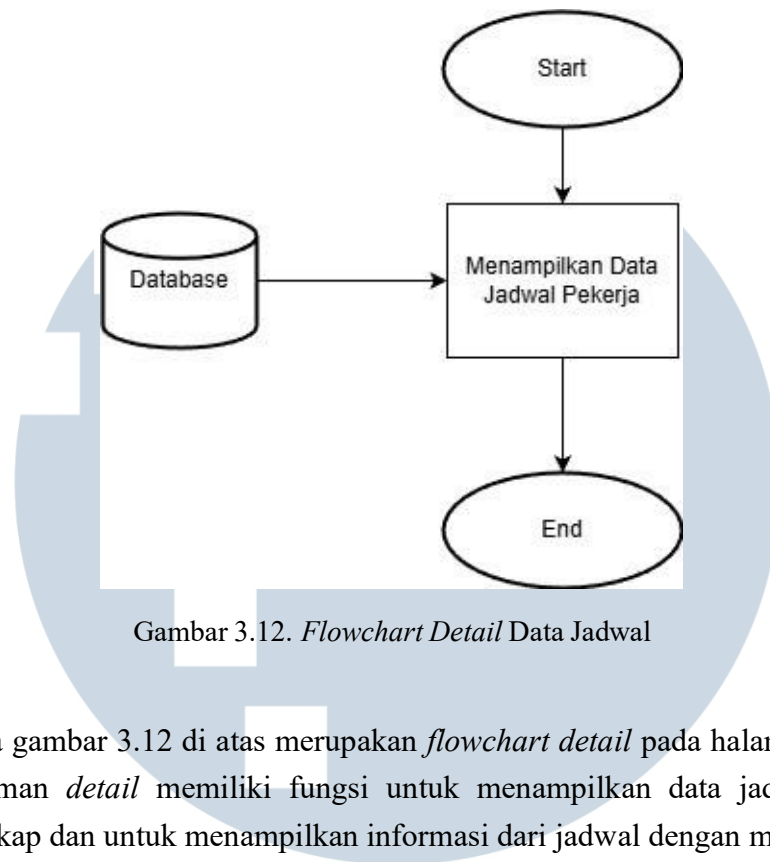


Gambar 3.11. Flowchart Halaman Jadwal

Gambar 3.11 di atas merupakan *flowchart* halaman jadwal. Memiliki fitur yang sama dengan halaman absensi seperti *detail*, *edit*, *create* dan *delete*.

10. *Flowchart Detail Data Jadwal*

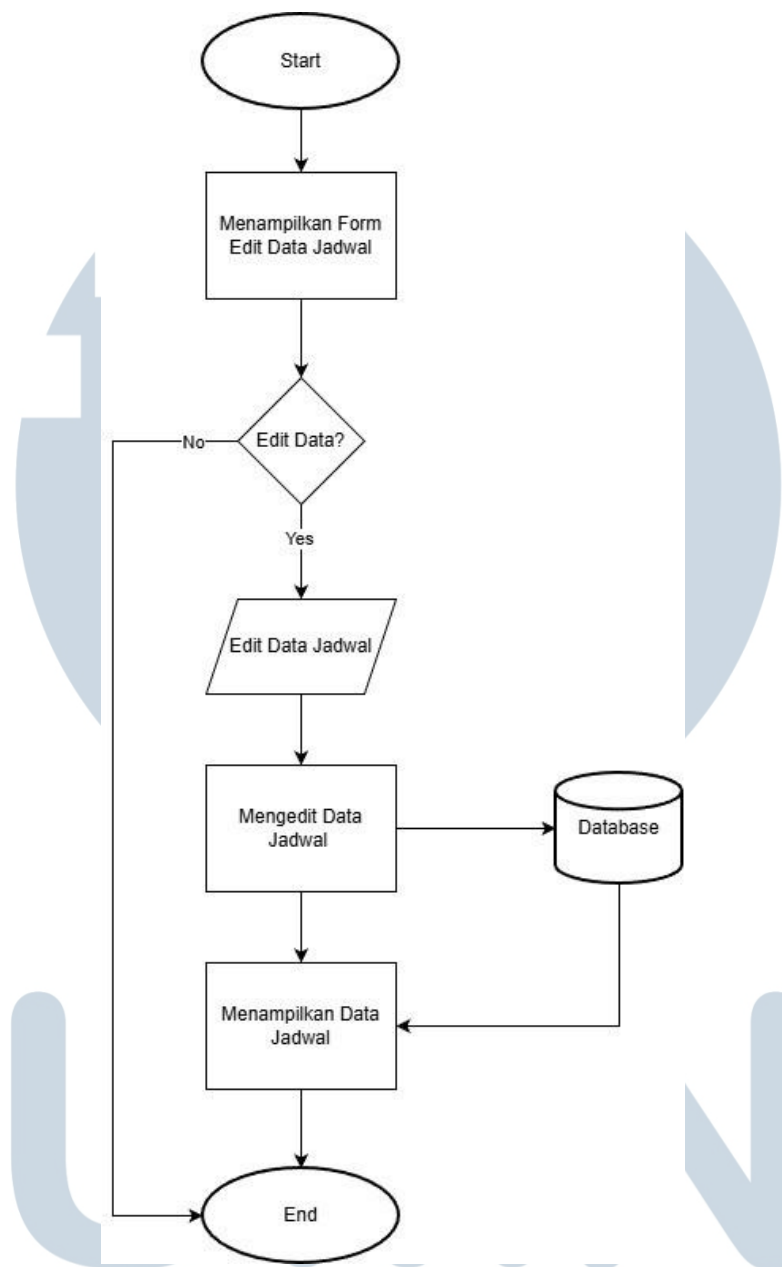
U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.12. *Flowchart Detail Data Jadwal*

Pada gambar 3.12 di atas merupakan *flowchart detail* pada halaman jadwal. halaman *detail* memiliki fungsi untuk menampilkan data jadwal secara lengkap dan untuk menampilkan informasi dari jadwal dengan maksimal.

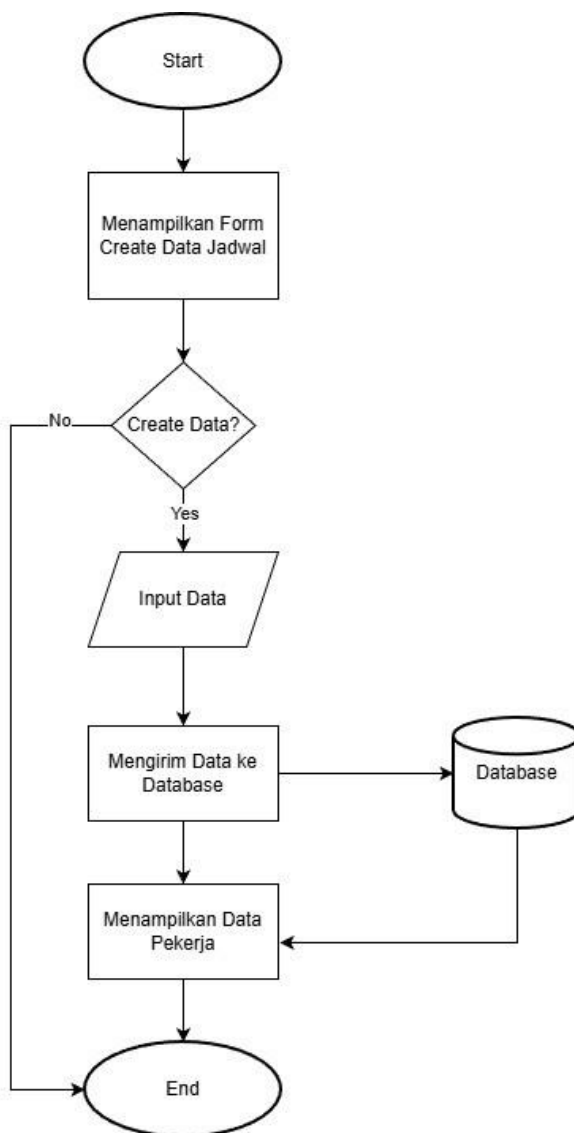
11. *Flowchart Edit Data Jadwal*



Gambar 3.13. *Flowchart Edit Data Jadwal*

Pada gambar 3.13 di atas merupakan *flowchart* untuk mengedit data jadwal di halaman jadwal. Saat ingin melakukan pengeditan data jadwal akan ada *form* yang muncul yang berisikan data yang sudah ada di *database*.

