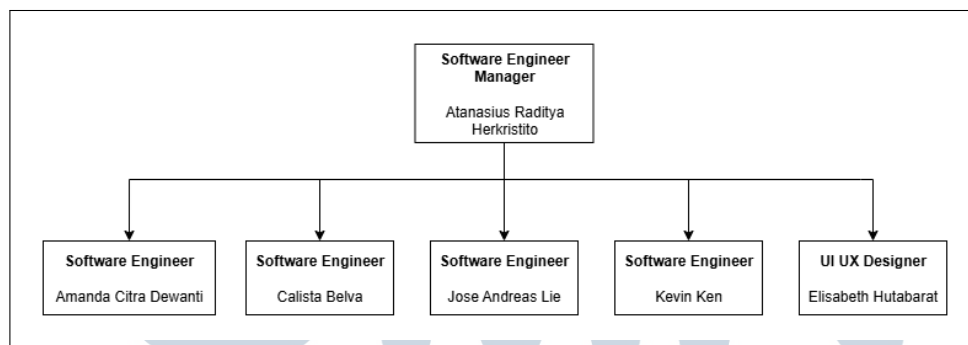


BAB 3

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Pelaksanaan kerja magang di PT Visi Karya Nusantara berlangsung dari tanggal 1 September 2025 hingga 1 Februari 2026, dengan struktur organisasi yang terlihat pada Gambar 3.1. Posisi yang ditempati selama kerja magang adalah *software engineer*, yang dipimpin dan dikoordinasi oleh Bapak Atanasius Raditya Herkristito selaku *senior software engineer & supervisor*. Senior software engineer memiliki tugas untuk mengawasi, membimbing, serta memberikan arahan selama proses kerja magang. Tim pengembang HRIS ini, melibatkan satu *senior software engineer*, satu *UI UX designer*, dan empat *software engineer*. *Software engineer* terdiri dari 2 *frontend engineer* dan 2 *backend engineer* dengan pembagian tugas yang berbeda. Koordinasi antar tim dilakukan secara *online* melalui aplikasi Discord. Setiap hari Kamis diadakan *weekly meeting* untuk membahas perkembangan dari pengerjaan HRIS, merencanakan tugas berikutnya, serta menerima *feedback* terkait hasil kerja tim.



Gambar 3.1. Struktur Organisasi Pelaksanaan Kerja Magang Product HRIS

Sumber: [10]

Dalam pelaksanaan praktek kerja magang, Git digunakan untuk mengontrol versi *project*. Setiap anggota tim memiliki *branch* masing-masing dalam pengerjaan sistem HRIS. Setiap *weekly meeting*, tim akan selalu mencatat *Minutes of Meeting* (MOM) yang berisi hasil diskusi hari ini serta rencana presentasi untuk minggu berikutnya. MOM tersebut kemudian dikirim melalui aplikasi Discord.

3.2 Tugas yang Dilakukan

Selama kerja magang, tugas utama yang dikerjakan adalah berfokus pada pengembangan modul *Daily Attendance* dan *Payroll* HRIS (*Human Resource Internal System*) berbasis web. Pengembangan dilakukan dengan menggunakan bahasa pemrograman Typescript dan SCSS. Selain itu, dilakukan pengujian secara manual terhadap sistem HRIS. Pengujian dilakukan secara manual dengan mengecek fitur-fitur pada halaman yang telah dikembangkan, serta mengevaluasi pengalaman *user* untuk memastikan aplikasi berjalan sesuai harapan. Pengujian ini mencakup pembuatan berbagai skenario pengujian yang bertujuan untuk menemukan *bug* atau masalah yang muncul selama pengembangan.

3.3 Uraian Pelaksanaan Magang

Selama pelaksanaan kerja magang, terdapat tugas-tugas yang dikerjakan dan dapat dilihat pada Tabel 3.1. Pelaksanaan kerja magang, dimulai dengan sesi *onboarding* pada minggu pertama. Sesi *onboarding* ini bertujuan untuk memperkenalkan struktur organisasi dan alur kerja perusahaan kepada karyawan baru. Setelah itu, *supervisor* memberikan tugas untuk mempelajari *boilerplate frontend* yang digunakan di perusahaan. *Boilerplate* adalah struktur kode awal yang berisi kumpulan kode pemrograman yang sering digunakan secara berulang dengan tujuan untuk mempermudah dan mempercepat proses pengembangan aplikasi.

Tabel 3.1. Tugas yang dilakukan setiap minggu selama pelaksanaan kerja magang

Minggu Ke -	Tugas yang dilakukan
1	Mengikuti <i>onboarding</i> dan mempelajari <i>boilerplate frontend</i> perusahaan
2	<i>Revamp table</i> pada halaman <i>User Management</i> , <i>Role Management</i> , <i>Project/Products Management</i> , <i>Leave Management</i> , <i>Leave Permit</i> , <i>Stand Up Feed</i> , <i>Activity Log</i> , <i>People Report</i> , <i>Reimbursement Management</i> , <i>Reimbursement Permit</i>
3	<i>Revamp</i> pada halaman <i>Role Detail</i> , <i>Project/Products Detail</i> , <i>Reimbursement Detail</i> dan mengerjakan halaman <i>User View</i> dan <i>Project/Products View</i>
4	Melakukan <i>testing</i> pada halaman <i>User View</i> dan <i>Project/Products View</i>
5	Mengerjakan halaman <i>Daily Attendance Management</i> , <i>Attendance Review</i> , dan <i>Daily Attendance</i>
6	Integrasi API pada halaman <i>Daily Attendance Management</i> , <i>Attendance Review</i> , dan <i>Daily Attendance</i>
7	Melakukan <i>testing</i> pada halaman <i>Daily Attendance Management</i> , <i>Attendance Review</i> , dan <i>Daily Attendance</i>
8	Integrasi API <i>pagination</i> pada halaman <i>Project/Products Management</i> , <i>Activity Log</i> , <i>Reimbursement Management</i> , <i>Reimbursement Permit</i>
9	Mengerjakan halaman <i>Payslip Management</i> , <i>Payslip Overview</i> , <i>Payslip Detail</i> , <i>Payslip Staff</i>
10	Integrasi API pada halaman <i>Payslip Management</i> , <i>Payslip Overview</i> , <i>Payslip Detail</i> , <i>Payslip Staff</i>
11	Melakukan <i>testing</i> pada halaman <i>Payslip Management</i> , <i>Payslip Overview</i> , <i>Payslip Detail</i> , <i>Payslip Staff</i>
12	Menambahkan fitur <i>disbursement</i> pada halaman <i>Payslip Detail</i>
13	Melakukan <i>testing</i> pada halaman <i>Payslip Detail</i>
14	Menambahkan kolom dan menyesuaikan alasan pada masing-masing karyawan sesuai dengan kontrak pada halaman <i>Attendance Review</i>
15	Mengerjakan halaman <i>Data Setup</i>

3.4 Perancangan Aplikasi

Subbab ini membahas perancangan HRIS yang mencakup penyusunan *User Requirement*, *Sitemap*, *Flowchart*, dan *Wireframe*. Perusahaan menggunakan metode *Component-Based Development*, yaitu metode pengembangan aplikasi yang tersusun dari komponen-komponen sederhana yang dapat digunakan kembali [11]. Penerapan konsep komponen ini memberikan pengaruh yang besar dalam proses pengembangan sistem, khususnya dalam efisiensi waktu dan biaya. Dengan durasi pengembangan yang lebih singkat, kebutuhan jam kerja yang diperlukan pengembang menjadi lebih rendah, sehingga biaya operasional tim pengembang lebih minim.

3.4.1 User Requirement

Sebelum memulai pengembangan, *supervisor* menjelaskan dan menyampaikan arahan terkait hal-hal yang perlu dikerjakan dalam pengembangan *Human Resource Internal System*, antara lain

1. Mengembangkan halaman *Daily Attendance Management* yang menampilkan daftar seluruh *daily attendance* karyawan. Pada halaman ini, terdapat tombol *Attendance Review* yang berfungsi untuk meninjau data kehadiran karyawan berdasarkan pada tanggal yang dipilih. Jika status *daily attendance* masih *Need Review*, maka akan muncul ikon berbentuk tempat sampah yang menandakan bahwa data tersebut masih dapat dihapus. Selain itu, terdapat opsi untuk melihat detail *daily attendance* pada tanggal tertentu dengan menekan ikon berbentuk mata. Halaman ini juga dilengkapi dengan fitur *search*, *sort*, dan filter untuk memudahkan pencarian data.
2. Mengembangkan halaman *Attendance Review* yang menampilkan data antara lain *Fullname*, *Job Title*, *Clock In*, *Clock Out*, *Lunch Start*, *Lunch End*, *Overtime*, *Status*, *Reason*, dan *Action*. Halaman ini berfungsi untuk melakukan input dan *submit* data kehadiran, serta untuk melihat detail alasan dari keterlambatan seperti *Clock In*, *early Clock Out*, dan keterlambatan saat *Lunch End*. Selain itu, *user* dapat melakukan input untuk alasan ketidakhadiran atau keterlambatan pada karyawan yang sedang ditinjau, seperti *Missing in Action (MIA)*, *late arrival (less than 2 hours)*, *late arrival (2+ hours, before lunch)*, dan *late arrival (after lunch break)*.

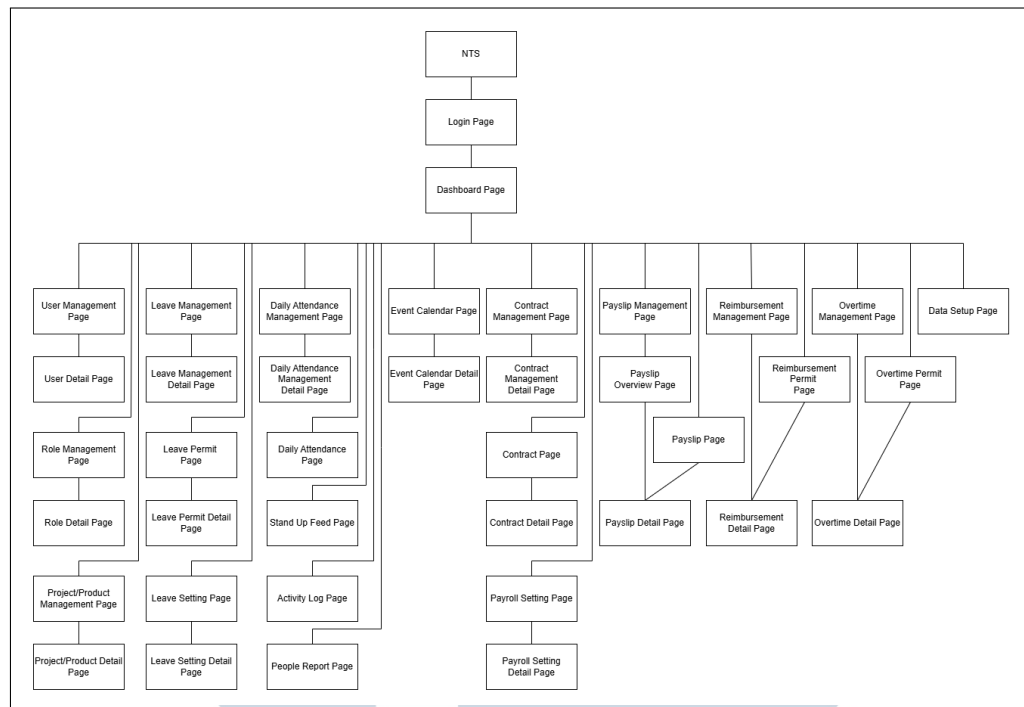
3. Mengembangkan halaman *Daily Attendance Staff* yang menampilkan seluruh kehadiran karyawan tersebut. Halaman ini menampilkan data antara lain *Date, Clock In, Clock Out, Lunch Start, Lunch End, Overtime, Status, Reason*, dan *Action*. Untuk menampilkan alasan keterlambatan, pulang lebih awal, atau keterlambatan setelah waktu makan siang secara detail, *user* dapat menekan ikon berbentuk mata yang terdapat pada kolom *Action*. Selain itu, halaman ini juga dilengkapi dengan fitur *search, sort*, dan filter untuk memudahkan pencarian data.
4. Mengembangkan halaman *Payslip Management* yang menampilkan daftar seluruh *payslip* yang telah digenerate. Pada halaman ini, terdapat tombol *Generate Payslip* yang berfungsi untuk meninjau data slip gaji karyawan berdasarkan pada tanggal yang dipilih. Jika status *payslip* masih *Pending*, maka akan muncul ikon berbentuk tempat sampah yang menandakan bahwa data tersebut masih dapat dihapus. Selain itu, terdapat opsi untuk melihat detail *payslip* pada tanggal tertentu dengan menekan ikon berbentuk mata, serta terdapat opsi untuk memperbarui *payslip* dengan menekan ikon berbentuk pensil. Halaman ini juga dilengkapi dengan fitur *search, sort*, dan filter untuk memudahkan pencarian data.
5. Mengembangkan halaman *Payslip Overview* yang menampilkan data antara lain *Salary Date, Employee Name, Job Title, Employment Status, Reviewed*, dan *Action*. Halaman ini berfungsi untuk menampilkan daftar karyawan yang menerima slip gaji pada tanggal yang telah di pilih sebelumnya pada halaman *Payslip Management*. Pada kolom *Reviewed* akan ditampilkan ikon berbentuk centang apabila *payslip* telah ditinjau. Untuk melakukan peninjauan, *user* dapat menekan ikon berbentuk catatan pada kolom *Action*.
6. Mengembangkan halaman *Payslip Detail* yang menampilkan informasi pribadi karyawan antara lain *Name, NIK, Contact, Job Title, Employment Status, Attendance, Income Name, Income Value, Deduction Name, Deduction Value, Take Home Pay, Bank Name, Bank Account Number, Bank Account Name, Salary Slip Date*. Pada halaman ini, dalam *mode edit*, *user* tertentu yang memiliki hak akses dapat menambahkan *item disbursement* yaitu *overtime* dan atau *reimbursement* dengan menekan tombol *Add Item*, kemudian dapat menambahkan *income* dan *deduction* dengan menekan tombol *Add Row*. Sementara itu, jika dalam *mode view only*, halaman hanya

