

BAB 3

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Selama pelaksanaan kegiatan magang di PT Ganda Visi Jayatama, kedudukan yang ditempati adalah *Frontend Engineer Intern* di bawah bimbingan dan pengawasan Bapak Edo Setiawan. Tugas utama selama magang adalah mengembangkan *Human Resource Information System* (HRIS), yang mencakup perancangan dan implementasi antarmuka pengguna untuk meningkatkan pengalaman serta efisiensi sistem dalam pengelolaan data kepegawaian.

Dalam pengembangan HRIS, tim terdiri dari satu *Project Manager* (PM), satu *Designer*, dua *Frontend Engineer*, dan dua *Backend Engineer*. Selama proses pengembangan, komunikasi dan koordinasi dilakukan secara langsung di kantor serta melalui aplikasi Discord untuk memastikan kelancaran kerja tim. Tinjauan progres dan *code review* pengembangan HRIS dilakukan setiap minggu, sementara *sprint retrospective* diadakan dua kali seminggu. Selain itu, terdapat sesi *one-on-one* dengan *supervisor* untuk memberikan *feedback* secara personal terkait perkembangan proyek. *Feedback* yang diperoleh kemudian diterapkan pada *sprint* berikutnya untuk meningkatkan kualitas pengembangan.

3.2 Tugas yang Dilakukan

Selama pelaksanaan magang di PT Ganda Visi Jayatama, tugas utama yang dikerjakan adalah mengembangkan *Human Resource Information System* (HRIS), yang nantinya akan digunakan oleh seluruh karyawan perusahaan. Adapun tugas yang dilakukan meliputi:

1. Merancang *design* dan *revamp* pada modul *leave permit*, *reimbursement*, *clock in late*, *people report*, *event* dan *payroll*
2. Membangun *frontend* pada modul *leave permit*, *clock in late*, *people report* dan *event*.
3. Melakukan manual *testing* terhadap HRIS.

3.3 Uraian Pelaksanaan Magang

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

Minggu Ke -	Pekerjaan yang dilakukan
1	<i>Onboarding</i> dan mempelajari <i>boilerplate frontend</i> perusahaan.
2	Mempelajari <i>Design System</i> serta merancang <i>wireframe</i> untuk fitur <i>clock-in late</i> .
3	Merancang <i>wireframe leave permit</i> dan <i>reimbursement</i> .
4	Merancang <i>design high-fidelity</i> untuk fitur <i>clock-in late</i> , <i>leave permit</i> dan <i>reimbursement</i> .
5	Melakukan pengembangan UI halaman <i>leave permit</i> .
6	Melakukan integrasi API dan <i>manual testing</i> pada halaman <i>leave permit</i> .
7	Melakukan pengembangan UI fitur <i>clock-in late</i> .
8	Melakukan integrasi API dan <i>IP wifi</i> pada fitur <i>clock-in late</i> .
9	Merancang desain <i>high-fidelity</i> untuk fitur <i>People Report</i> serta mengembangkan UI pada halaman <i>People Report</i> .
10	Melakukan integrasi API dan <i>manual testing</i> pada halaman <i>People Report</i> .
11	Research mengenai fitur <i>Payroll</i> dan <i>Event Calendar</i> . Lalu, merancang <i>wireframe Payroll</i> dan <i>Event Calendar</i> .
12	Melanjutkan Perancangan <i>Wireframe</i> dan <i>high-fidelity</i> untuk fitur <i>Payroll</i> dan <i>Event Calendar</i> .
13	Mengerjakan UI halaman <i>Event big calendar</i> dan <i>Event Input</i> .
14	Melakukan integrasi API dan <i>manual testing</i> pada halaman <i>Event big calendar</i> .
15	Melakukan integrasi API dan <i>manual testing</i> pada halaman <i>Event Input</i> .
16	Melakukan <i>manual testing</i> pada seluruh fitur halaman, dilanjutkan dengan perbaikan bug (<i>bug fixing</i>) dan proses <i>deployment</i> .
17	Mengerjakan UI halaman dan integrasi API Event Type.
18	Melakukan <i>manual testing</i> pada halaman Event Type dan seluruh fitur halaman, dilanjutkan dengan perbaikan bug (<i>bug fixing</i>) dan proses <i>deployment</i> .

Tabel 3.1 di atas merupakan uraian pelaksanaan magang yang dilaksanakan pada PT Ganda Visi Jayatama.

Selama 18 minggu, terlibat dalam pengembangan *frontend* dengan memulai *onboarding* dan mempelajari *boilerplate* perusahaan. Mendalami *Design System* serta merancang *wireframe* untuk fitur *clock-in late*, *leave permit*, dan *reimbursement*, yang kemudian dikembangkan menjadi desain *high-fidelity*. Selanjutnya, memasuki pengembangan UI untuk halaman *leave permit* serta *leave management*, dan melakukan integrasi API serta *manual testing*. Dilanjutkan dengan mengembangkan UI fitur *clock-in late* dan mengintegrasikan API serta *IP Wifi*. Lalu, merancang desain *high-fidelity* untuk fitur *people report*, mengembangkan UI, serta melakukan integrasi API dan *manual testing* pada halaman tersebut. Dilanjutkan dengan melakukan *research* terkait fitur *Payroll* dan *Event Setting*. Berikutnya, merancang *wireframe* dan *high-fidelity* untuk fitur *Payroll* dan *Event Setting*. Setelah itu, mengerjakan UI untuk *Event Setting* serta melakukan integrasi API dan *manual testing* pada halaman tersebut.

3.4 Perancangan

Sub bab ini menjelaskan proses perancangan aplikasi *Human Resource Information System (HRIS)*, yang mencakup *User Requirement*, *Sitemap*, *Flowchart*, dan *Wireframe*.