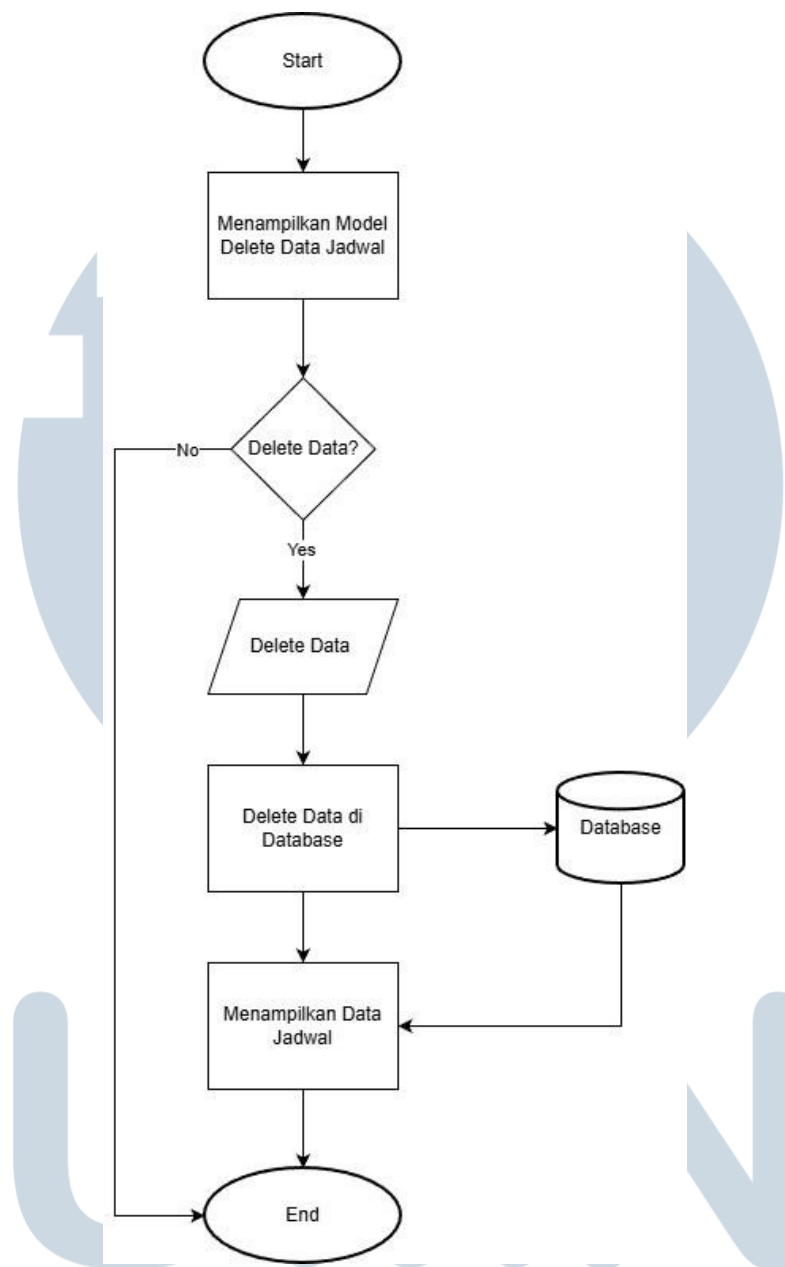
12. *Flowchart Create Data Jadwal*



Gambar 3.14. Flowchart *Create Data Jadwal*

Pada Gambar 3.14 di atas merupakan *flowchart* untuk menambahkan data atau membuat data baru pada halaman jadwal. Saat menambahkan data akan muncul *form* yang menampilkan data jadwal yang perlu di isi sesuai dengan kebutuhan perusahaan.

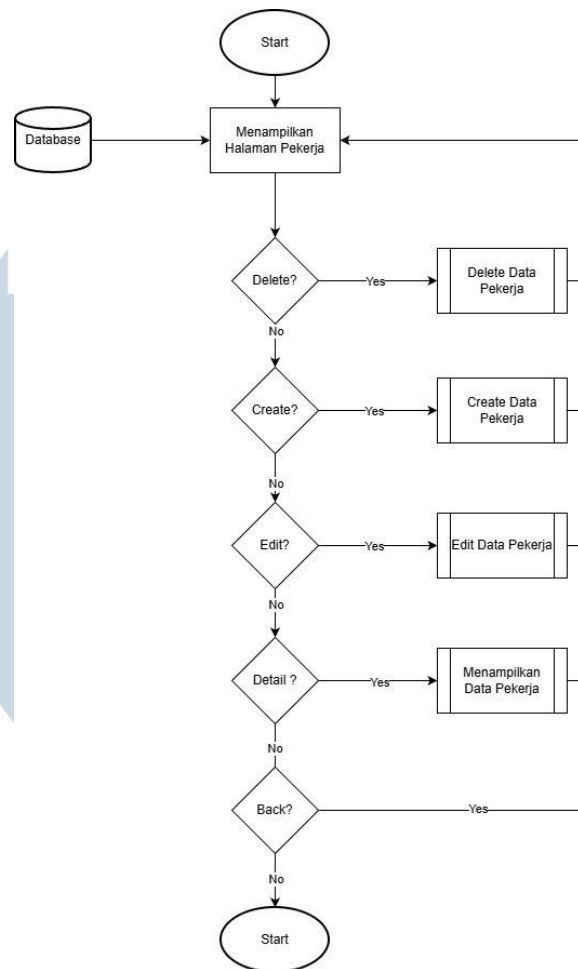
13. *Flowchart Delete Data Jadwal*



Gambar 3.15. *Flowchart Delete Data Jadwal*

pada gambar 3.15 di atas adalah *flowchart* untuk menghapus data pada halaman jadwal. Penghapusan data pada halaman ini memiliki beberapa proses seperti gambar yang ada di 3.15. Saat penghapusan data jadwal di lakukan maka data yang di hapus akan terhapus juga di dalam *database* sehingga tampilan pada halaman jadwal akan berubah sesuai dengan isi dari *database*.