akan menampilkan informasi karyawan saja tanpa dapat diedit.

7. Mengembangkan halaman *Payslip List Staff* yang menampilkan seluruh *payslip* karyawan tersebut. Halaman ini menampilkan data antara lain *Start Date*, *End Date*, *Employee Name*, *Job Title*, *Employment Status*, dan *Action*. Untuk menampilkan slip gaji secara detail, *user* dapat menekan ikon berbentuk mata yang terdapat pada kolom *Action*. Selain itu, halaman ini juga dilengkapi dengan fitur *search*, dan *sort*.

3.4.2 Sitemap

Gambar 3.2 adalah gambar *sitemap* dari *Human Resource Internal System*. Berdasarkan *sitemap* tersebut, halaman pertama yang akan terlihat oleh *user* adalah halaman *login*. Ketika *user* berhasil melakukan *login*, maka *user* akan diarahkan ke halaman *Dashboard*. Selanjutnya *user* dapat mengakses halaman-halaman berikutnya, antara lain *User Management* yang dapat diarahkan ke halaman *User Detail*, *Role Management* yang dapat diarahkan ke halaman *Role Detail*, *Project/Products* yang dapat diarahkan ke halaman *Project/Products Detail*, *Leave Management* yang dapat diarahkan ke halaman *Leave Management Detail*, *Leave Permit* yang dapat diarahkan ke halaman *Leave Permit Detail*, *Leave Setting* yang dapat diarahkan ke halaman *Leave Setting Detail*, *Daily Attendance Management* yang dapat diarahkan ke halaman *Attendance Review*, *Daily Attendance*, *Stand Up Feed*, *Event Calendar* yang dapat diarahkan ke halaman *Event Calendar Detail*, *Contract Management* yang dapat diarahkan ke halaman *Contract Management Detail*, *Contract* yang dapat diarahkan ke halaman *Contract Detail*, *Payslip Management* yang dapat diarahkan ke halaman *Payslip Overview* kemudian dapat diarahkan ke halaman *Payslip Detail*, *Payslip* yang dapat diarahkan ke halaman *Payslip Detail*, *Payroll Setting* yang dapat diarahkan ke halaman *Payroll Setting Detail*, *Activity Log*, *People Report*, *Reimbursement Management* yang dapat diarahkan ke halaman *Reimbursement Detail*, *Reimbursement Permit* yang dapat diarahkan ke halaman *Reimbursement Detail*, *Data Setup*, *Overtime Management* yang dapat diarahkan ke halaman *Overtime Detail*, dan *Overtime Permit* yang dapat diarahkan ke halaman *Overtime Detail*.



Gambar 3.2. Sitemap Human Resource Internal System

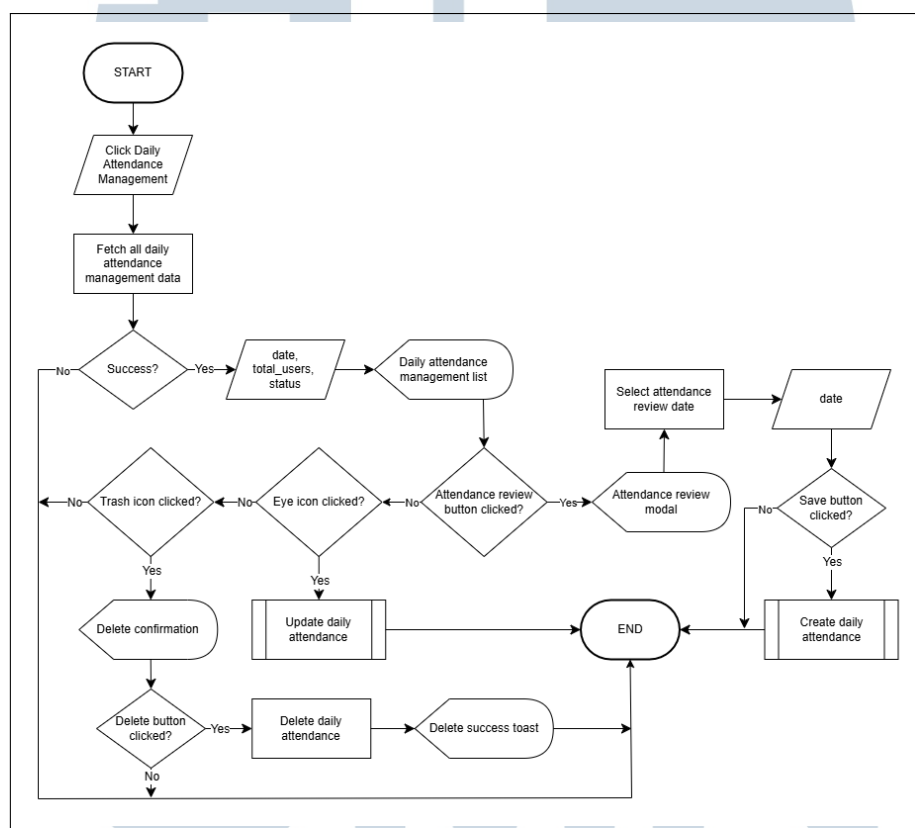
3.4.3 Flowchart

Berikut ditampilkan *flowchart* untuk menggambarkan alur antarmuka dari sisi user.

A Flowchart Daily Attendance Management

Gambar 3.3 adalah gambar *flowchart* pada halaman *Daily Attendance Management*. Proses akan dimulai dari pemanggilan GET API ke *endpoint* `"/api/v1/attendance-dates/"`. Jika permintaan GET API berhasil dengan status *code* 200 dan *message success*, maka seluruh daftar *daily attendance management* akan ditampilkan yang ditunjukkan pada Gambar 3.30. Selanjutnya, *user* yang memiliki akses dapat memilih untuk menambahkan *daily attendance review* baru dengan menekan tombol *Attendance Review* yang akan menampilkan modal *Attendance Review* untuk memilih tanggal atau *date*. Setelah itu dilanjutkan dengan menekan tombol *Save* yang akan melakukan pemanggilan POST API ke *endpoint* `"/api/v1/attendance-dates/"`. Jika permintaan POST API berhasil dengan status *code* 200 dan *message success*, maka *daily attendance* pada tanggal yang telah dipilih berhasil dibuat yang ditunjukkan pada Gambar 3.31. User dapat melihat

secara detail dan memperbarui *daily attendance* yang dipilih dengan menekan ikon berbentuk mata dan akan melakukan pemanggilan GET API ke *endpoint* `"/api/v1/daily-attendances/detail/{attendance_date_id}"` dan diarahkan ke halaman *Attendance Review*. Selain itu, user juga dapat menghapus *daily attendance* yang dipilih dengan menekan ikon berbentuk tempat sampah dan akan menampilkan modal konfirmasi dan dapat dilanjutkan dengan menekan tombol *Delete* yang akan melakukan pemanggilan DELETE API ke *endpoint* `"/api/v1/attendance-dates/{id}"`.

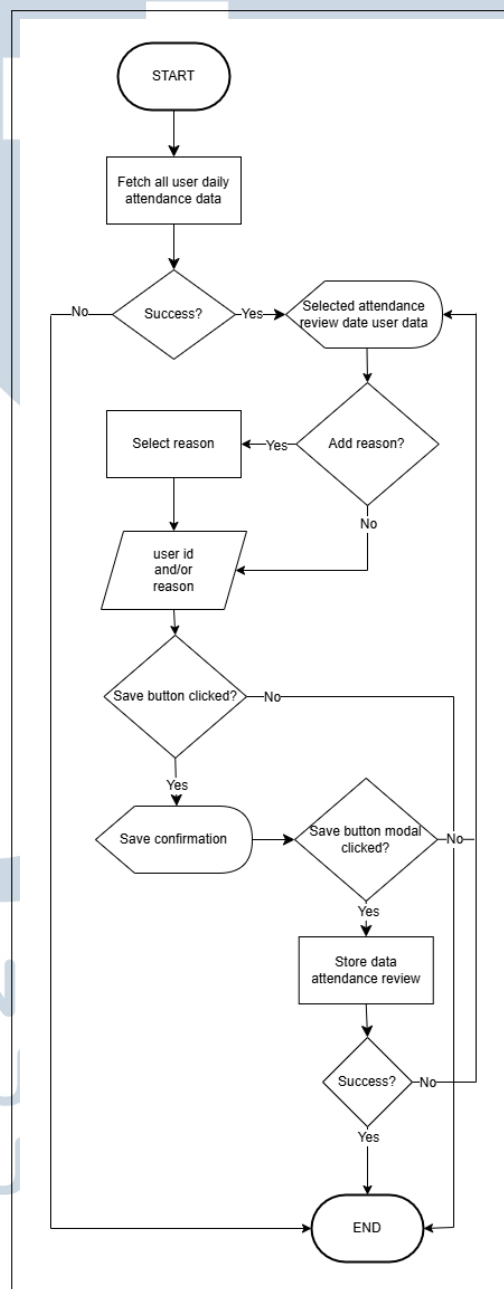


Gambar 3.3. Flowchart Daily Attendance Management

B Flowchart Create atau Update Daily Attendance Management

Gambar 3.4 adalah gambar *flowchart* pada halaman *Attendance Review*. Proses akan dimulai dari pemanggilan GET API ke *endpoint* `"/api/v1/daily-attendances/detail/{attendance_date_id}"` untuk menampilkan seluruh data *daily attendance* pada tanggal yang telah dipilih. Jika permintaan GET API berhasil dengan status *code* 200 dan *message success*, maka seluruh daftar *daily attendance* akan ditampilkan yang ditunjukkan pada Gambar 3.31. Jika *daily attendance*