3.4.1 User Requirement

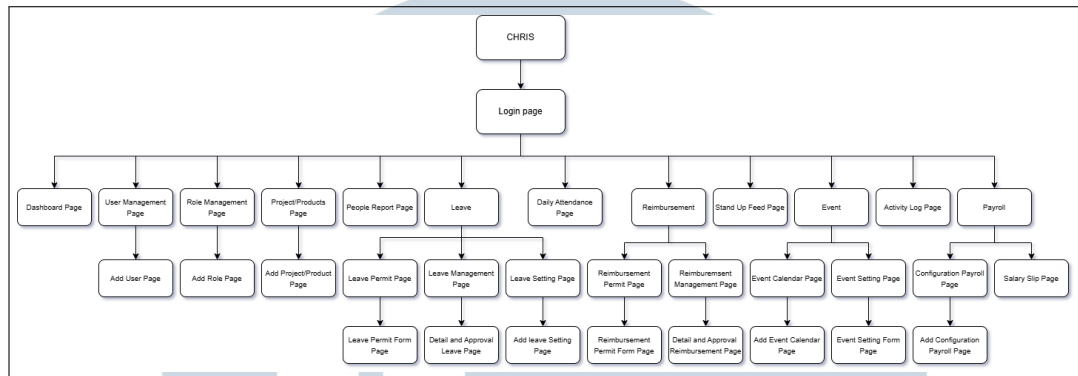
Sebelum memulai pengembangan, *Product Manager* memberikan arahan terkait tugas yang akan dilakukan dalam pengembangan *Human Resource Information System (HRIS)*. Aspek yang perlu dikembangkan di antaranya adalah sebagai berikut:

1. Mengembangkan halaman *leave permit* untuk pengajuan cuti, serta *leave management* untuk pencatatan karyawan yang melakukan izin. Selain itu, daftar cuti akan ditampilkan di *Dashboard* agar dapat melihat siapa saja yang mengambil izin dalam minggu tersebut dan siapa *Person in Charge* yang menggantikan mereka. Persetujuan atau penolakan cuti hanya dapat dilakukan oleh user yang mendapatkan *permission action*-nya.
2. Mengembangkan halaman *People Report* dengan menambahkan fitur *summary* kehadiran setiap karyawan. *Summary* tersebut mencakup total

absensi yang telah diterima, jumlah ketidakhadiran, dan total hari kerja yang seharusnya dipenuhi dalam satu bulan.

3. Mengembangkan fitur *Clock In* untuk menampilkan karyawan yang terlambat. Saat input stand-up, karyawan yang mengalami keterlambatan akan ditandai dengan *badge "Late"* serta kolom *Reason Late* untuk mencatat alasan keterlambatan. Fitur ini memungkinkan seluruh karyawan di kantor melihat siapa saja yang terlambat.
4. Merancang halaman *Reimbursement* dengan menambahkan dua fitur utama, yaitu *Reimbursement Permit* untuk pengajuan penggantian dana pribadi oleh karyawan, serta *Reimbursement Management* untuk pencatatan dan pengelolaan pengajuan, termasuk proses *accept* atau *reject* oleh pihak yang berwenang.
5. Merancang halaman *Event Setting* dengan membangun fitur *Big Calendar* yang digunakan untuk melihat jadwal kegiatan dan hari libur perusahaan. Pada setiap kegiatan yang dibuat, hanya karyawan yang tercantum dalam daftar undangan yang dapat melihatnya. Namun, apabila jadwal libur ditambahkan, maka secara otomatis seluruh karyawan akan terundang. Selain itu, disediakan fitur *Event Input* untuk menambahkan kegiatan beserta tanggal pelaksanaannya.
6. Merancang halaman *Payroll* dengan membangun beberapa fitur utama, yaitu *Configuration Payroll* untuk membuat konfigurasi awal berupa tipe tunjangan berdasarkan status karyawan yang hanya dapat diakses oleh admin, *Salary Slip* yang memungkinkan karyawan melihat slip gaji setiap bulan sementara admin dapat mengakses seluruh slip gaji karyawan, serta mengembangkan fitur *A1 Generate* dan *Payroll Information* pada halaman *User Management* yang juga hanya dapat diakses oleh admin. Dalam fitur *A1 Generate*, admin dapat mengunggah file A1 untuk setiap karyawan, sehingga karyawan hanya perlu mengunduh file tersebut melalui halaman *Salary Slip*. Sedangkan, *Payroll Information* digunakan untuk melakukan pengaturan gaji berdasarkan konfigurasi yang dipilih serta menambahkan *one-time bonus* ke dalam daftar *payroll* karyawan.