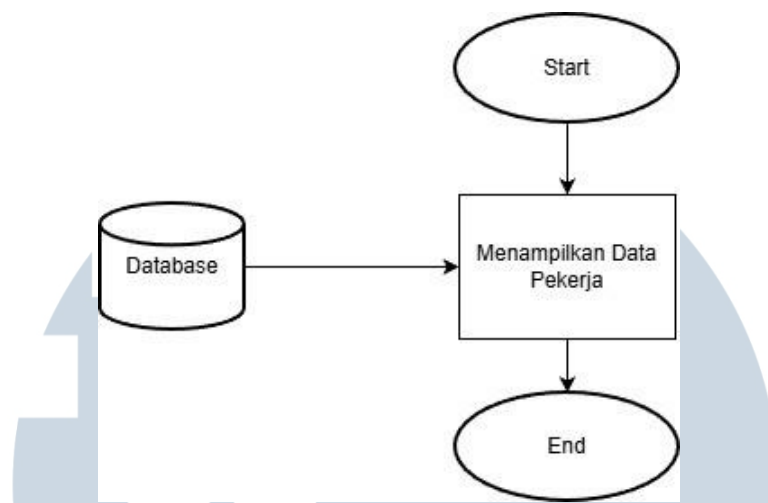
14. *Flowchart* Halaman Pekerja



Gambar 3.16. *Flowchart* Halaman Pekerja

Gambar 3.16 di atas merupakan *flowchart* halaman pekerja. Memiliki fitur yang sama dengan halaman absensi dan juga jadwal seperti *detail*, *edit*, *create* dan *delete*.

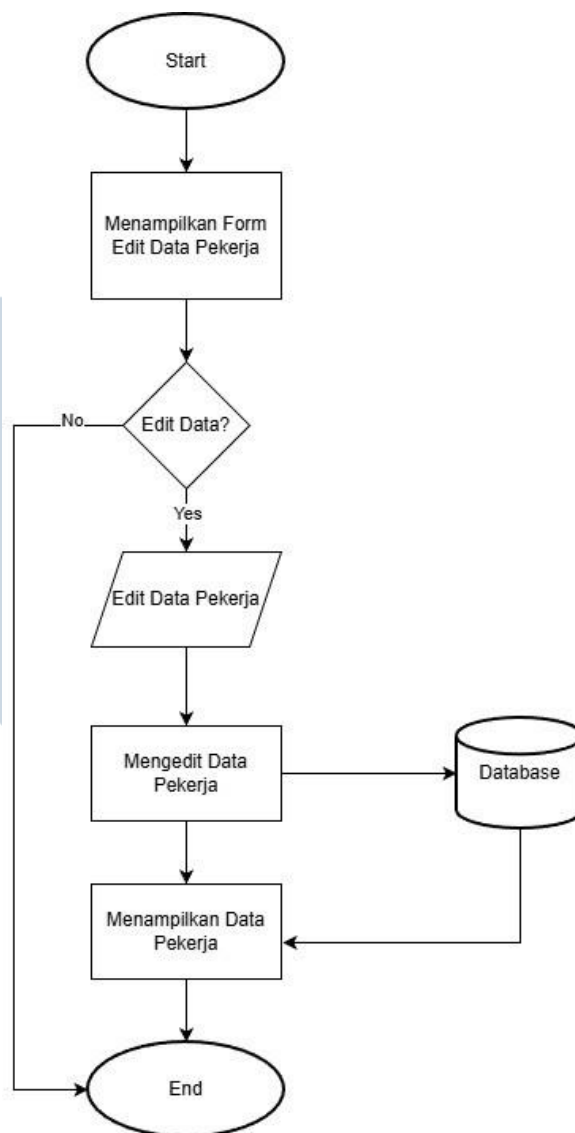
15. *Flowchart* Detail Data Pekerja



Gambar 3.17. *Flowchart Detail Data Pekerja*

Pada gambar 3.17 di atas merupakan *flowchart detail* pada halaman pekerja. halaman *detail* memiliki fungsi untuk menampilkan data pekerja secara lengkap dan untuk menampilkan informasi dari para pekerja dengan maksimal.

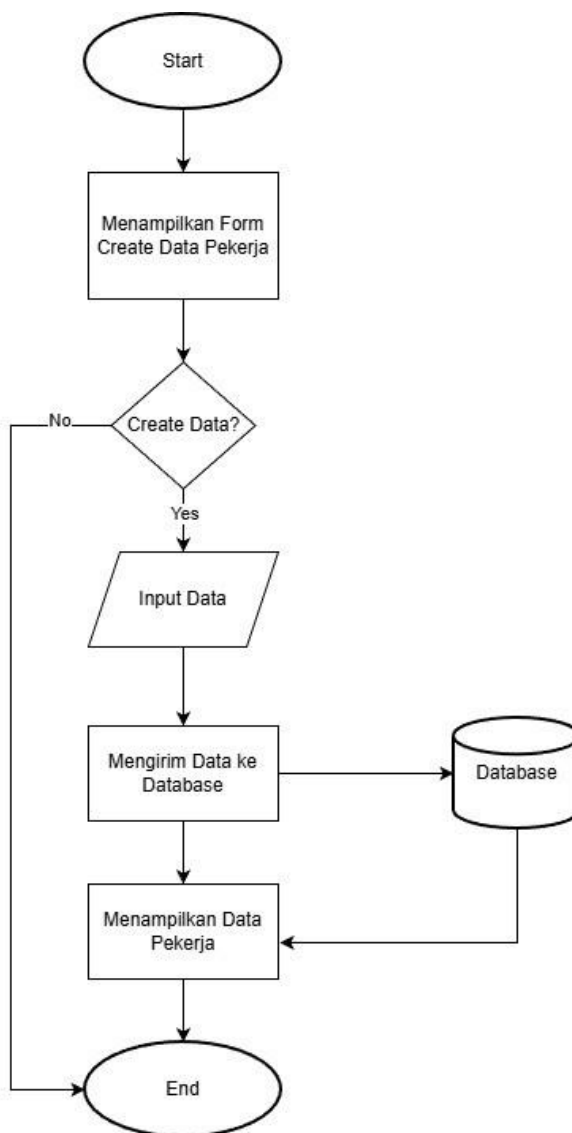
16. *Flowchart Edit Data Pekerja*



Gambar 3.18. *Flowchart Edit Data Pekerja*

Pada gambar 3.18 di atas merupakan *flowchart* untuk mengedit data para pekerja di halaman pekerja. Saat ingin melakukan pengeditan data pekerja akan ada *form* yang muncul yang berisikan data yang sudah ada di *database*.

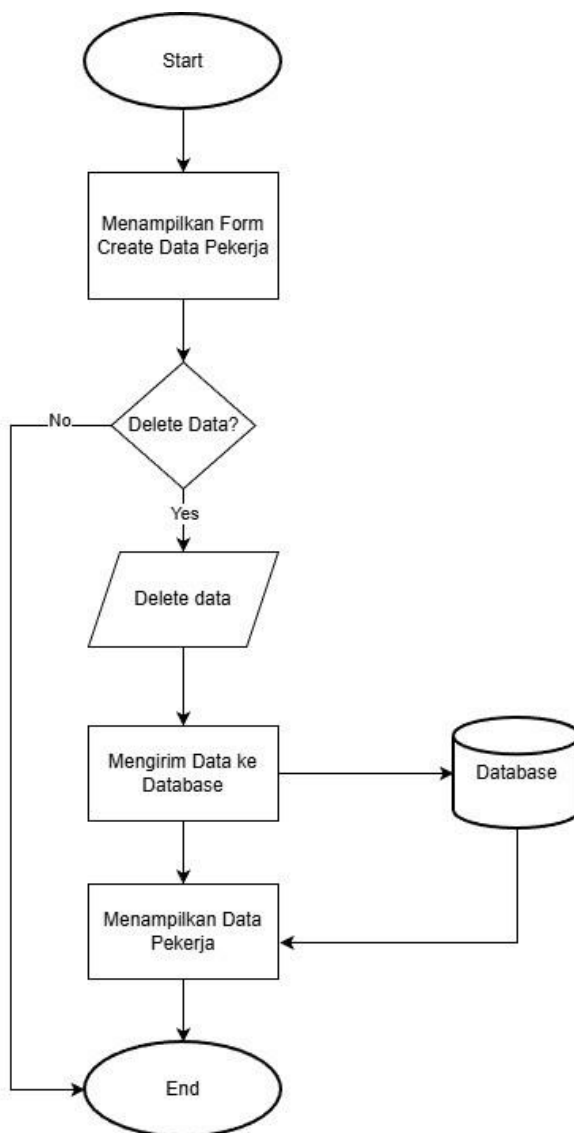
17. *Flowchart Create Data Pekerja*



Gambar 3.19. *Flowchart Create Data Pekerja*

Pada Gambar 3.19 di atas merupakan *flowchart* untuk menambahkan data atau membuat data baru pada halaman pekerja. Saat menambahkan data akan muncul *form* yang menampilkan data para pekerja yang harus diisi sesuai dengan pekerja yang masuk atau yang belum terdata dan data harus terisi dengan lengkap.