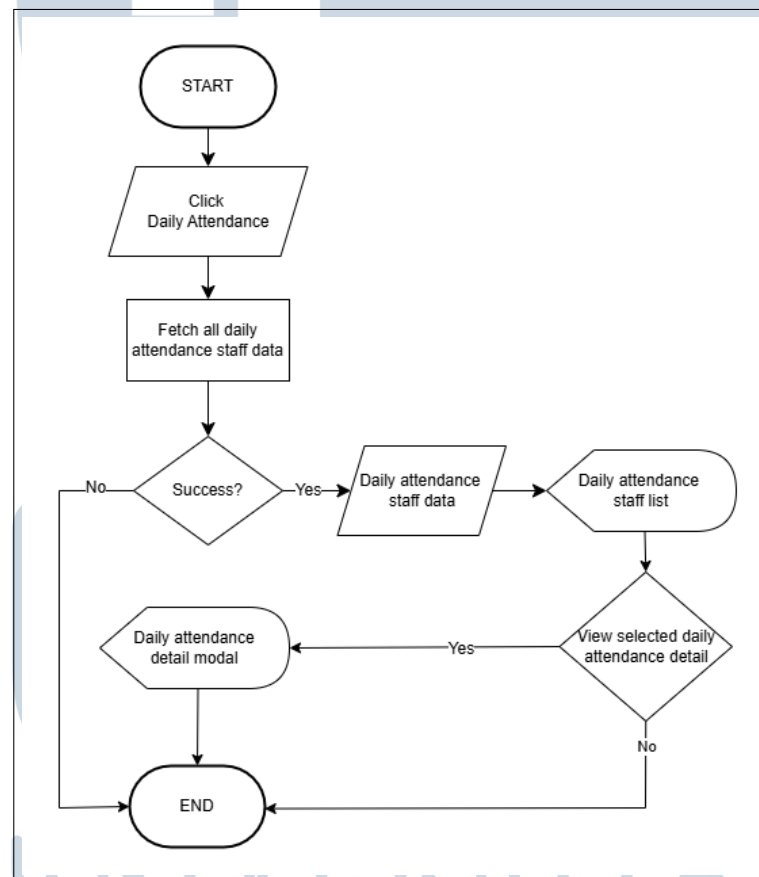
memiliki *reason late* atau alasan keterlambatan, *user* dapat melihat secara detail *daily attendance* yang dipilih dengan menekan ikon berbentuk mata dan akan menampilkan modal *Attendance Details*. *User* juga dapat memilih *reason* yang bersifat optional untuk dilakukan *review* terhadap *daily attendance* masing-masing karyawan. Jika sudah selesai melakukan *review*, maka dapat dilanjutkan dengan menekan tombol *Save* dan akan melakukan pemanggilan POST API ke *endpoint* `"/api/v1/daily-attendances/detail/approve/{attendance_date_id}"`.



Gambar 3.4. Flowchart Create atau Update Daily Attendance

C Flowchart Daily Attendance Staff

Gambar 3.5 adalah gambar *flowchart* pada halaman *Daily Attendance*. Proses akan dimulai dari pemanggilan GET API ke *endpoint* `"/api/v1/daily-attendances/detail/user"`. Jika permintaan GET API berhasil dengan status *code* 200 dan *message success*, maka seluruh daftar *daily attendance* akan ditampilkan yang ditunjukkan pada Gambar 3.32. Selanjutnya, jika *user* memiliki *reason late* atau alasan keterlambatan, *user* dapat melihat secara detail *daily attendance* yang dipilih dengan menekan ikon berbentuk mata dan akan menampilkan modal *Attendance Details*.

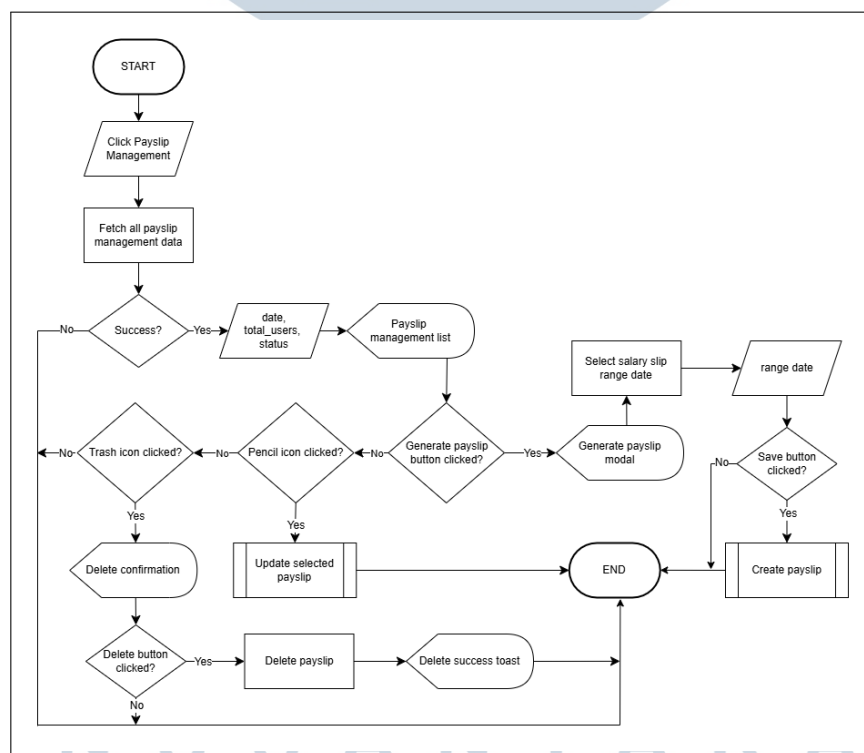


Gambar 3.5. Flowchart Daily Attendance Staff

D Flowchart Payslip Management

Gambar 3.6 adalah gambar *flowchart* pada halaman *Payslip Management*. Proses akan dimulai dari pemanggilan GET API ke *endpoint* `"/api/v1/payslip"`. Jika permintaan GET API berhasil dengan status *code* 200 dan *message success*,

maka seluruh daftar *payslip management* akan ditampilkan yang ditunjukkan pada Gambar 3.33. Selanjutnya, *user* dapat memilih untuk menambahkan *payslip* baru dengan menekan tombol *Generate Payslip* yang akan menampilkan modal *generate payslip* untuk memilih rentang tanggal atau *range salary date*. Setelah itu dilanjutkan dengan menekan tombol *Save* yang akan melakukan pemanggilan POST API ke *endpoint* `"/api/v1/payslip/date/"`. Jika permintaan POST API berhasil dengan status *code* 200 dan *message success*, maka *payslip* pada rentang tanggal yang telah dipilih berhasil dibuat yang ditunjukkan pada Gambar 3.34. *User* dapat melihat secara detail *payslip* yang dipilih dengan menekan ikon berbentuk mata atau menekan ikon berbentuk pensil untuk memperbarui dan akan melakukan pemanggilan GET API ke *endpoint* `"/api/v1/payslip/date/payslip_date_id"` dan diarahkan ke halaman *Payslip Overview*. Selain itu, *user* juga dapat menghapus *payslip* yang dipilih dengan menekan ikon berbentuk tempat sampah dan akan menampilkan modal konfirmasi dan dapat dilanjutkan dengan menekan tombol *Delete* yang akan melakukan pemanggilan DELETE API ke *endpoint* `"/api/v1/payslip/date/{payslip_date_id}"`.

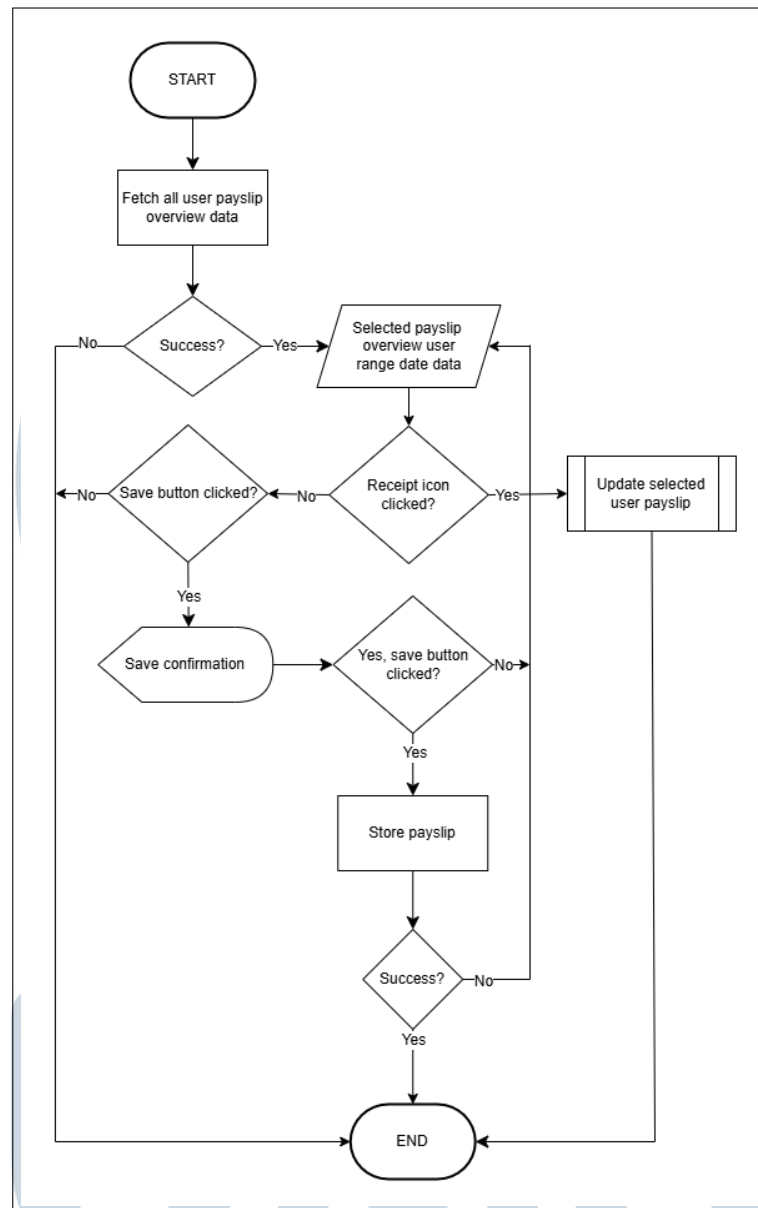


Gambar 3.6. Flowchart Payslip Management

E Flowchart Payslip Overview

Gambar 3.7 adalah *flowchart* pada halaman *Payslip Overview*. Proses akan dimulai dari ditampilkannya *general information* yaitu *salary date* dan *employee list* dengan menampilkan sebuah tabel yang memiliki kolom penomoran, *Employee Name*, *Job Title*, *Employment Status*, *Reviewed*, dan *Action* dari pemanggilan GET API ke *endpoint* `"/api/v1/payslip/date/{payslip_date_id}"`. Pada kolom *Employee Name* akan ditampilkan seluruh nama karyawan yang telah disetujui kehadirannya pada tanggal yang telah dipilih, masing-masing karyawan memiliki posisi yang berbeda dan dapat dilihat pada kolom *Job Title*, serta keterangan status karyawan pada kolom *Employment Status*. Jika *payslip* karyawan telah ditinjau, maka akan ditampilkan ikon berbentuk centang pada kolom *Reviewed*. Peninjauan masing-masing *payslip* karyawan dapat dilakukan dengan menekan ikon yang berbentuk catatan pada kolom *Action* yang dipilih. Jika user telah melakukan peninjauan pada masing-masing *payslip*, dilanjutkan dengan menekan tombol *Save* untuk mengirimkan data *payslip* tersebut kepada masing-masing *staff* yang akan melakukan pemanggilan PATCH API ke *endpoint* `"/payslip/date/{payslip_date_id}"`.