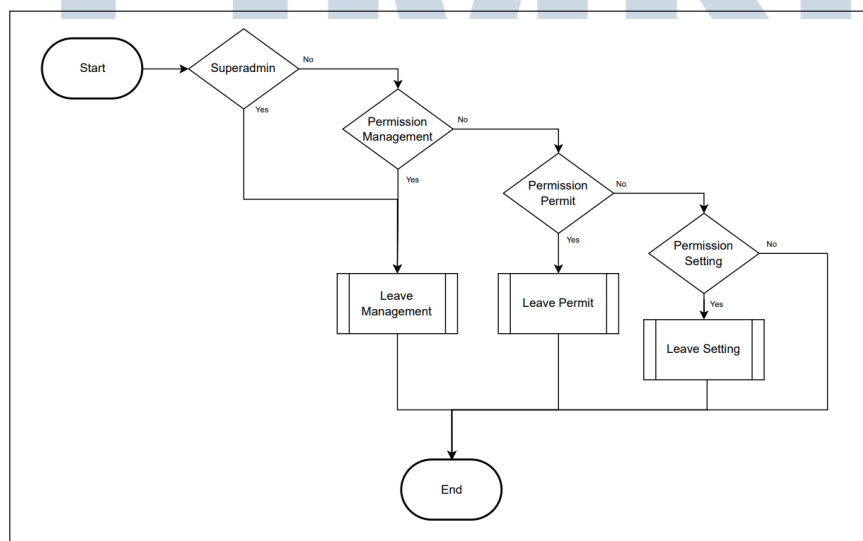
3.4.2 Sitemap



Gambar 3.1. Sitemap Human Resource Information System

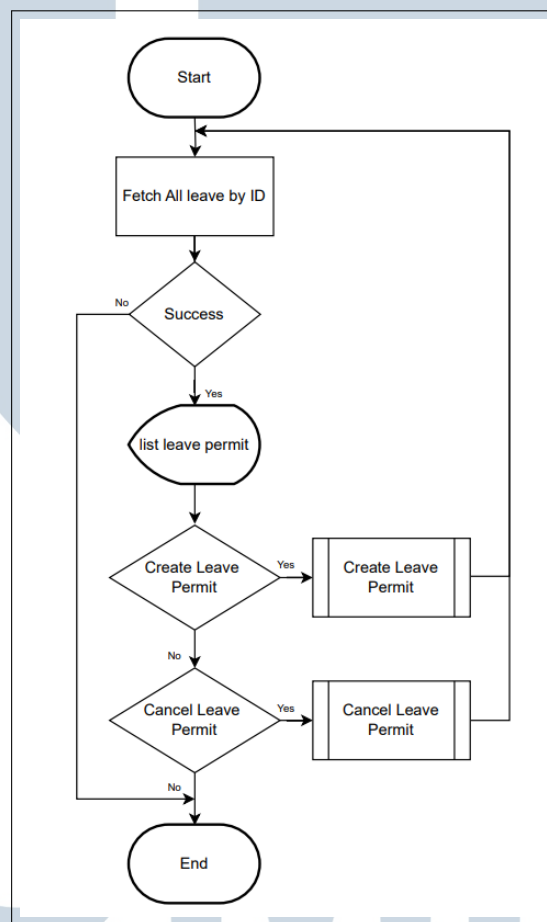
Gambar 3.1 menunjukkan *sitemap* dari sistem *Human Resource Information System* (HRIS) yang telah dikembangkan. Berdasarkan gambar tersebut, halaman pertama yang akan ditampilkan kepada pengguna adalah halaman *login*. Setelah berhasil *login*, pengguna akan langsung diarahkan ke halaman *dashboard*. Dari halaman *dashboard* ini, pengguna dapat mengakses berbagai halaman lainnya seperti *Leave Permit*, *People Report*, *Event Setting*, dan halaman-halaman lain yang tersedia.

3.4.3 Flowchart



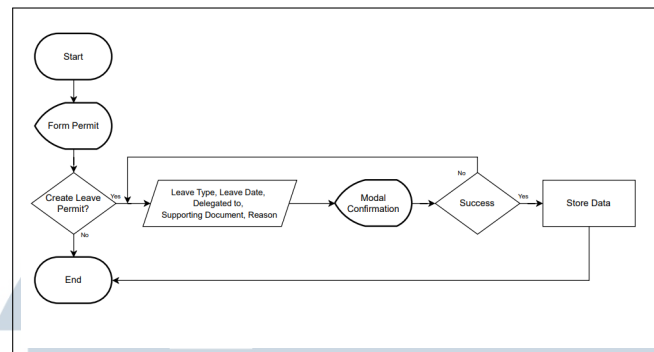
Gambar 3.2. Flowchart Leave Permission Page

Gambar 3.2 merupakan *flowchart* pembagian halaman berdasarkan *permission* yang diberikan pada halaman *Leave*. Sistem akan memeriksa apakah pengguna adalah *superadmin* atau bukan. Jika bukan *superadmin*, sistem akan mengecek apakah pengguna memiliki izin untuk mengakses *Leave Management*. Berdasarkan *permission* tersebut, setiap pengguna dapat mengakses halaman-halaman yang tersedia.



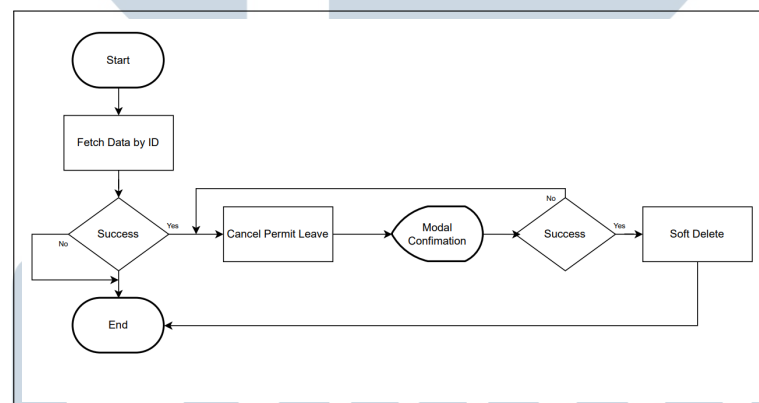
Gambar 3.3. Flowchart Leave Permit

Gambar 3.3 memperlihatkan *flowchart* pada halaman *Leave Permit*. Proses diawali dengan pemanggilan *GET* API untuk menampilkan seluruh daftar *leave* berdasarkan pengguna yang sedang *login*. Selanjutnya, pengguna dapat memilih untuk mengajukan *leave permit* baru atau membatalkan *leave* yang telah dibuat.



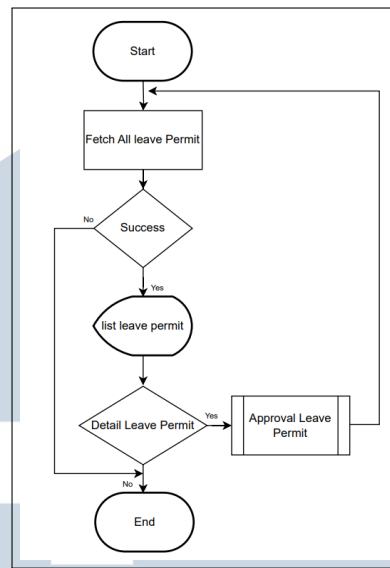
Gambar 3.4. *Flowchart Create Leave Permit*

Gambar 3.4 memperlihatkan *flowchart* pada halaman *Leave Input Form*. Proses dimulai dengan menampilkan *form field* untuk pengisian data perizinan cuti. Pengguna dapat mengunggah dan menghapus dokumen pendukung. Selanjutnya, akan ditampilkan modal konfirmasi pengajuan. Setelah konfirmasi, data perizinan akan disimpan ke dalam *database*.



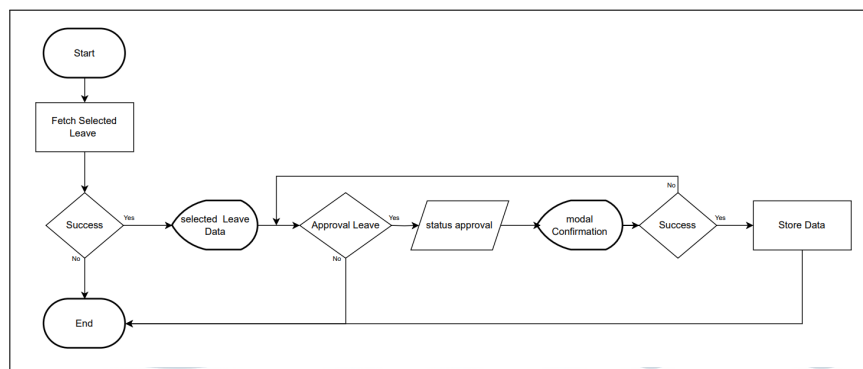
Gambar 3.5. *Flowchart Cancel Leave Permit*

Gambar 3.5 memperlihatkan *flowchart* proses pembatalan *leave permit*. Proses dimulai dengan pengambilan data perizinan yang telah diajukan sebelumnya. Jika data berhasil diambil, seluruh daftar *leave permit* akan ditampilkan. Selanjutnya, pengguna dapat memilih untuk membatalkan pengajuan cuti atau tidak. Setelah itu, akan ditampilkan modal konfirmasi. Apabila dikonfirmasi, data perizinan tersebut akan dihapus.