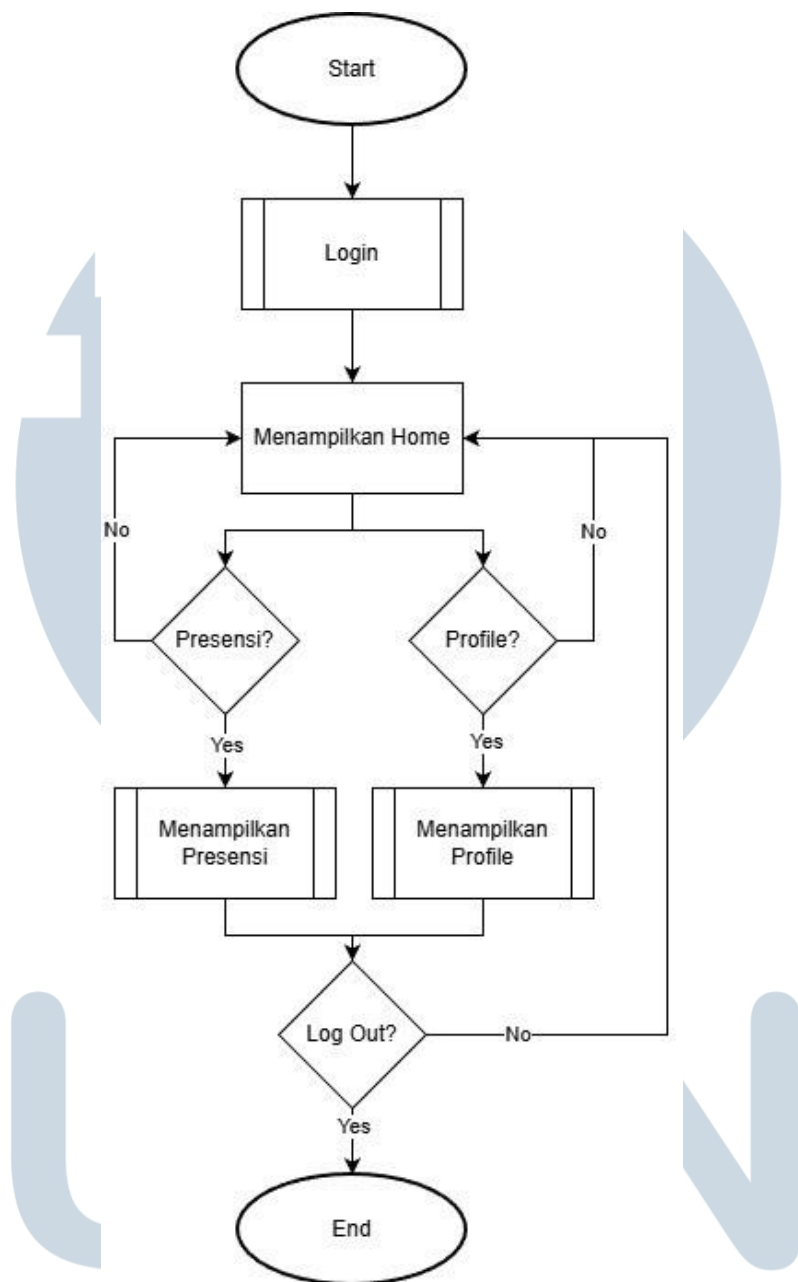
18. *Flowchart Delete Data Pekerja*



Gambar 3.20. *Flowchart Delete Data Pekerja*

pada gambar 3.20 di atas adalah *flowchart* untuk menghapus data pada halaman pekerja. Penghapusan data pada halaman ini memiliki beberapa proses seperti gambar yang ada di 3.20. Saat penghapusan data pekerja di lakukan maka data yang di hapus akan terhapus juga di dalam *database* sehingga tampilan pada halaman pekerja akan berubah sesuai dengan isi dari *database*.

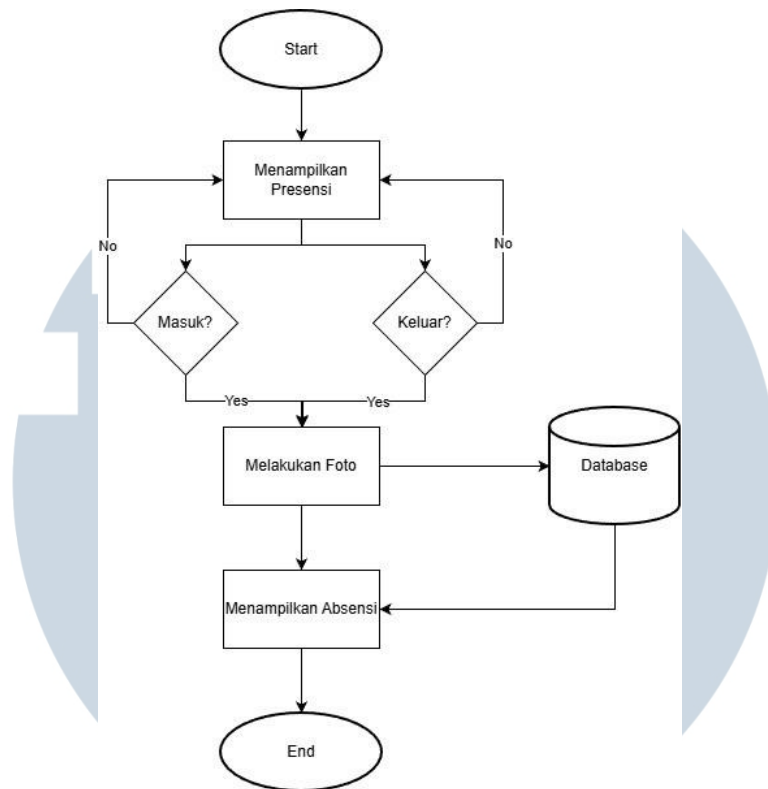
19. Flowchart *Mobile*



Gambar 3.21. *Flowchart Mobile*

Gambar 3.21 di atas merupakan *flowchart Mobile*. Pada tampilan awal *mobile* menampilkan *home* dengan isinya berupa jadwal pekerja jadi agar para pekerja mudah untuk melihatnya. Lalu selain menu *home* ada juga beberapa fitur seperti presensi dan juga *profile*.