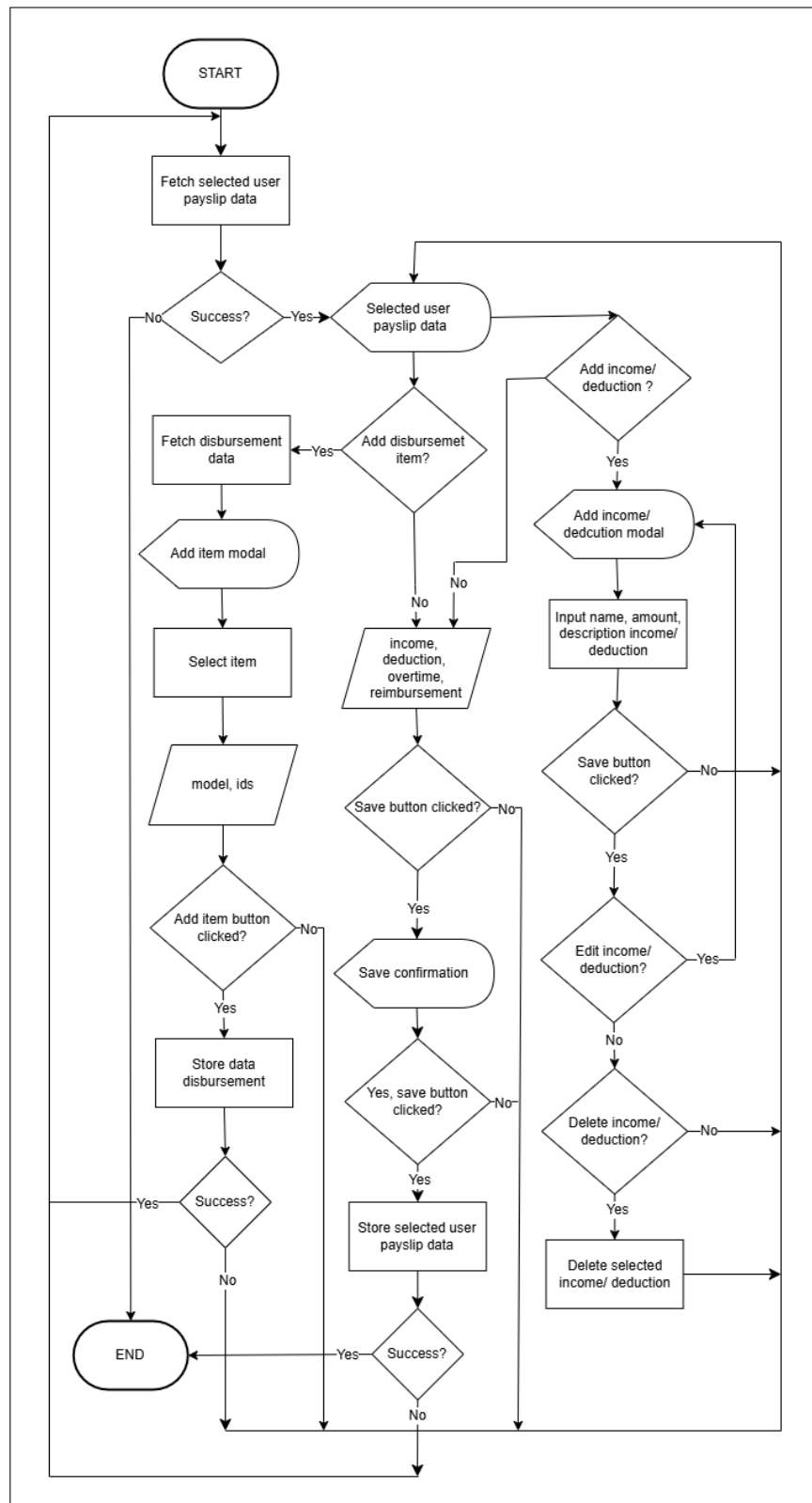
Gambar 3.7. Flowchart Payslip Overview

F Flowchart Update Selected User Payslip

Gambar 3.8 adalah gambar *flowchart* pada halaman *Payslip Detail*. Proses akan dimulai dari pemanggilan GET API ke *endpoint* `"/api/v1/payslip/date/{payslip_date_id}/contract/{contract_id}"`. Jika *user* menekan ikon berbentuk mata pada halaman *Payslip Management*, maka pada halaman *Payslip Detail* akan dimulai dari pemanggilan GET API ke *endpoint* `"/payslip/user/{payslip_id}"`. Jika permintaan GET API berhasil dengan status

code 200 dan *message success*, maka seluruh data *payslip* untuk staff tersebut akan ditampilkan yang ditunjukkan pada Gambar 3.37. Selanjutnya, *user* dapat menambahkan *income* dan atau *deduction* dengan menekan *Add Row* pada bagian bawah masing-masing tabel *income* atau *deduction*. Jika *payslip* sudah sesuai, maka *user* dapat menekan tombol *Save* untuk memperbarui *payslip* untuk *staff* tersebut yang akan melakukan pemanggilan PATCH API ke *endpoint* `"/payslip/date/{payslip_date_id}/contract/{contract_id}"`, tampilan ditunjukkan pada Gambar 3.35.

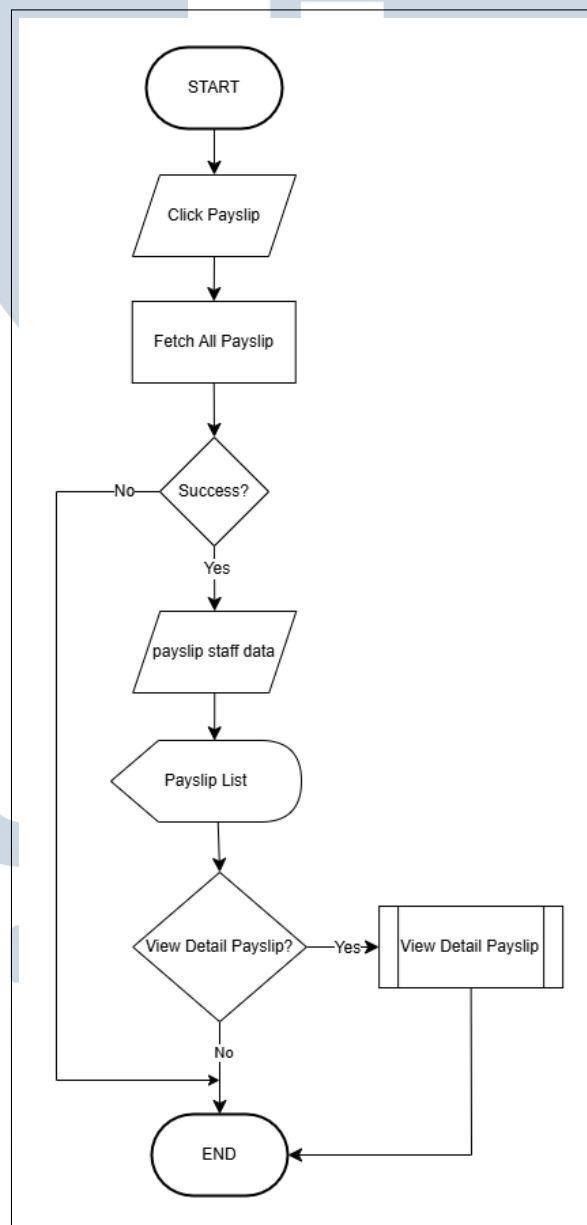




Gambar 3.8. Flowchart Update Selected User Payslip

G Flowchart User Payslip List

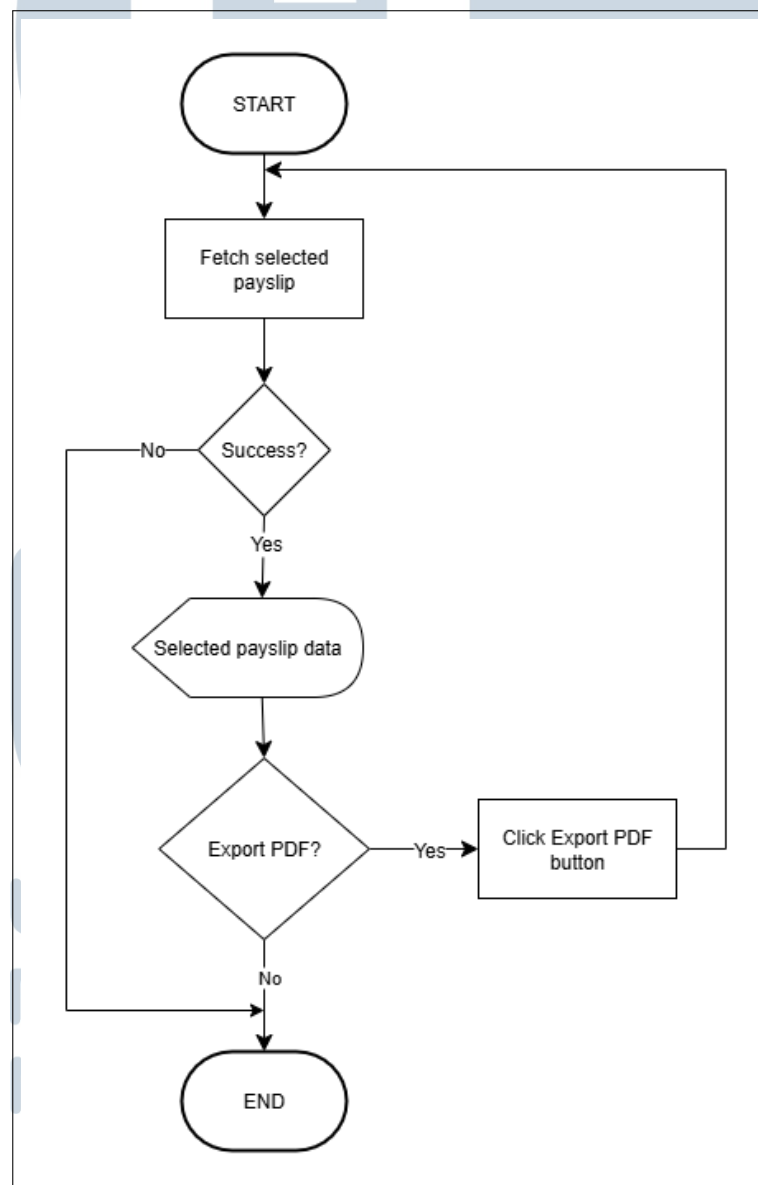
Gambar 3.9 adalah gambar *flowchart* pada halaman *User Payslip List*. Proses akan dimulai dari pemanggilan GET API ke *endpoint* `"/api/v1/payslip/user"`. Jika permintaan GET API berhasil dengan *status code* 200 dan *message success*, maka seluruh daftar *payslip user* akan ditampilkan yang ditunjukkan pada Gambar 3.36. Selanjutnya, *user* dapat melihat secara detail *payslip* yang dipilih dengan menekan ikon berbentuk mata dan akan diarahkan ke halaman *Payslip Detail*.



Gambar 3.9. Flowchart User Payslip

H Flowchart Selected User Payslip Detail

Gambar 3.10 adalah gambar *flowchart* pada halaman *Payslip Detail*. Proses akan dimulai dari pemanggilan GET API ke *endpoint* `"/api/v1/payslip/user/{payslip_id}"`. Jika permintaan GET API berhasil dengan *status code* 200 dan *message success*, maka seluruh data *payslip* yang dipilih akan ditampilkan yang ditunjukkan pada Gambar 3.37. Selanjutnya, *user* dapat mengonversi *payslip* tersebut ke dalam bentuk PDF dengan menekan tombol *Export PDF*.