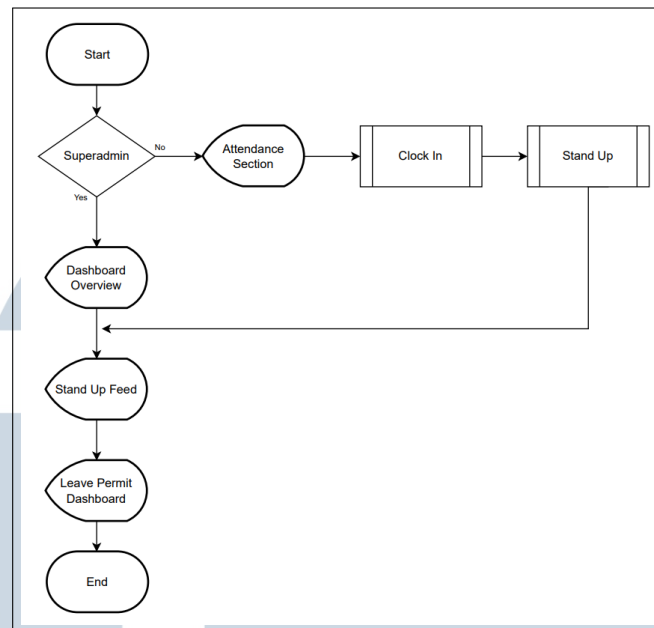
Gambar 3.6. *Flowchart Leave Management*

Gambar 3.6 adalah flowchart pada halaman *Leave Management*. Proses dimulai dengan pemanggilan GET API untuk menampilkan seluruh data *leave* pengguna. Pengguna atau *superadmin* kemudian dapat memilih data untuk melakukan proses *approval*.



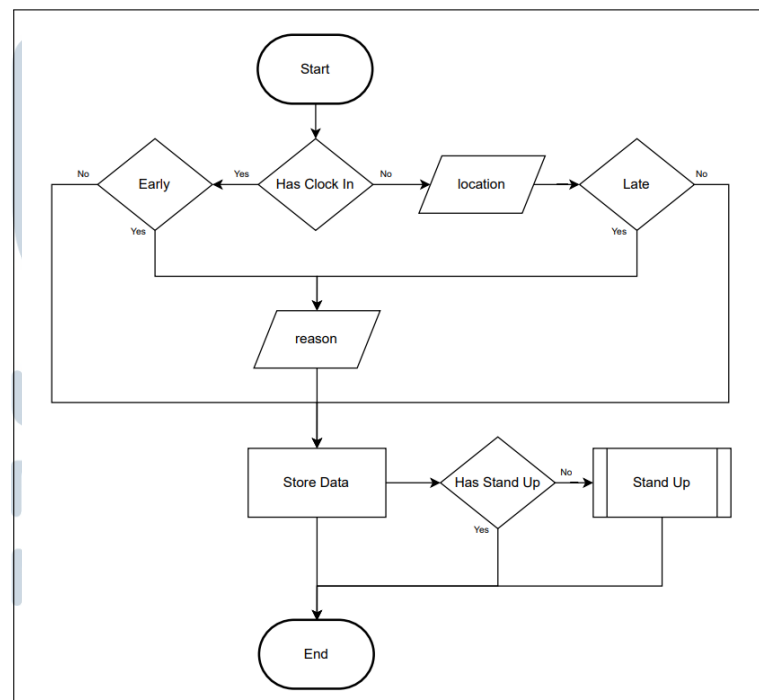
Gambar 3.7. *Flowchart Leave Approval*

Gambar 3.7 memperlihatkan *flowchart* pada halaman *Leave Approval*. Proses dimulai dengan pemanggilan GET API untuk mengambil data *leave* yang dipilih untuk di *review*. Selanjutnya, hasil *review* berupa keputusan *accept* atau *reject*. Setelah itu, akan ditampilkan modal konfirmasi. Setelah konfirmasi dilakukan, data perizinan yang telah di *review* akan disimpan ke dalam *database* dan dikirimkan kepada pengguna yang bersangkutan.



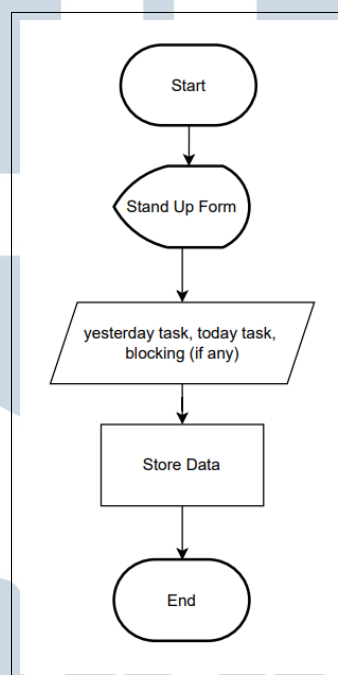
Gambar 3.8. Flowchart Dashboard Permission

Gambar 3.8 memperlihatkan *flowchart* pada *dashboard permission*. Jika pengguna merupakan *superadmin*, maka akan ditampilkan *dashboard overview* yang berisi data-data internal. Jika bukan, maka yang ditampilkan adalah *attendance section* untuk melakukan absensi.