20. *Flowchart Mobile Presensi*

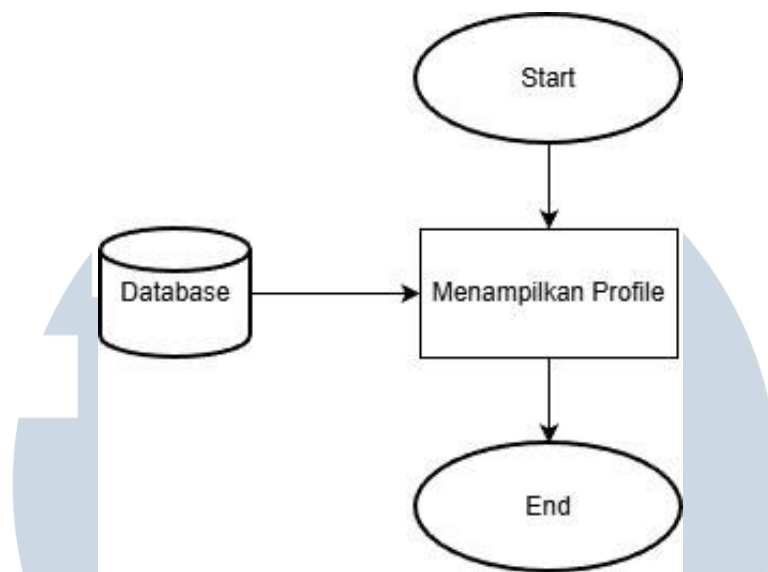


Gambar 3.22. *Flowchart Mobile Presensi*

Gambar 3.22 di atas merupakan *flowchart mobile presensi*. Presensi di mana para pekerja untuk absensi saat berada dalam jam kerja dan ada 2 fitur yaitu presensi masuk dan keluar dan saat presensi para pekerja diwajibkan untuk melakukan foto agar memberikan bukti tambahan untuk presensi.

21. *Flowchart Mobile Profile*

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

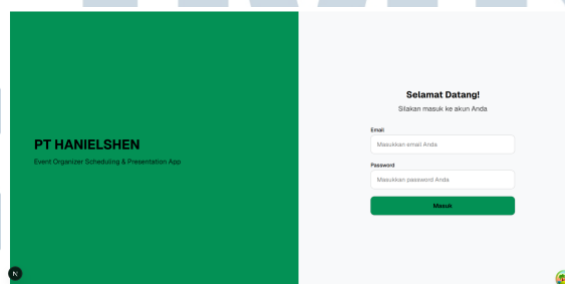


Gambar 3.23. Flowchart Mobile Profile

Gambar 3.23 di atas merupakan *flowchart mobile profile*. Pada bagian *profile* hanya menampilkan data diri pekerja saja.

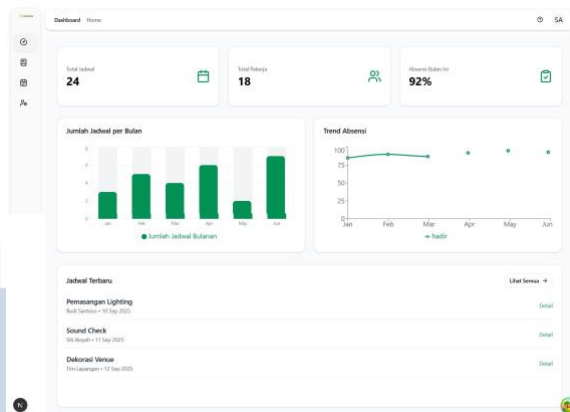
3.3 Aplikasi Hanielshen

Tampilan awal dari *website application* adalah tampilan untuk *login*. Dengan memasukkan *username* dan *password* yang terdapat di *database*, pekerja yang memiliki *role* admin dapat masuk ke dalam tampilan utama *dashboard* jika *login* berhasil.



Gambar 3.24. Login Dashboard Hanielshen

Selanjutnya adalah tampilan *dashboard*. *Dashboard* menampilkan info - info tentang jadwal baru yang ada lalu ada juga jumlah pekerja, jumlah jadwal perbulan ,absensi bulan ini dan *trend* absensi.



Gambar 3.25. Dashboard Hanielshen

Selanjutnya adalah tampilan halaman absensi. Halaman absensi ini menampilkan absensi karyawan sesuai dengan jadwal mereka yang sudah di buat oleh admin dan nantinya mereka akan melakukan absensi di versi *mobile*. Dan di dalam halaman absensi ini Dalam halaman ini terdapat ”+ Tambah Absensi” yang berfungsi untuk memasukan data absensi baru jika ada data absensi yang belum masuk selain itu di halaman ini juga admin bisa mengedit absensi dengan ikon pensil berwarna hijau dan juga dapat menghapus absensi pekerja dengan ikon tong sampah merah jika tidak sesuai.

The table displays the following data:

Nama	Tanggal	Jam Masuk	Jam Keluar	Status	Aksi
Andi Supriya	2023-01-11	08:00	17:00	Hadir	[Edit] [Hapus]
Budi Santosa	2023-01-11	08:15	17:00	Terdenda	[Edit] [Hapus]
Citra Dewi	2023-01-11	08:00	17:00	Hadir	[Edit] [Hapus]
Dedi Gunawan	2023-01-11	09:00	08:00	Absen	[Edit] [Hapus]
Eka Lestari	2023-01-11	08:00	17:00	Hadir	[Edit] [Hapus]
Fajar Nugraha	2023-01-11	09:00	17:00	Terdenda	[Edit] [Hapus]
Gita Pratika	2023-01-11	08:00	17:00	Hadir	[Edit] [Hapus]
Hendra Wijaya	2023-01-11	08:05	17:00	Terdenda	[Edit] [Hapus]
Irena Marlina	2023-01-11	08:00	17:00	Hadir	[Edit] [Hapus]
Joko Setiawan	2023-01-11	08:00	17:00	Absen	[Edit] [Hapus]

Gambar 3.26. Absensi Hanielshen

Berikut ini adalah tampilan *form* untuk menambahkan absensi dan admin bisa memasukan data yang sesuai dengan *form* tersebut.

Gambar 3.27. *Form Absensi Hanielshen*

Selanjutnya adalah tampilan dari halaman jadwal. Halaman jadwal ini menampilkan jadwal karyawan yang telah di buat oleh admin. Selain itu dalam halaman jadwal terdapat ”+ Tambah Jadwal” yang berfungsi untuk memasukan data , ikon pensil berwarna hijau yang berfungsi untuk memperbarui data dan ikon tempat sampah berwarna merah yang berfungsi untuk menghapus data jadwal.

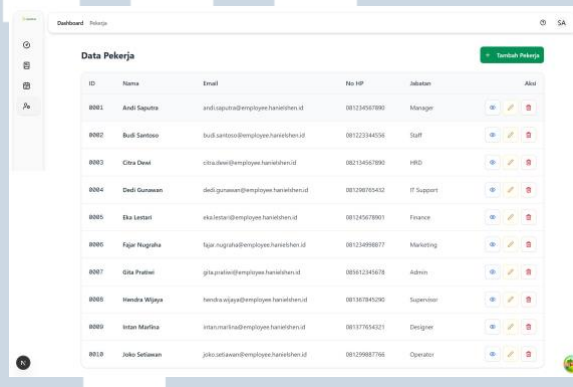
Gambar 3.28. *Jadwal Hanielshen*

Berikut ini adalah tampilan *form* untuk menambahkan jadwal dan admin bisa memasukan data yang sesuai dengan *form* tersebut.

Gambar 3.29. *Form Jadwal Hanielshen*

Selanjutnya adalah tampilan dari halaman pekerja. Halaman pekerja ini menampilkan jadwal pekerja yang telah di buat oleh admin. Selain itu

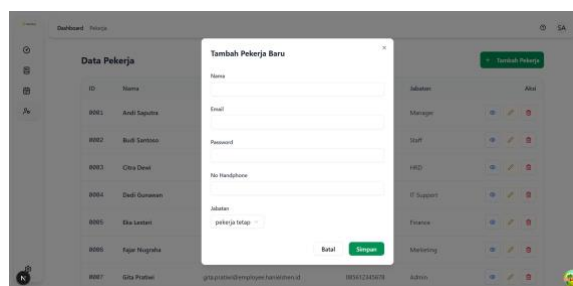
dalam halaman pekerja terdapat ”+ Tambah Jadwal” yang memiliki fungsi untuk memasukan data yang telah di isi di dalam *form*, ikon pensil yang berwarna hijau memiliki fungsi untuk mengubah data atau memperbarui data dan yang terakhir terdapat ikon tempat sampah berwarna merah yang memiliki fungsi untuk menghapus data para pekerja.