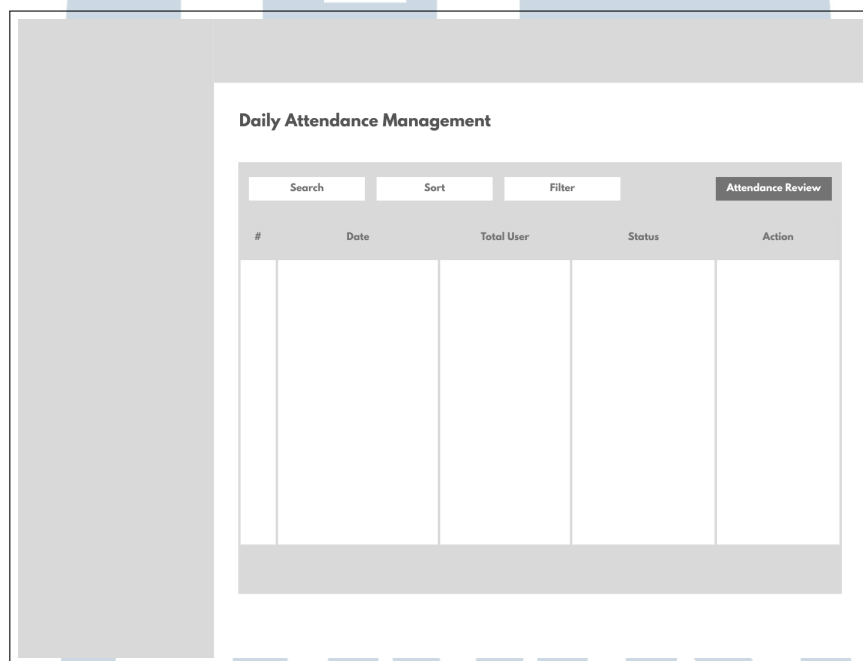


Gambar 3.10. Flowchart Selected User Payslip Detail

3.4.4 Wireframe

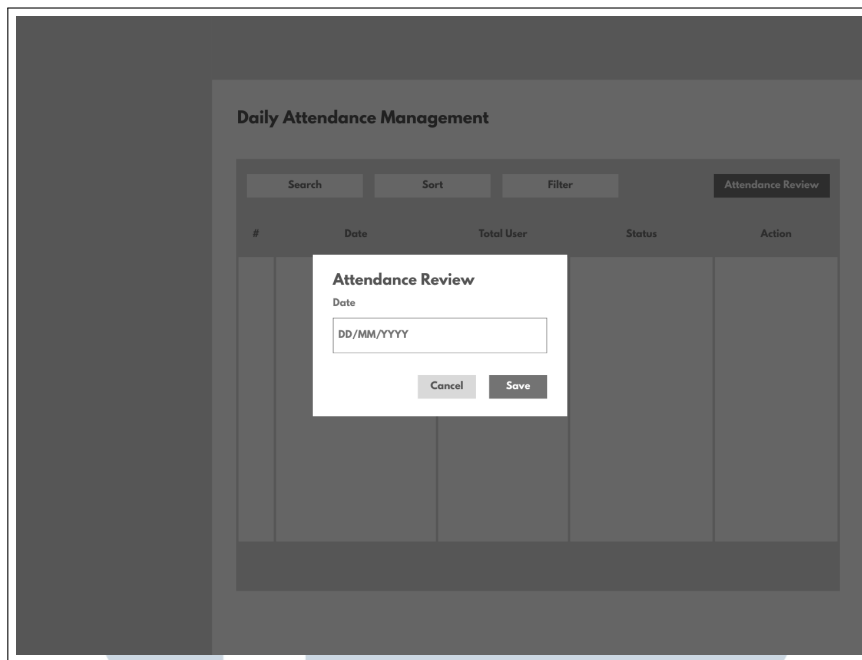
Wireframe adalah kerangka dasar yang dirancang untuk merepresentasikan gambaran yang sederhana dari antarmuka aplikasi. Berikut ini akan ditampilkan gambaran awal HRIS pada modul *Daily Attendance* dan *Payroll*.

Gambar 3.11 adalah gambar *wireframe* pada halaman *Daily Attendance Management*. Tampilan ini menampilkan seluruh *daily attendance management*. *User* dapat menambahkan *daily attendance* baru dengan menekan tombol *Attendance Review*. Pada halaman ini didukung dengan fitur *search*, *sort*, dan *filter*.



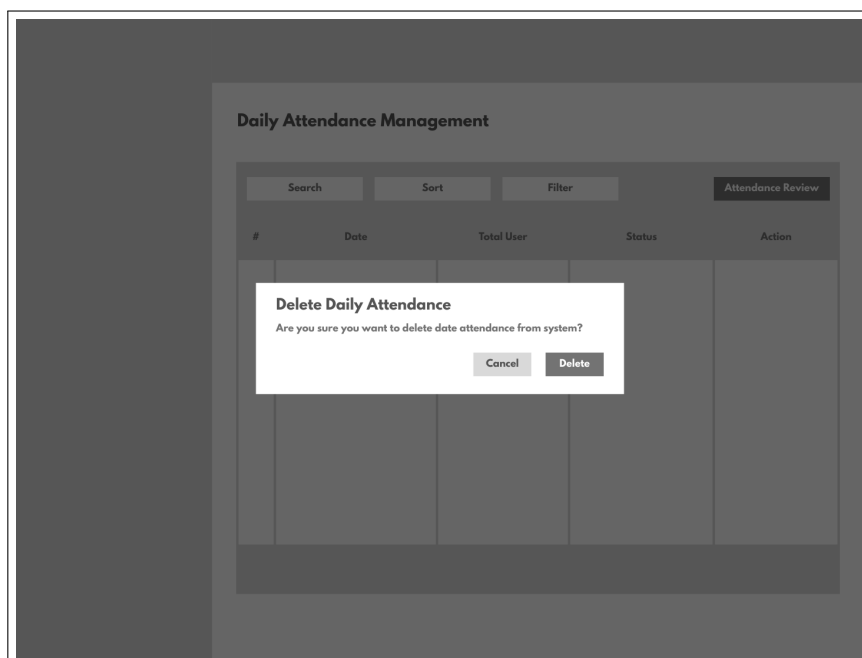
Gambar 3.11. Wireframe Daily Attendance Management

Gambar 3.12 adalah gambar *wireframe* modal ketika *user* menekan tombol *attendance review*. *User* dapat memilih tanggal *attendance* yang akan ditinjau pada *form field*.



Gambar 3.12. Wireframe Modal Add Daily Attendance

Gambar 3.13 adalah gambar *wireframe* modal konfirmasi *delete daily attendance* ketika *user* menekan simbol berbentuk tempat sampah.

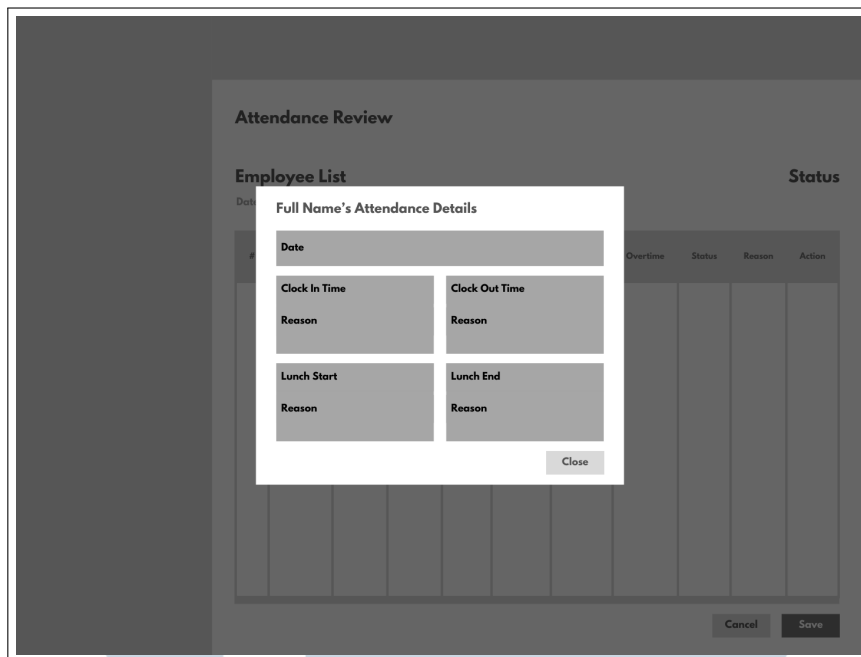


Gambar 3.13. Wireframe Modal Delete Daily Attendance

Gambar 3.14 adalah gambar *wireframe* pada halaman *Attendance Review*.

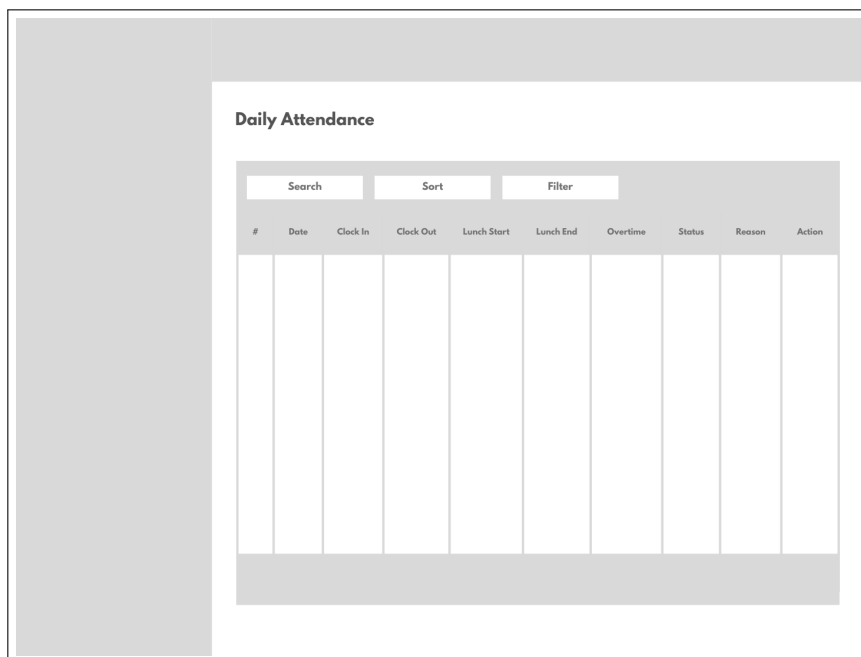
[illegible]

modal 3.15 adalah gambar *wireframe* modal ketika user menekan mata. Modal ini menampilkan alasan-alasan *Staff* yang dipilih.



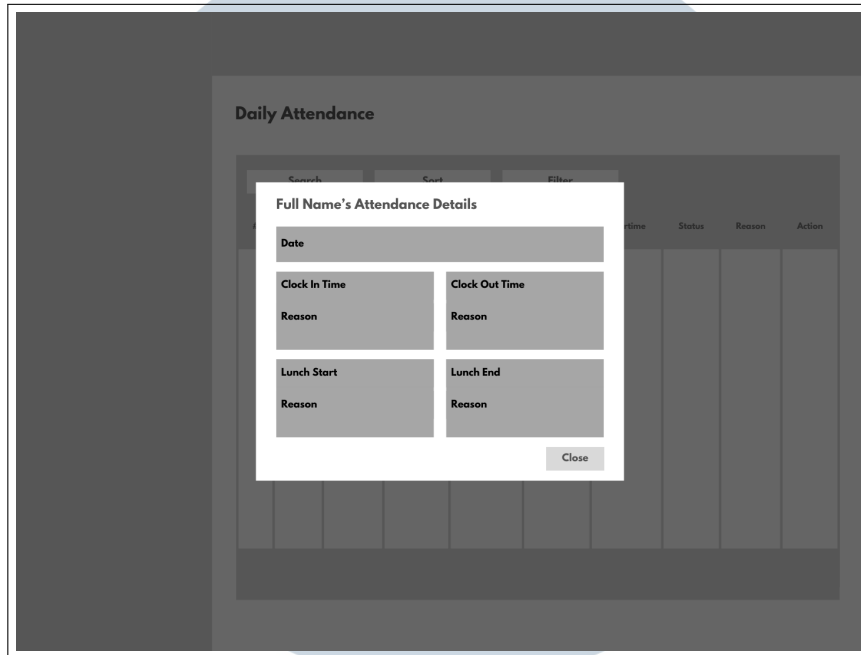
Gambar 3.15. Wireframe Modal Detail Attendance Review

Gambar 3.16 adalah gambar *wireframe* pada halaman *Daily Attendance Staff*. Halaman ini menampilkan seluruh data *daily attendance staff* tersebut. Pada halaman ini didukung dengan fitur *search*, *sort*, dan *filter*.