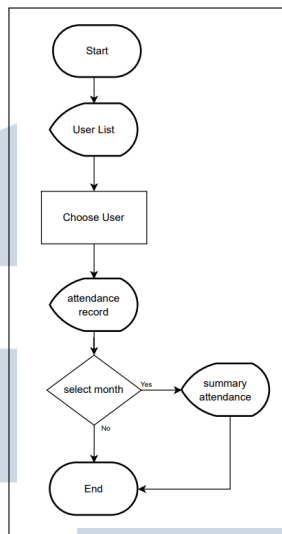
Gambar 3.9. Flowchart ClockIn

Gambar 3.9 memperlihatkan *flowchart* proses pengisian *clock-in* dan *clock-out*. Proses diawali dengan pengecekan apakah pengguna sudah melakukan *clock-in*. Jika belum, sistem akan meminta input berupa lokasi dan memeriksa apakah pengguna terlambat. Apabila terlambat, pengguna harus mengisi alasan keterlambatan sebelum data disimpan ke dalam *database*. Sebaliknya, jika pengguna sudah melakukan *clock-in*, sistem akan memeriksa apakah pengguna mencoba *clock-out* terlalu awal. Jika iya, pengguna akan diminta mengisi alasan, kemudian data akan disimpan ke dalam *database*.



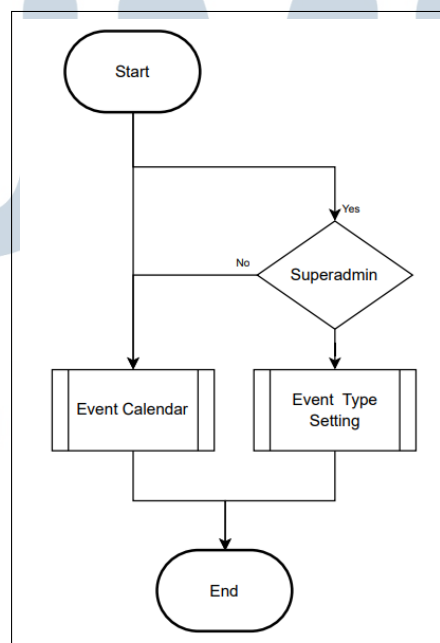
Gambar 3.10. *Flowchart Stand Up*

Gambar 3.10 memperlihatkan *flowchart* proses pengisian *daily standup*. Proses dimulai dengan munculnya *form input* yang memuat kolom *yesterday task*, *today task*, dan *blocking*. Setelah pengguna mengisi semua data, informasi tersebut akan disimpan ke dalam *database*.



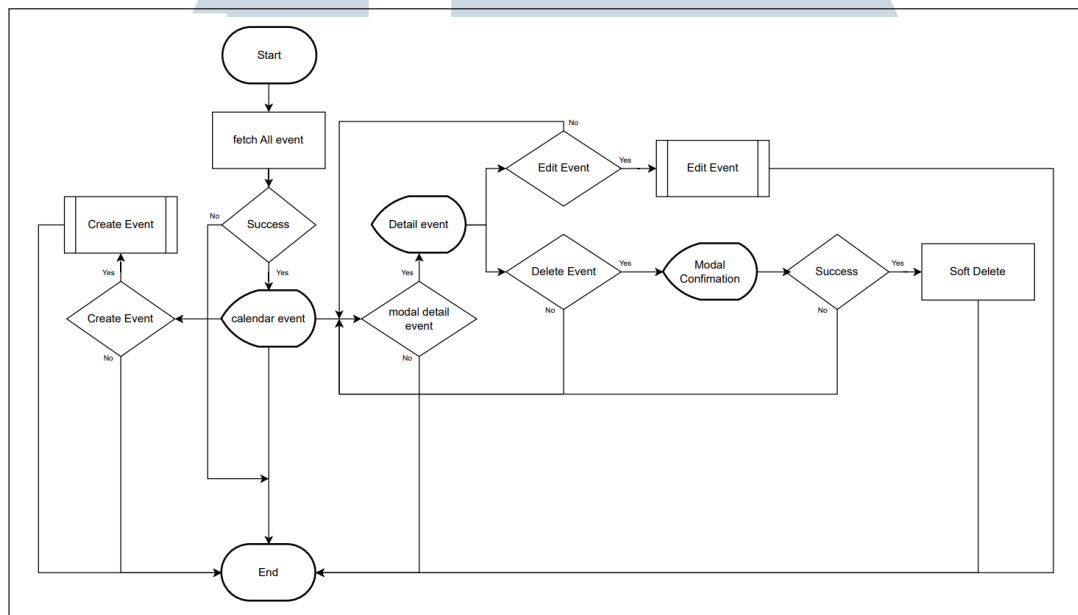
Gambar 3.11. *Flowchart Summary Attendance*

Gambar 3.11 memperlihatkan *flowchart* pada halaman *People Report* bagian *Summary Attendance*. Saat membuka halaman *People Report*, pengguna harus terlebih dahulu memilih pengguna yang ingin dilihat ringkasan absensinya. Setelah itu, akan ditampilkan total kehadiran, total ketidakhadiran, serta jumlah hari kerja dalam satu bulan. Selain itu, tersedia opsi untuk memilih periode waktu dalam format bulan dan tahun guna melihat rekam jejak absensi pada bulan-bulan sebelumnya.



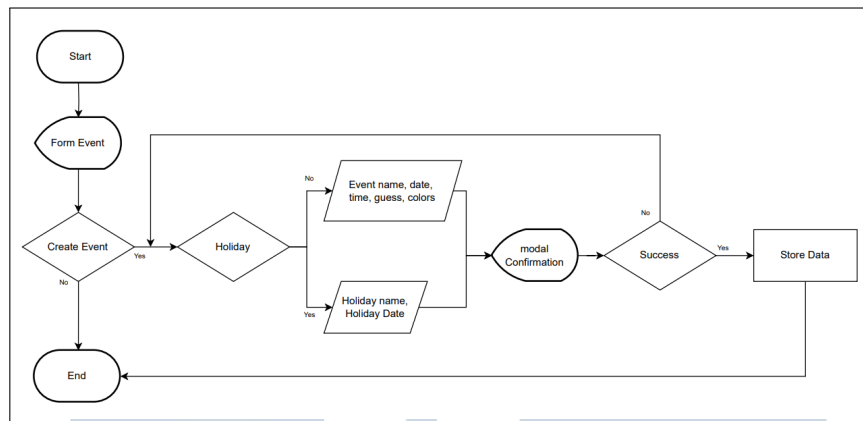
Gambar 3.12. *Flowchart Event Permission*

Gambar 3.12 memperlihatkan *flowchart* pembagian halaman berdasarkan *permission* yang diberikan pada halaman *Event*. Sistem akan terlebih dahulu memeriksa apakah pengguna merupakan *superadmin* atau bukan. Hanya pengguna dengan peran *superadmin* yang diberikan akses untuk melakukan pengaturan pada *Event Type Setting*, sedangkan pengguna lainnya hanya dapat melihat atau mengakses *event* yang tersedia sesuai dengan izin yang dimiliki.