ID	Nama	Email	No HP	Jabatan	Aksi
0001	Andi Saputra	andi.saputra@employee.hanielshen.id	081234567890	Manager	[Add] [Edit] [Delete]
0002	Budi Santoso	budi.santoso@employee.hanielshen.id	081234567890	Staff	[Add] [Edit] [Delete]
0003	Citra Dewi	citra.dewi@employee.hanielshen.id	082345678901	HRD	[Add] [Edit] [Delete]
0004	Dedi Gunawan	dedi.gunawan@employee.hanielshen.id	081234567890	IT Support	[Add] [Edit] [Delete]
0005	Eka Santia	eka.santia@employee.hanielshen.id	081234567890	Finance	[Add] [Edit] [Delete]
0006	Fajar Nugraha	fajar.nugraha@employee.hanielshen.id	081234567890	Marketing	[Add] [Edit] [Delete]
0007	Gita Pratika	gita.pratika@employee.hanielshen.id	081234567890	Admin	[Add] [Edit] [Delete]
0008	Hendra Wijaya	hendra.wijaya@employee.hanielshen.id	081234567890	Supervisor	[Add] [Edit] [Delete]
0009	Irena Marlina	irena.marlina@employee.hanielshen.id	081234567890	Designer	[Add] [Edit] [Delete]
0010	Joko Setiawan	joko.setiawan@employee.hanielshen.id	081234567890	Operator	[Add] [Edit] [Delete]

Gambar 3.30. Data Pekerja Hanielshen

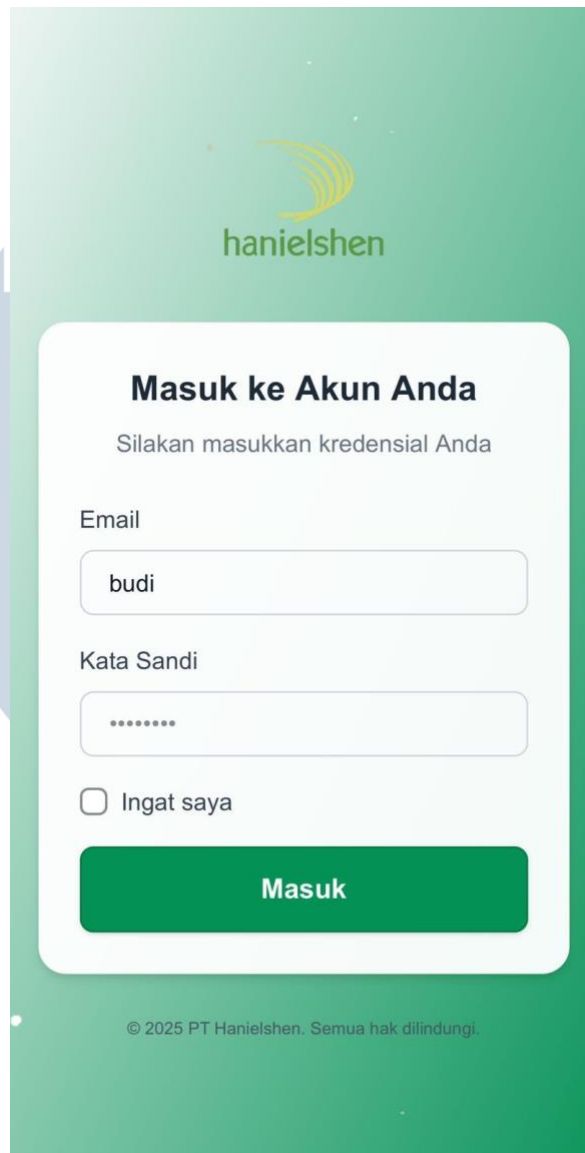
Berikut ini adalah tampilan *form* untuk menambahkan pekerja dan admin bisa memasukan data yang sesuai dengan *form* tersebut.



Gambar 3.31. Data Pekerja Hanielshen

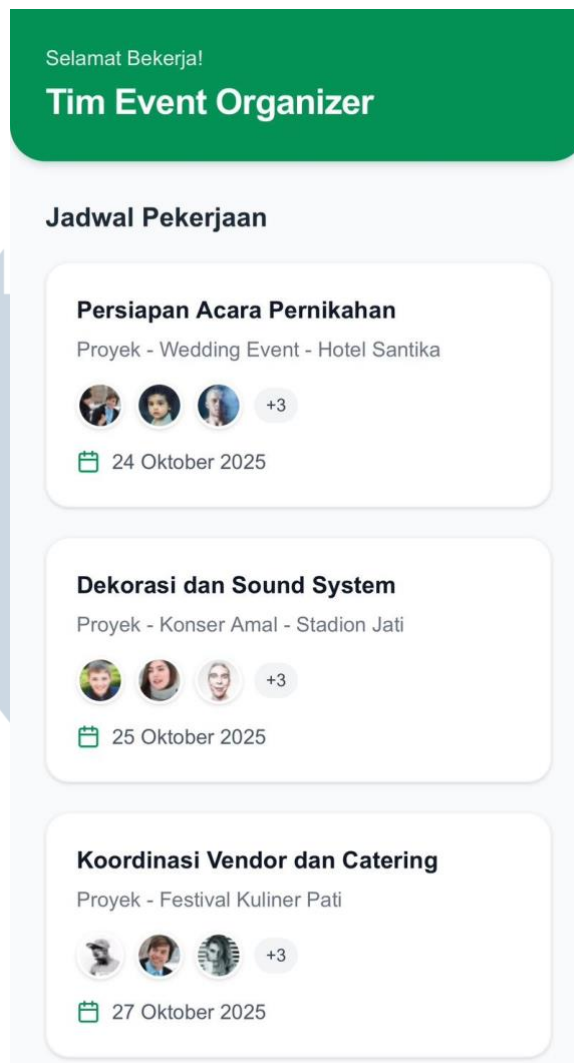
3.4 Aplikasi *Mobile* Hanielshen

Pada tampilan awal dari aplikasi *mobile* terdapat tampilan *login*. Yang mewajibkan kita untuk memasukan *username* dan *password* yang telah di berikan oleh admin. Semua pekerja dapat mengakses aplikasi *mobile* apabila proses *login* berhasil dilakukan.

The image shows a mobile login interface for 'hanielshen'. At the top, there is a logo consisting of a yellow stylized bird or wing shape above the text 'hanielshen' in green. Below the logo, the title 'Masuk ke Akun Anda' is displayed in bold black text, followed by the instruction 'Silakan masukkan kredensial Anda' in a smaller font. The form contains two input fields: 'Email' with the value 'budi' and 'Kata Sandi' (Password) with masked characters '.....'. There is a checkbox labeled 'Ingat saya' (Remember me) which is currently unchecked. A large green button with the text 'Masuk' (Login) is positioned below the inputs. At the bottom of the screen, a small copyright notice reads '© 2025 PT Hanielshen. Semua hak dilindungi.' (All rights reserved).