Gambar 3.16. Wireframe Daily Attendance Staff

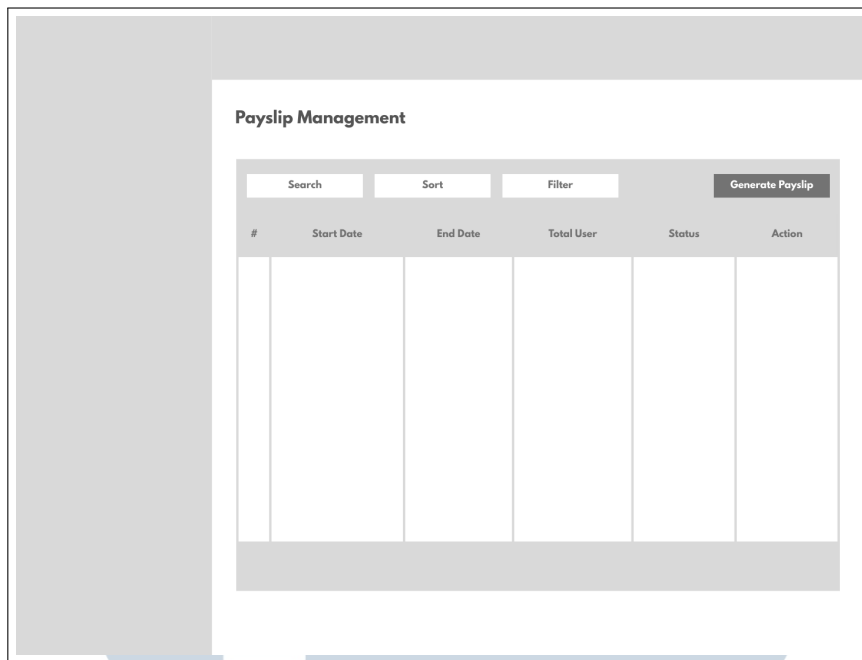
Gambar 3.17 adalah gambar *wireframe* modal ketika *staff* menekan simbol berbentuk mata. Modal ini menampilkan alasan-alasan pada *daily attendance* yang dipilih.



Gambar 3.17. Wireframe Modal Detail Daily Attendance Staff

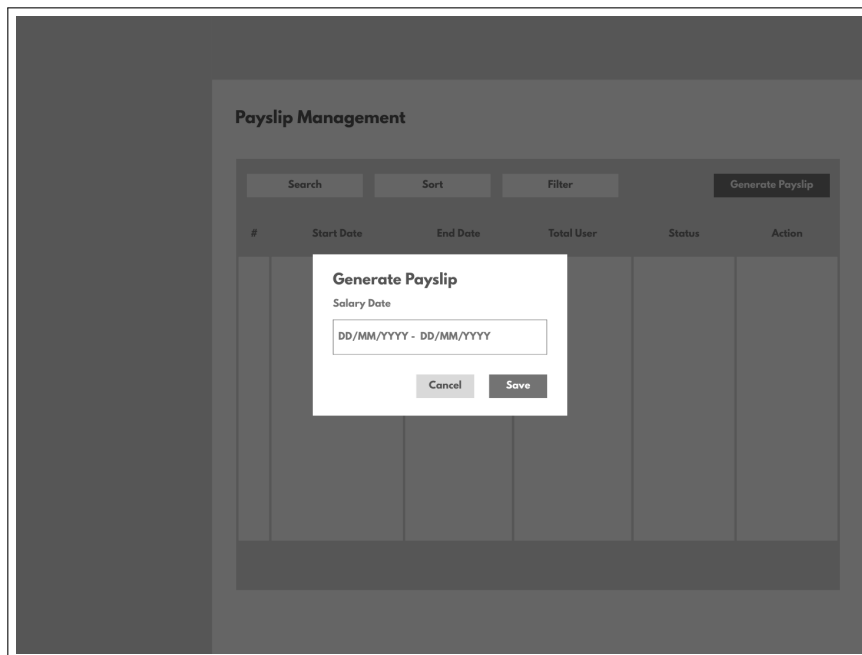
Gambar 3.18 adalah gambar *wireframe* pada halaman *Payslip Management*. Tampilan ini menampilkan seluruh *payslip management*. User dapat menambahkan *payslip* baru dengan menekan tombol *Generate Payslip*. Pada halaman ini didukung dengan fitur *search*, *sort*, dan *filter*.

UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



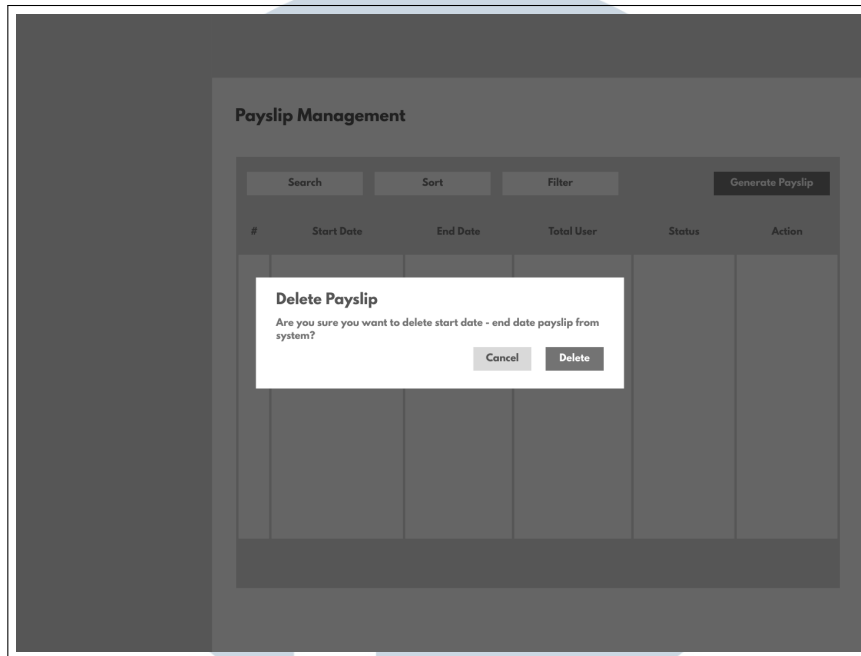
Gambar 3.18. Wireframe Payslip Management

Gambar 3.19 adalah gambar *wireframe* modal ketika *user* menekan tombol *Generate Payslip*. *User* dapat memilih rentang tanggal *salary slip* yang akan ditinjau pada *form field*.



Gambar 3.19. Wireframe Modal Add Payslip

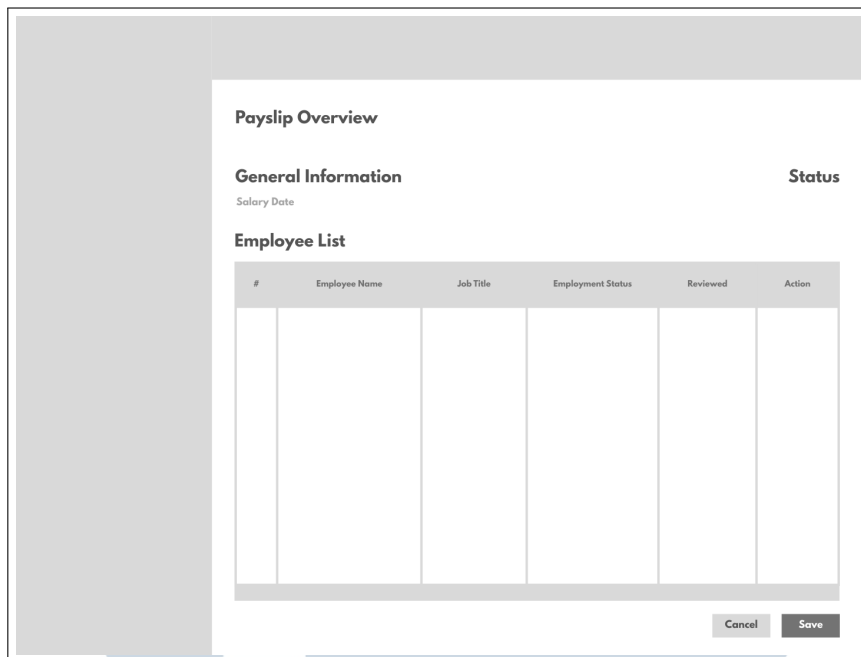
Gambar 3.20 adalah gambar *wireframe* modal konfirmasi *delete payslip* ketika *user* menekan simbol berbentuk tempat sampah.



Gambar 3.20. Wireframe Modal Delete Payslip

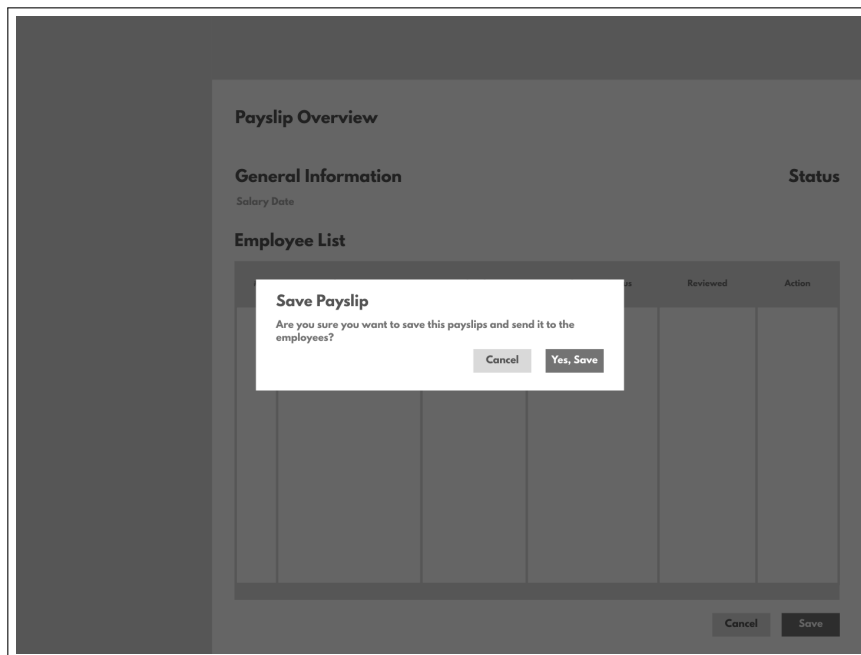
Gambar 3.21 adalah gambar *wireframe* pada halaman *Payslip Overview*. Halaman ini menampilkan tanggal *payslip* yang dipilih. Selain itu, halaman ini menampilkan data seluruh *staff*. Kemudian *user* dapat melihat secara detail *payslip* masing-masing *staff* dengan menekan simbol berbentuk nota pada kolom *action*.

UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.21. Wireframe Payslip Overview

Gambar 3.22 adalah gambar *wireframe* modal konfirmasi ketika *user* menekan tombol *save*. Jika *user* menekan tombol *Yes, Save* maka *payslip* yang telah ditinjau akan dikirimkan kepada masing-masing *staff*.



Gambar 3.22. Wireframe Save Payslip Overview

Gambar 3.23 adalah gambar *wireframe* pada halaman *Payslip Detail* pada *payslip* yang telah dipilih. *User* dapat menambahkan *item disbursement* dengan menekan tombol *Add Item*. Kemudian, untuk menambahkan *income* atau *deduction*, *user* dapat menekan *Add Row*.