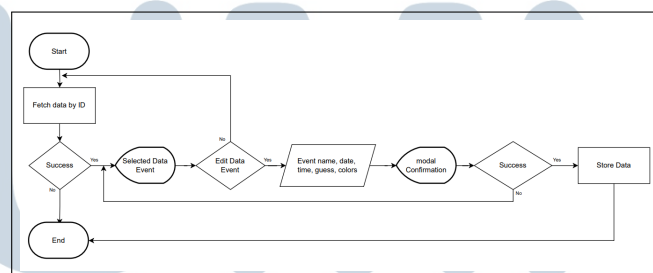
Gambar 3.13. *Flowchart Event Calendar*

Gambar 3.13 memperlihatkan *flowchart* pada halaman *Event Calendar*. Proses dimulai dengan pemanggilan GET API untuk mengambil data *event* dan hari libur yang telah tercatat. Pengguna dapat membuka detail *event* dengan menampilkan modal yang berisi informasi *event*, serta opsi untuk mengedit atau menghapus *event* tersebut. Jika pengguna memilih untuk menghapus *event*, sistem akan menampilkan modal konfirmasi sebelum penghapusan dilakukan, dan hanya pengguna yang membuat *event* tersebut yang memiliki hak untuk menghapusnya. Pengguna juga dapat membuat *event* baru atau menjadwalkan hari libur, apabila memiliki *permission* yang sesuai.



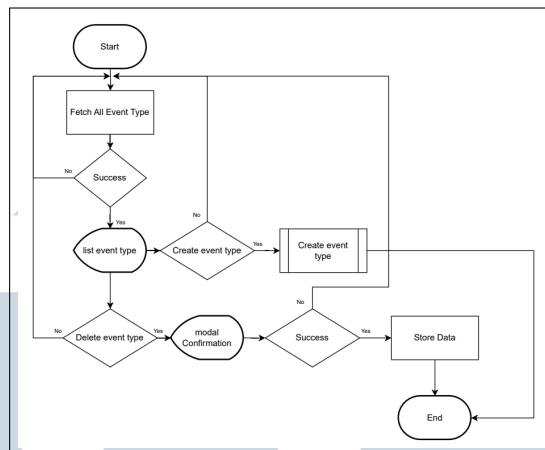
Gambar 3.14. *Flowchart Create Event Form*

Gambar 3.14 memperlihatkan *flowchart* pada halaman *Create Event*. Proses dimulai dengan menampilkan *form field* untuk mengisi data *event*. Terdapat dua pilihan utama untuk *event type*, yaitu *meeting* dan *holiday*. Apabila pengguna memilih *meeting*, maka data yang perlu diisi meliputi nama *meeting*, tanggal, waktu pelaksanaan, daftar karyawan yang terlibat, serta warna yang akan ditampilkan pada kalender. Sementara itu, jika memilih *holiday*, pengguna hanya perlu mengisi tanggal dan judul hari libur. Setelah pengisian data selesai, akan ditampilkan modal konfirmasi. Apabila disetujui, data *event* tersebut akan dikirimkan dan disimpan ke dalam *database*.



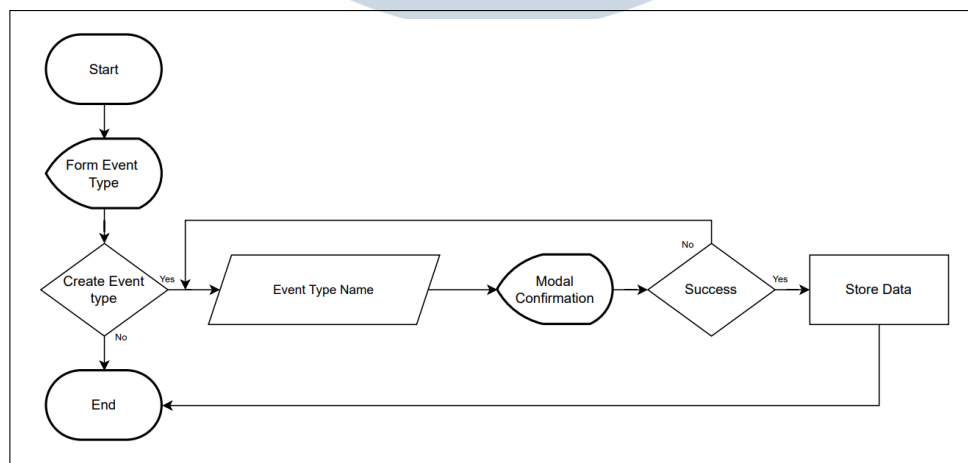
Gambar 3.15. *Flowchart Edit Event*

Gambar 3.15 memperlihatkan *flowchart* pada halaman *Edit Event*. Proses dimulai dengan pemanggilan GET API untuk mengambil data *event* yang dipilih untuk diubah. Pengguna dapat melakukan perubahan pada nama *event*, waktu pelaksanaan, daftar karyawan yang terlibat, serta warna *event*. Setelah seluruh perubahan selesai dilakukan, akan ditampilkan modal konfirmasi. Apabila konfirmasi disetujui, data *event* yang telah diperbarui akan dikirimkan dan disimpan ke dalam *database*.



Gambar 3.16. Flowchart Event Type

Gambar 3.16 memperlihatkan *flowchart* pada halaman *Event Type*. Proses dimulai dengan pemanggilan GET API untuk menampilkan data *event type* dalam bentuk tabel. Pengguna dapat membuat *event type* baru ataupun menghapus *event type* yang sudah ada. Jika pengguna memilih untuk menghapus, sistem akan menampilkan modal konfirmasi sebelum proses penghapusan dilakukan.