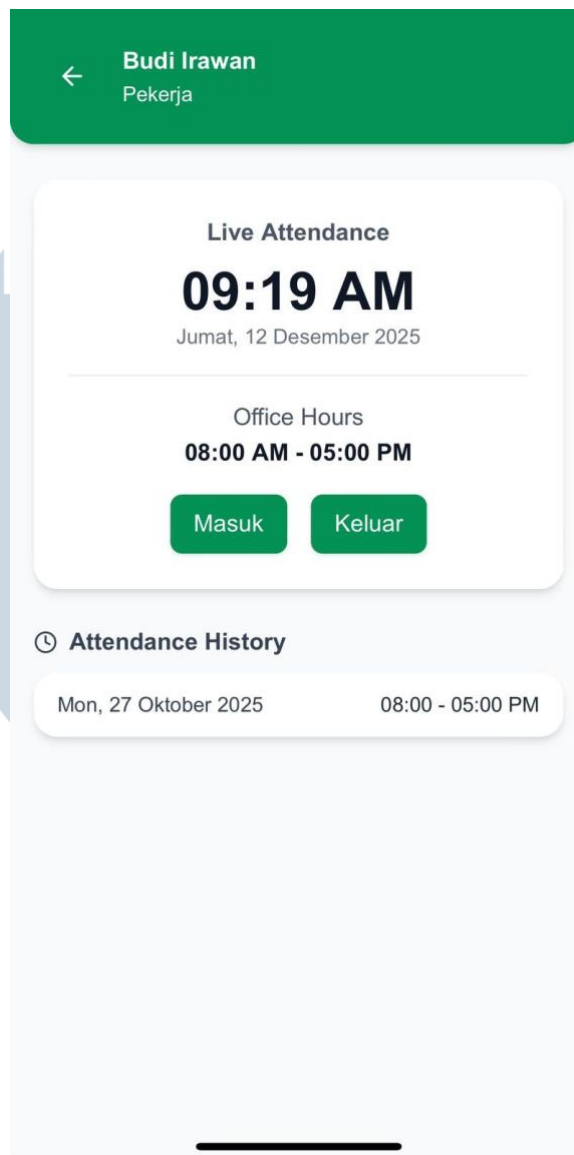
Gambar 3.32. Login *Mobile*

Selanjutnya terdapat tampilan *home* pada *mobile*. *Mobile* menampilkan jadwal pekerja yang di ambil dari *database* yang sebelumnya di buat oleh admin di *dekstop*. Selain itu terdapat juga 2 fitur lainnya yaitu absensi dan juga *profile*.



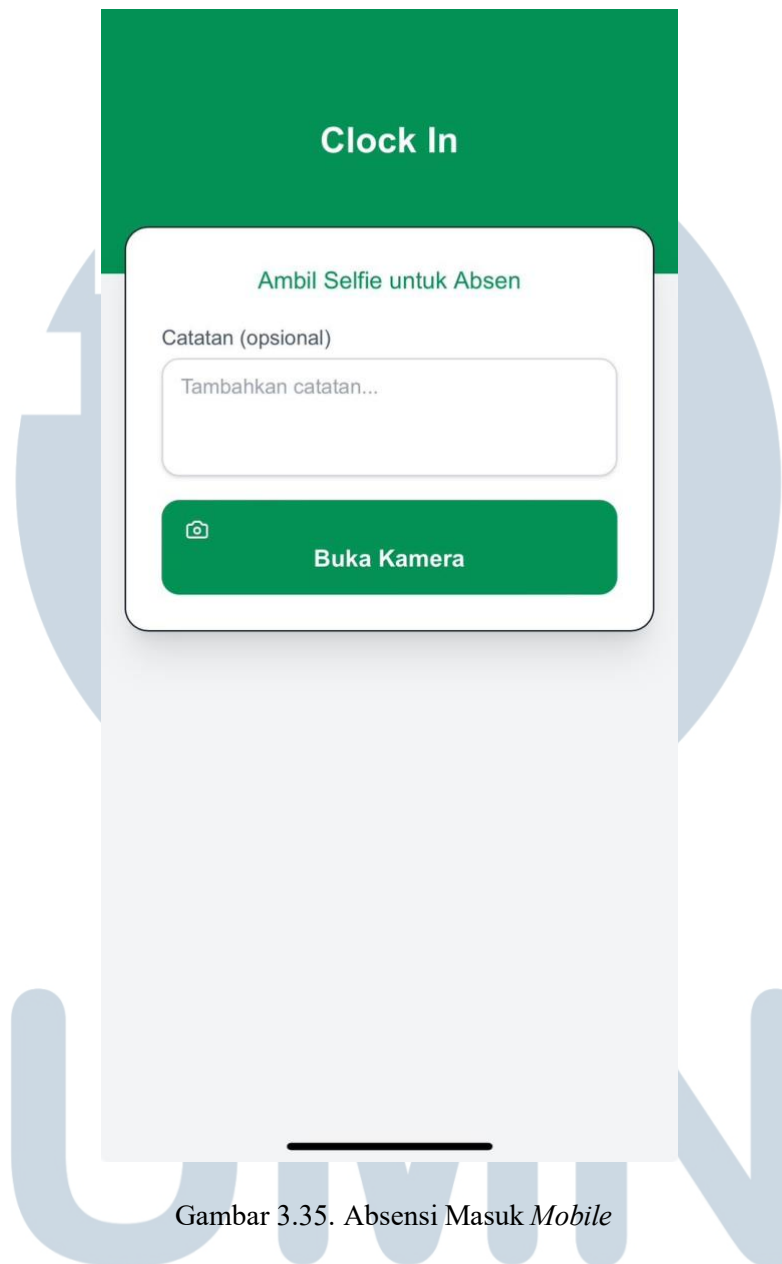
Gambar 3.33. *Home Mobile*

Selanjutnya adalah tampilan untuk absensi yang menampilkan jam dan juga tanggal hari ini serta ada fitur untuk masuk absensi dan keluar absensi jika di *click* akan menuju halaman lain.