Payslip

Employee Name Payslip Add Item

Salary Slip Start Date - End Date

Name : Employee Name
 NIK : xxxxxxxxxxxxxxxx
 Contact : xxxxxxxxxxxxxxxx

Job Title : Employee Name
 Employment Status : xxxxxxxxxxxxxxxx
 Attendance : xxxxxxxxxxxxxxxx

Income	Value (Rp)	Action

Add Row

Deduction	Value (Rp)	Action

Add Row

Take Home Pay Rp Value

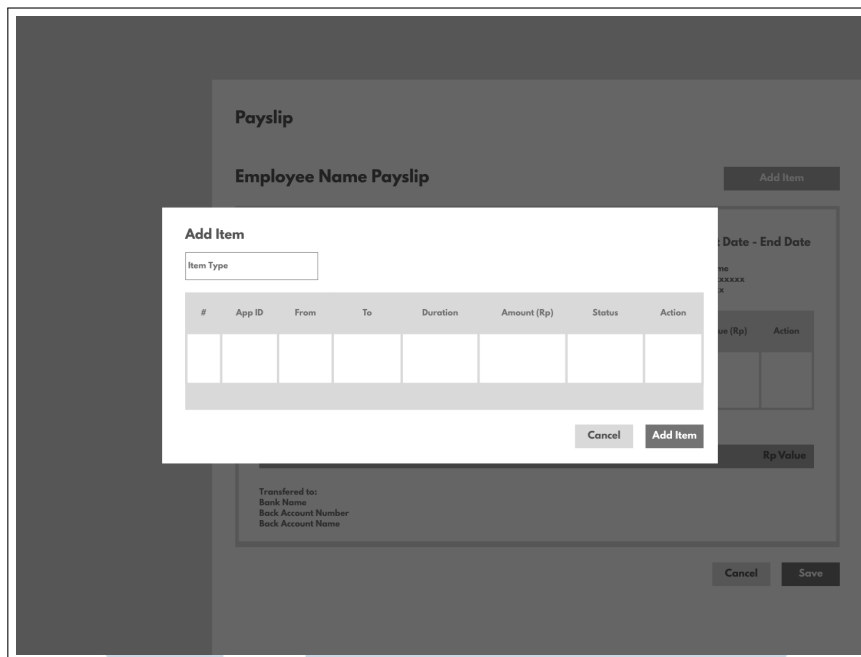
Transferred to:
 Bank Name
 Bank Account Number
 Bank Account Name

Cancel Save

Gambar 3.23. Wireframe Selected Payslip Detail

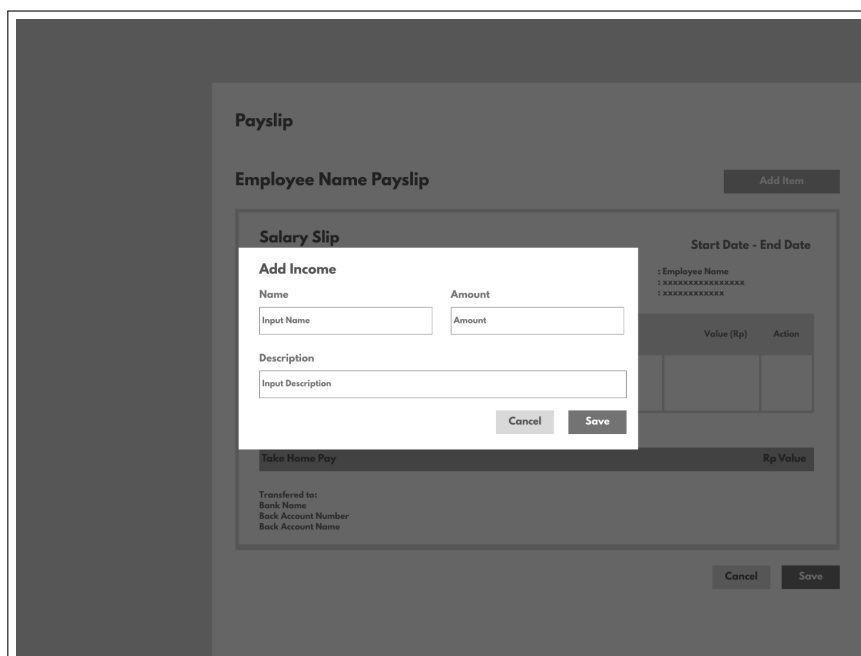
Gambar 3.24 adalah gambar *wireframe* modal *Add Item*. *User* dapat memilih jenis *item* yang akan dicairkan. Terdapat 2 jenis *item*, yaitu *overtime* dan *reimbursement*. Setelah memilih jenis *item*, *user* dapat memilih daftar mana yang akan dicairkan dengan mencentang pada kolom #.

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 3.24. Wireframe Modal Add Item

Gambar 3.25 adalah gambar *wireframe* modal *Add Income*. User dapat mengisi *name*, *amount*, dan *description* pada *form field*.



Gambar 3.25. Wireframe Modal Add Income

Gambar 3.26 adalah gambar *wireframe* modal *Add Deduction*. User dapat

mengisi *name*, *amount*, dan *description* pada *form field*.

The image shows a wireframe of a web application interface. At the top, there's a header 'Payslip'. Below it, a section 'Employee Name Payslip' contains an 'Add Item' button. A modal titled 'Salary Slip' is open, featuring an 'Add Deduction' form. This form has three input fields: 'Name' (with a placeholder 'Input Name'), 'Amount' (with a placeholder 'Amount'), and 'Description' (with a placeholder 'Input Description'). Below these fields are 'Cancel' and 'Save' buttons. The background interface also shows a table with columns 'Value (Rp)' and 'Action', and a 'Take Home Pay' section with a 'Rp Value' field. At the bottom right of the background interface are 'Cancel' and 'Save' buttons.

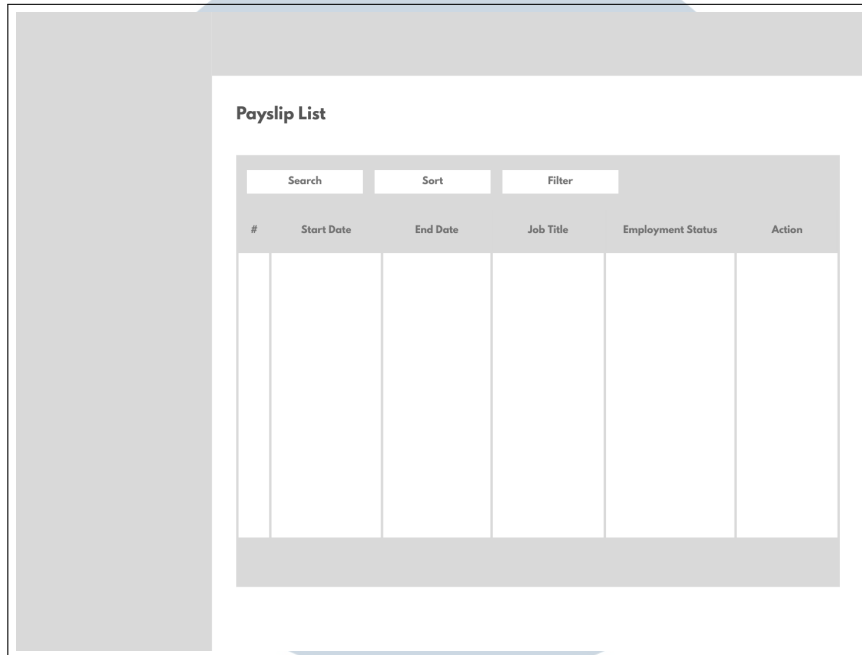
Gambar 3.26. Wireframe Modal Add Deduction

Gambar 3.27 adalah gambar *wireframe* modal konfirmasi *save payslip* pada *payslip* yang telah dipilih dan dilakukan tinjauan.

The image shows a wireframe of a web application interface, similar to the previous one. A modal titled 'Save Payslip' is open, displaying the message: 'Are you sure you want to save Employee Name's payslip into the system?'. Below the message are 'Cancel' and 'Yes, Save' buttons. The background interface shows the 'Employee Name Payslip' section with an 'Add Item' button, a table with 'Value (Rp)' and 'Action' columns, and a 'Take Home Pay' section with a 'Rp Value' field. At the bottom right of the background interface are 'Cancel' and 'Save' buttons.

Gambar 3.27. Wireframe Modal Save Selected Payslip

Gambar 3.28 adalah gambar *wireframe* pada halaman *Payslip List Staff*. Halaman ini menampilkan seluruh data *payslip staff* tersebut. Pada halaman ini didukung dengan fitur *search*, *sort*, dan *filter*.



Gambar 3.28. Wireframe Payslip List Staff

Gambar 3.29 adalah gambar *wireframe* pada halaman *Payslip Detail Staff*. Halaman ini menampilkan seluruh data *payslip* tersebut. Halaman ini didukung dengan fitur *Export PDF* pada *salary slip* yang dipilih.

UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

Payslip

Employee Name Payslip

Salary Slip

Start Date - End Date

Name : Employee Name
NIK : XXXXXXXXXXXXXXXX
Contact : XXXXXXXXXXXXXXXX

Job Title : Employee Name
Employment Status : XXXXXXXXXXXXXXXX
Attendance : XXXXXXXXXXXXXXXX

Income	Value (Rp)	Action

Deduction	Value (Rp)	Action

Take Home Pay Rp Value

Transferred to:
Bank Name
Bank Account Number
Bank Account Name

[Back](#) [Export PDF](#)

Gambar 3.29. Wireframe Payslip Detail Staff

3.5 Implementasi

Pengembangan modul *Daily Attendance & Payroll Human Resource Internal System* telah diimplementasikan di *mode production* dan telah digunakan oleh karyawan pada PT Visi Karya Nusantara. Berikut merupakan hasil akhir modul *Daily Attendance & Payroll HRIS* yang telah selesai dikembangkan.

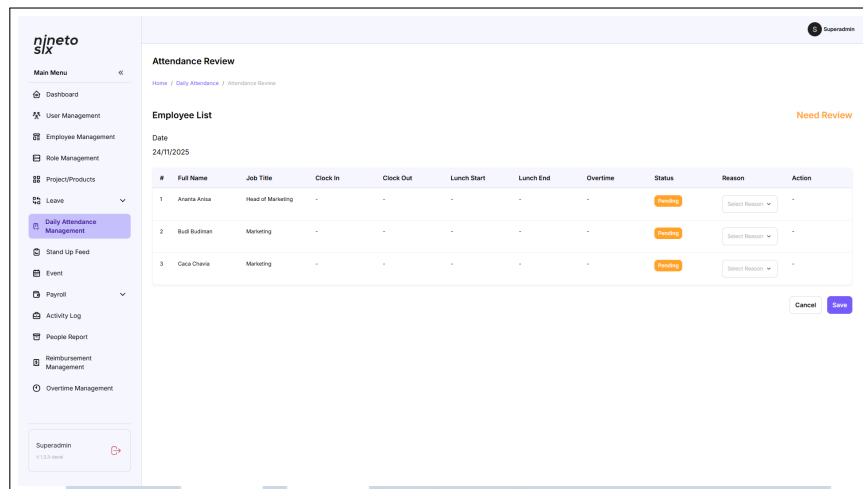
Pada halaman *Daily Attendance Management*, terdapat daftar seluruh *daily attendance management*. Daftar *daily attendance management* muncul setelah proses GET API ke *endpoint "/attendance-dates/"* dengan mengirimkan params berupa *page*, *row*, *pagination*, *search* jika ada, *sort* jika ada, dan *filter* jika ada. Setelah mendapatkan respons, data diambil dengan memasukkan *array* yang berisi daftar *daily attendance* yaitu *res.data.data.rows* ke dalam suatu variabel yaitu *attendanceData*. Data disimpan ke dalam *state* dengan *setAttendanceData(attendanceData)*. Kemudian, *user* dapat menambahkan *daily attendance* baru dengan menekan tombol *Attendance Review* di kanan atas. Selain itu, terdapat simbol berbentuk mata pada kolom *Action* yang berfungsi untuk melihat secara detail dan berfungsi untuk memperbarui *daily attendance* yang masih berstatus *Pending*. Jika status *daily attendance* masih *Pending*, maka terdapat simbol berbentuk tempat sampah di sebelah kanan simbol berbentuk mata yang berfungsi untuk menghapus *daily attendance* yang dipilih. Selain itu, didukung

dengan fitur *Search*, *Sort*, dan *Filter*. Kemudian, *user* dapat mencari Status *daily attendance management* dengan fitur *search*, melakukan *sort by* terhadap tabel *Daily Attendance Management* berdasarkan *Created* secara *ascending* maupun *descending*, dan dapat melakukan *filter by* berdasarkan Status, antara lain *Need Review* dan *Complete*.