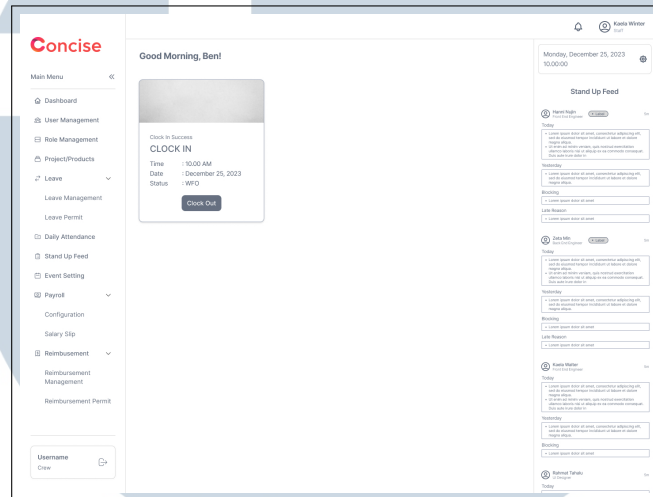


Gambar 3.17. Flowchart Create Event Type

Gambar 3.17 memperlihatkan *flowchart* pada halaman *Event Type Form*. Proses dimulai dengan menampilkan *form field* untuk membuat *event type* baru, di mana pengguna hanya perlu mengisi jenis *event type* yang akan dibuat. Setelah *form* diisi dan pengguna menekan tombol *submit*, sistem akan menampilkan modal konfirmasi. Apabila disetujui, data *event type* tersebut akan dikirimkan dan disimpan ke dalam *database*.

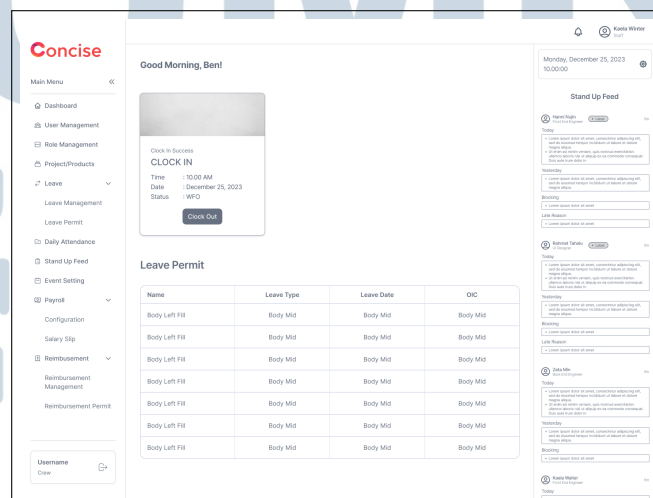
3.4.4 Wireframe

Wireframe adalah gambaran awal tampilan aplikasi yang menunjukkan susunan elemen pada halaman. Bagian ini akan menampilkan *wireframe* yang telah dikembangkan untuk aplikasi *Human Resource Information System* sebagai panduan dalam proses desain dan pengembangan.



Gambar 3.18. Wireframe ClockIn Badge Late

Gambar 3.18 adalah *wireframe* pada halaman *Dashboard*. Pada halaman ini, dikembangkan fitur *badge* untuk keterlambatan (*late badge*) dan alasan keterlambatan (*late reason*) yang ditampilkan pada *stand-up feed*. Fitur ini berfungsi untuk memudahkan *monitoring* keterlambatan setiap karyawan yang melakukan *clock-in* melebihi waktu yang telah ditentukan.



Gambar 3.19. Wireframe Leave Permit Dashboard

Gambar 3.19 adalah *wireframe* pada halaman *Dashboard*. Pada halaman ini ditambahkan fitur tabel *Leave Permit* yang berfungsi untuk memudahkan karyawan dalam melihat daftar rekan kerja yang mengajukan cuti pada minggu ini.

#	Leave Type	From	To	OIC	Status	Detail
1	Body Left Fill	Body Mid	Body Mid	Body Mid	Body Mid	⊗
2	Body Left Fill	Body Mid	Body Mid	Body Mid	Body Mid	⊗
3	Body Left Fill	Body Mid	Body Mid	Body Mid	Body Mid	⊗
4	Body Left Fill	Body Mid	Body Mid	Body Mid	Body Mid	⊗
5	Body Left Fill	Body Mid	Body Mid	Body Mid	Body Mid	⊗
6	Body Left Fill	Body Mid	Body Mid	Body Mid	Body Mid	⊗
7	Body Left Fill	Body Mid	Body Mid	Body Mid	Body Mid	⊗
8	Body Left Fill	Body Mid	Body Mid	Body Mid	Body Mid	⊗
9	Body Left Fill	Body Mid	Body Mid	Body Mid	Body Mid	⊗
10	Body Left Fill	Body Mid	Body Mid	Body Mid	Body Mid	⊗

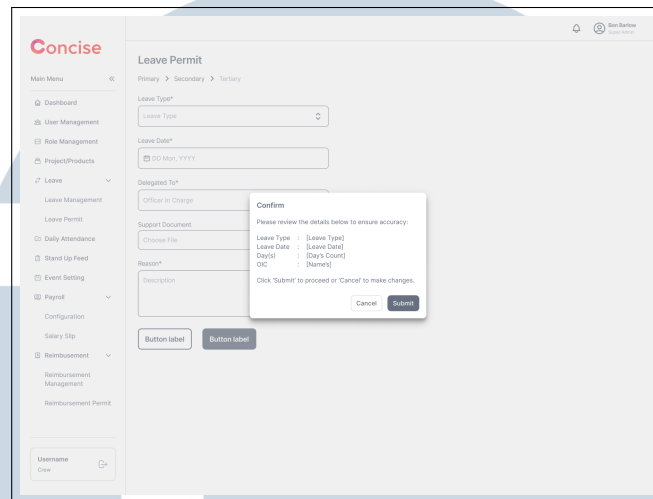
Gambar 3.20. *Wireframe Leave Permit List*

Gambar 3.20 menunjukkan *wireframe* halaman *Leave Permit*. Pada halaman ini terdapat tabel yang memuat kolom *Leave Type*, *From Date*, *To Date*, *Officer In Charge*, *Status*, dan *Detail*. Selain itu, terdapat tombol untuk membuat izin cuti. Halaman ini ditampilkan pada tampilan karyawan.