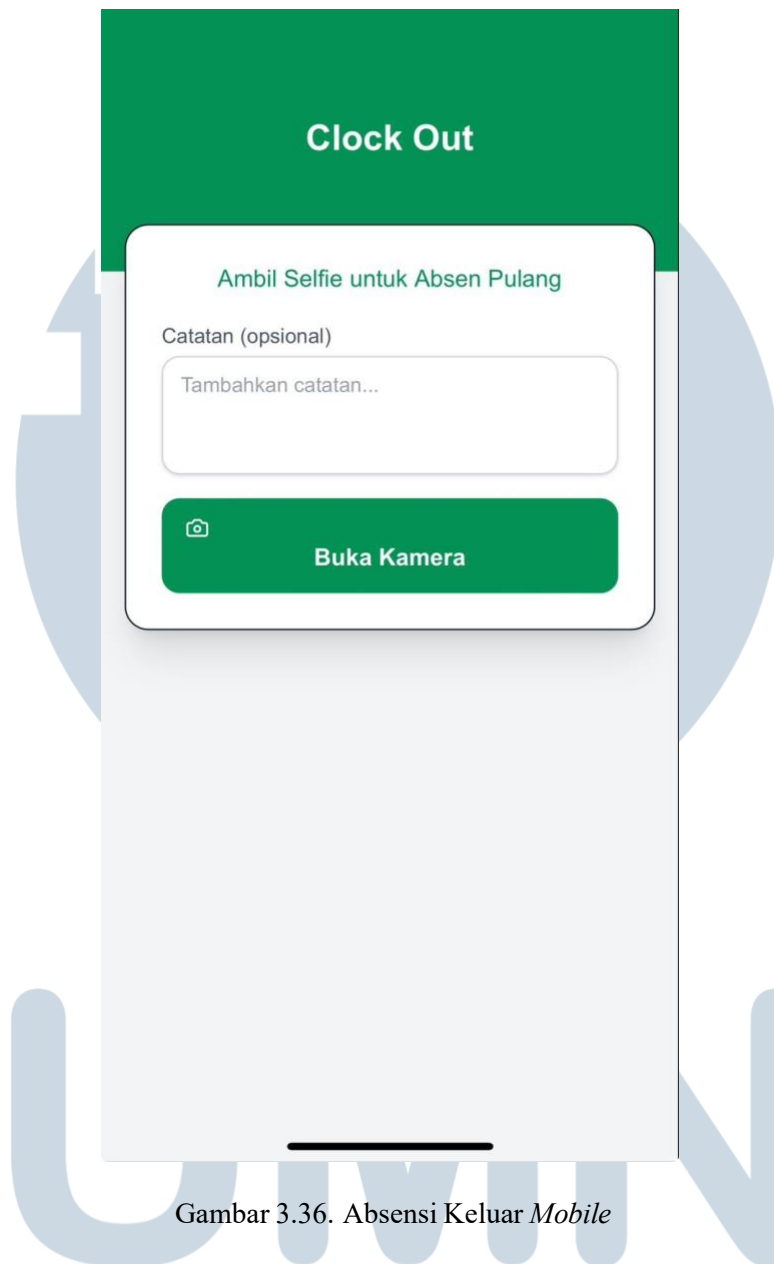
Gambar 3.34. Absensi *Mobile*

selanjutnya adalah halaman untuk presensi masuk jadi di saat *click* masuk akan muncul halaman seperti ini . Saat absen diwajibkan untuk foto terlebih dahulu dan bisa absen masuk Jika ingin menambahkan catatan itu opsional.



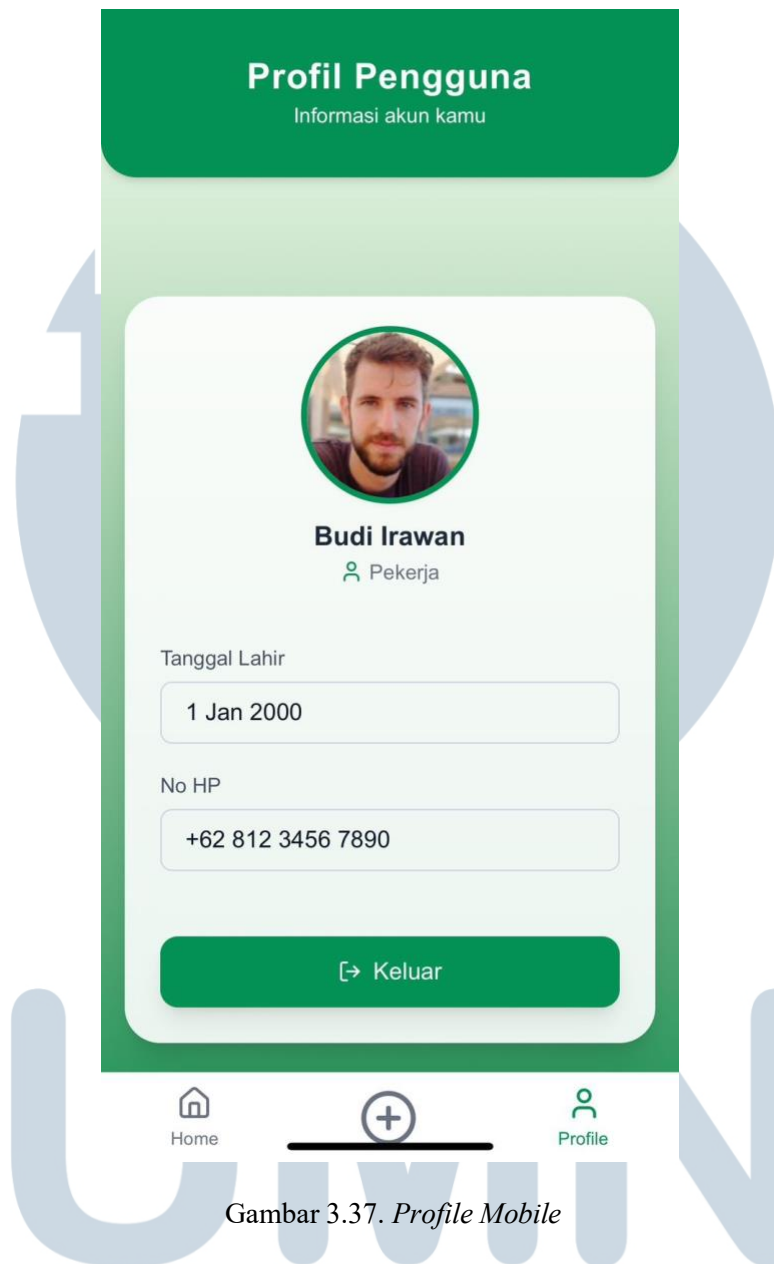
Gambar 3.35. Absensi Masuk *Mobile*

Selanjutnya adalah halaman ketika presensi untuk keluar jadi di saat *click* keluar akan muncul halaman seperti ini . Saat absen keluar diwajibkan untuk foto terlebih dahulu dan bisa absen keluar Jika ingin menambahkan catatan itu opsional.



Gambar 3.36. Absensi Keluar *Mobile*

Dan yang terakhir ada halaman untuk melihat *profile* pekerja yang isinya ada nomer *handphone*, tanggal lahir , nama , foto pekerja dan jabatan.



Gambar 3.37. *Profile Mobile*

3.5 Kendala dan Solusi yang Ditemukan

Selama pelaksanaan magang, kendala pertama yang dihadapi adalah mengalami kesulitan dalam memahami alur bisnis perusahaan, khususnya terkait proses absensi karyawan dan penjadwalan kerja yang berbeda-beda antar divisi. Setiap divisi memiliki kebutuhan dan pola kerja yang tidak selalu sama, sehingga cukup menyulitkan dalam menentukan struktur *database* dan alur sistem yang tepat. Selain itu ada kendala dalam merancang relasi antar tabel *database*, terutama dalam menentukan penggunaan *foreign key* dan *constraint* antara tabel *users*,

attendance, *schedule*, dan *shift*. Kesalahan dalam perancangan awal menyebabkan beberapa *query* tidak berjalan dengan optimal. Dan kendala terakhir ada perbedaan kebutuhan antara aplikasi *desktop* dan *mobile* menyebabkan penulis mengalami kendala dalam menyamakan struktur API. Beberapa fitur yang tersedia di *desktop* tidak sepenuhnya digunakan di *mobile*, sehingga perlu dilakukan penyesuaian pada *back-end*.

Sebagai solusi, melakukan beberapa diskusi dan koordinasi secara intensif dengan *supervisor* serta tim terkait. Penulis juga mempelajari dokumen kebutuhan sistem *shift(requirement)* dan melakukan penyesuaian ERD berdasarkan hasil diskusi. Dengan pendekatan ini, sistem yang dirancang dapat lebih sesuai dengan kebutuhan operasional perusahaan. lalu dengan melakukan evaluasi ulang terhadap ERD yang telah dibuat serta melakukan uji coba *query* secara bertahap. Penulis juga melakukan revisi struktur tabel berdasarkan masukan dari *supervisor*. Dengan perbaikan tersebut, relasi antar tabel menjadi lebih terstruktur dan *query* dapat berjalan dengan baik. Dan memisahkan beberapa *endpoint* yang bersifat spesifik untuk *mobile*, tanpa mengubah struktur utama *back-end desktop*. Dengan cara ini, *back-end* tetap terorganisir dan dapat melayani kebutuhan kedua platform dengan baik.