#	Date	Total User	Status	Action
1	26/11/2025	3	Complete	
2	27/11/2025	3	Complete	
3	28/11/2025	3	Complete	
4	29/11/2025	3	Complete	
5	24/11/2025	3	Complete	
6	21/11/2025	3	Complete	
7	20/11/2025	3	Complete	
8	18/11/2025	3	Complete	
9	18/11/2025	3	Complete	
10	07/11/2025	3	Complete	

Gambar 3.30. Tampilan Daily Attendance Management

Pada halaman *Attendance Review*, *user* dapat melihat secara detail *daily attendance* masing-masing karyawan seperti *Full Name*, *Job Title*, *Clock In*, *Clock Out*, *Lunch Start*, *Lunch End*, *Overtime*, *Status*, *Reason*, dan *Action*. Jika terdapat alasan-alasan seperti keterlambatan kedatangan, pulang awal, keterlambatan makan siang maka pada kolom *Action* akan muncul simbol berbentuk mata yang dapat ditekan untuk melihat detail alasan-alasannya. Teks pada kolom *Clock In*, *Clock Out*, dan *Lunch Start* akan berubah warna dari hitam menjadi merah jika terdapat alasan-alasannya. Jika terdapat *overtime*, maka *user* dapat menekan *application id overtime* tersebut yang akan diarahkan ke halaman *Overtime Detail*. Selain itu, *user* dapat melakukan *review* dengan memilih salah satu alasan pada kolom *Reason* jika ada. Jika sudah selesai melakukan *review*, *user* dapat menekan tombol *Save*. Kemudian, pada kolom Status masing-masing karyawan akan berubah dari *Pending* menjadi *Approved*.



Gambar 3.31. Tampilan Attendance Review

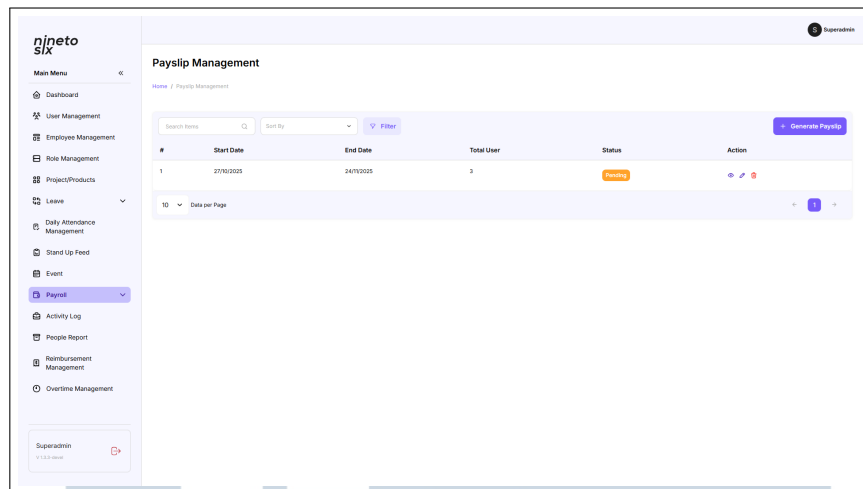
Pada halaman *Daily Attendance*, *user* hanya dapat melihat *daily attendancenya* secara pribadi. *User* dapat melihat *Date*, *Clock In*, *Clock Out*, *Lunch Start*, *Lunch End*, *Overtime*, *Status*, *Reason*, dan *Action*. Jika terdapat alasan-alasan seperti keterlambatan kedatangan, pulang awal, keterlambatan makan siang, maka pada kolom *Action* akan muncul simbol berbentuk mata yang dapat ditekan untuk melihat detail alasan-alasannya. Jika terdapat *overtime*, maka *user* dapat menekan *application id overtime* tersebut yang akan diarahkan ke halaman *Overtime Detail*. Selain itu, didukung dengan fitur *Search*, *Sort*, dan *Filter*. *User* dapat mencari Status *daily attendance management* dengan fitur *search*, melakukan *sort by* terhadap tabel *Daily Attendance Management* berdasarkan *Created* secara *ascending* maupun *descending*, dan dapat melakukan *filter by* berdasarkan status, antara lain *Need Review* dan *Complete*.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

#	Date	Clock In	Clock Out	Lunch Start	Lunch End	Overtime	Status	Reason	Action
1	26/7/2025	09:00	18:00	-	-	-	Approved	-	-
2	27/7/2025	09:00	18:00	-	-	-	Approved	-	-
3	28/7/2025	09:00	18:00	-	-	-	Approved	-	-
4	25/7/2025	09:00	18:00	-	-	-	Approved	-	-
5	24/7/2025	09:00	18:00	-	-	-	Approved	-	-
6	21/7/2025	09:00	18:00	-	-	-	Approved	-	-
7	20/7/2025	09:00	18:00	-	-	-	Approved	-	-
8	18/7/2025	09:00	18:00	-	-	-	Approved	-	-
9	16/7/2025	09:00	18:00	-	-	-	Approved	-	-
10	17/7/2025	09:00	18:00	-	-	-	Approved	-	-

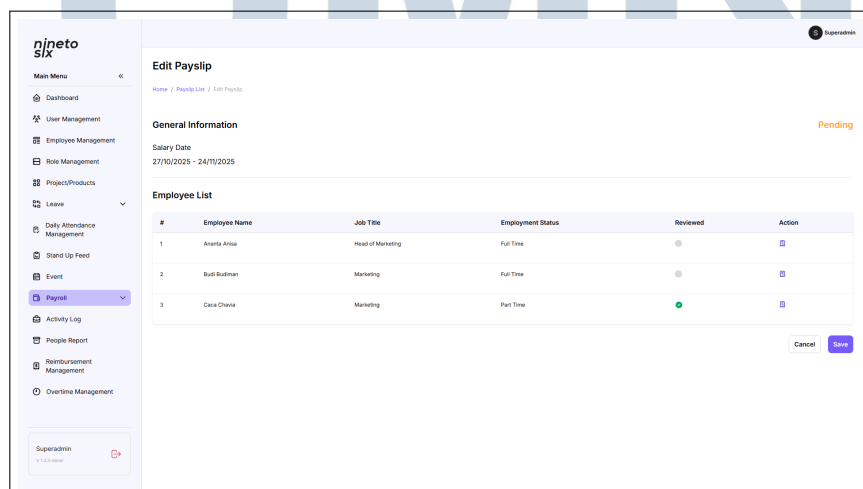
Gambar 3.32. Tampilan Daily Attendance

Pada halaman *Payslip Management*, terdapat daftar seluruh *payslip management*. Daftar *payslip management* muncul setelah proses GET API ke *endpoint* `"/payslip"` dengan mengirimkan params berupa *page*, *row*, *pagination*, *search* jika ada, *sort* jika ada, dan *filter* jika ada. Setelah mendapatkan respons, data diambil dengan memasukkan *array* yang berisi daftar *payslip management* yaitu `res.data.data.rows` ke dalam suatu variabel yaitu *payslips*. Data disimpan ke dalam *state* dengan `setPayslipData(payslips)`. Kemudian, *user* dapat menambahkan *payslip* baru dengan menekan tombol *Generate Payslip* di kanan atas. Selain itu, terdapat simbol berbentuk mata pada kolom *Action* yang berfungsi untuk melihat secara detail dan juga terdapat simbol berbentuk pencil yang berfungsi untuk memperbarui *payslip* yang masih berstatus *Pending*. Jika status *Payslip* masih *Pending*, maka terdapat simbol berbentuk tempat sampah di sebelah kanan simbol berbentuk pensil yang berfungsi untuk menghapus *payslip* yang dipilih. Selain itu, didukung dengan fitur *Search*, *Sort*, dan *Filter*. Kemudian, *user* dapat mencari Status *payslip management* dengan fitur *search*, melakukan *sort by* terhadap tabel *Payslip Management* berdasarkan *Updated* secara *ascending* maupun *descending*, dan dapat melakukan *filter by* berdasarkan Status, antara lain *Pending* dan *Complete*.



Gambar 3.33. Tampilan Payslip Management

Pada halaman *Payslip Overview*, user dapat melihat *payslip* sesuai dengan tanggal yang dipilih yaitu dibawah *Salary Date*. Terdapat *Employee List* yang menampilkan masing-masing informasi karyawan seperti *Employee Name*, *Job Title*, *Employment Status*, *Reviewed*, dan *Action*. User dapat melihat secara detail *payslip* masing-masing karyawannya dengan menekan simbol berbentuk kuitansi pada kolom *Action*. Ketika user menekan simbol tersebut, maka akan diarahkan ke halaman *Payslip Detail* karyawan yang dipilih. Jika *payslip* sudah ditinjau, maka pada kolom *Reviewed* akan muncul simbol berbentuk centang berwarna hijau. Jika seluruh *payslip* sudah ditinjau maka user dapat mengirimkan *payslip* kepada masing-masing karyawan dengan menekan tombol *Save* pada bagian kanan paling bawah.



Gambar 3.34. Tampilan Payslip Overview

Pada halaman *Payslip Detail*, user dapat melihat *payslip* karyawan yang dipilih secara detail sesuai dengan tanggal yang dipilih yaitu dikanan *Salary Slip*. Halaman ini menampilkan informasi karyawan seperti *Name*, *NIK*, *Contact*, *Job Title*, *Employment Status*, *Attendance*, *Income*, *Deduction*, *Take Home Pay*, *Bank Name*, *Bank Number*, dan *Bank Account Name*. User dapat menambahkan *Overtime* dan *Reimbursement* jika ada dengan menekan tombol *Add Item*, kemudian user dapat memilih *item* yang akan dicairkan dengan menekan pilihan *Item Type*. Setelah pilihan sudah dipilih, maka akan muncul data-data yang dipilih jika ada, kemudian, user dapat centang pada data yang akan dicairkan. Setelah selesai, user dapat menekan tombol *Add Item* dan data-data yang akan dicairkan masuk ke *Income payslip* karyawan tersebut. User juga dapat memperbarui *item* tersebut dengan menekan simbol berbentuk pensil pada kolom *Action* dan dapat menghapus *item* yang dipilih dengan menekan simbol berbentuk tempat sampah. Selain itu, jika terdapat tambahan *income* atau *deduction*, user dapat menambahkannya dengan menekan *Add Row* pada masing-masing bagian. *Income* atau *Deduction* yang ditambahkan dapat diperbarui dengan menekan simbol berbentuk pensil dan dapat dihapus dengan menekan simbol berbentuk tempat sampah. Seluruh total *Income* dan *Deduction* akan terlihat pada *Take Home Pay*. Jika *payslip* sudah ditinjau, maka user dapat menekan tombol *Save*.

Gambar 3.35. Tampilan Selected User Payslip Detail

Pada halaman *Payslip List*, user hanya dapat melihat *payslip* secara pribadi. User dapat melihat *Start Date*, *End Date*, *Employee Name*, *Job Title*, *Employment Status*, dan *Action*. User dapat melihat *payslip* secara detail dengan menekan simbol berbentuk mata pada kolom *Action* yang akan diarahkan ke halaman *Payslip*

Detail. Selain itu, didukung dengan fitur *Search*, *Sort*, dan *Filter*. Kemudian, *user* dapat mencari Status *payslip* dengan fitur *search*, melakukan *sort by* terhadap tabel *Payslip* berdasarkan *Updated* secara *ascending* maupun *descending*, dan dapat melakukan *filter by* berdasarkan Status, antara lain *Pending* dan *Complete*.

Gambar 3.36. Tampilan Payslip List

Pada halaman *Payslip Detail*, *user* dapat melihat *payslip* secara detail sesuai dengan tanggal yang dipilih yaitu dikanan *Salary Slip*. Halaman ini menampilkan informasi karyawan seperti *Name*, *NIK*, *Contact*, *Job Title*, *Employment Status*, *Attendance*, *Income*, *Deduction*, *Take Home Pay*, *Bank Name*, *Bank Number*, dan *Bank Account Name*. Selain itu, *user* dapat mengunduh slip gaji tersebut dengan menekan tombol *Export PDF*.

Gambar 3.37. Tampilan Selected Staff Payslip Detail

3.6 Kendala dan Solusi yang Ditemukan

Beberapa kendala yang dialami selama melakukan magang di PT Visi Karya Nusantara sebagai berikut.

1. Terbatasnya pengetahuan terhadap *boilerplate* perusahaan.
2. Komunikasi kurang efektif karena adanya perbedaan waktu respon di *Discord*.
3. Tidak adanya batasan yang jelas antara waktu kerja dan waktu istirahat.

Solusi yang ditemukan untuk menghadapi masalah tersebut sebagai berikut.

1. Memerlukan waktu untuk mempelajari *boilerplate* perusahaan, sebelum memulai pengembangan fitur-fitur pada HRIS.
2. Meningkatkan efektivitas komunikasi dengan melakukan panggilan langsung.
3. Menerapkan manajemen waktu dengan menyusun *to-do list* harian.