Gambar 3.21. *Wireframe Leave Permit Form*

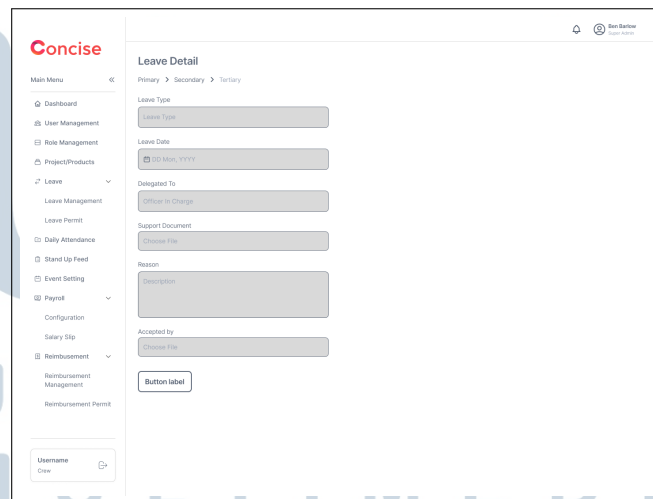
Gambar 3.21 menunjukkan *wireframe* halaman *Leave Permit Form*. Pada halaman ini terdapat beberapa *form fields*, yaitu *Leave Type*, *Date Range*, *Delegated*

To, dan *Reason*. Selain itu, terdapat tombol *Back* untuk kembali ke halaman sebelumnya, serta tombol *Submit* untuk mengirim data ke *database*.



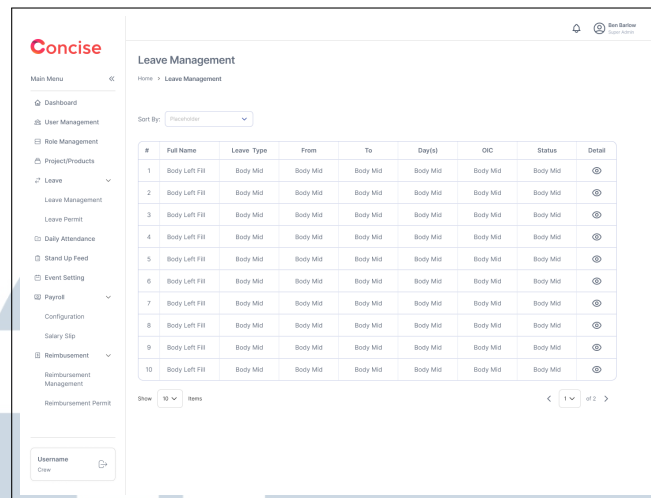
Gambar 3.22. Wireframe Leave Permit Modal Confirmation

Gambar 3.22 menunjukkan wireframe halaman *Leave Permit Form* pada bagian modal konfirmasi. Modal ini berfungsi untuk memastikan data yang akan dikirimkan, yang mencakup *Leave Type*, *Leave Date*, total hari izin, serta *Officer In Charge* yang ditunjuk.



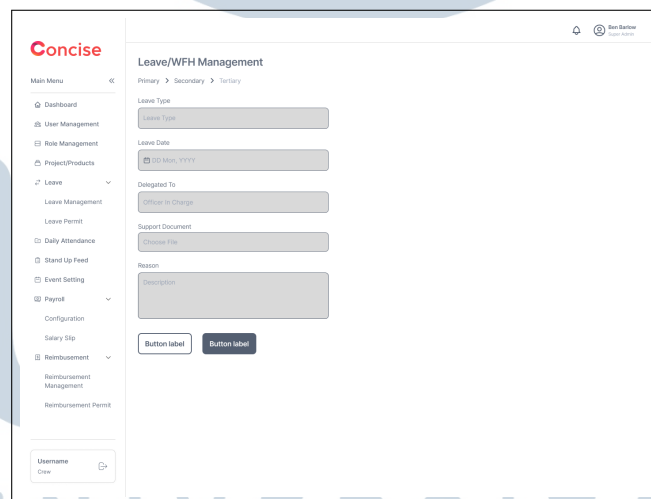
Gambar 3.23. Wireframe Leave Permit Detail

Gambar 3.23 menunjukkan wireframe halaman *Leave Permit Detail*. Halaman ini menampilkan detail data perizinan yang telah diinput dan ditinjau oleh *supervisor* untuk kemudian diterima (*accept*) atau ditolak (*reject*).



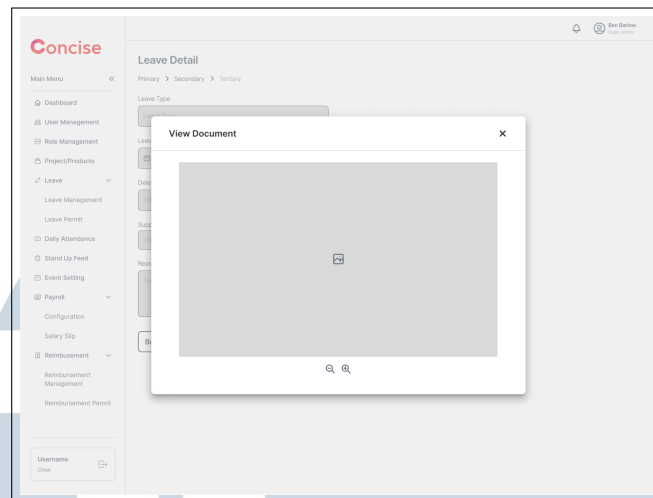
Gambar 3.24. Wireframe Leave Management List

Gambar 3.24 menunjukkan *wireframe* halaman *Leave Management*. Pada halaman ini terdapat tabel yang memuat kolom *Name*, *Leave Type*, *From Date*, *To Date*, *Day(s)*, *Officer In Charge*, *Status*, dan *Detail*. Tombol *Detail* digunakan untuk meninjau data perizinan secara lebih lengkap.



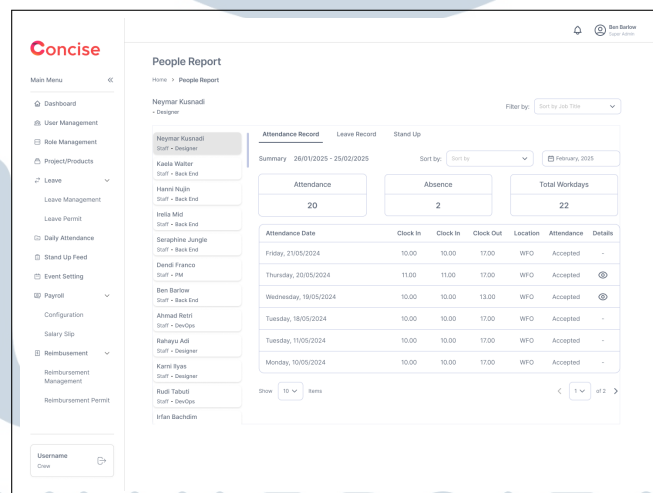
Gambar 3.25. Wireframe Leave Management Approval

Gambar 3.25 menunjukkan *wireframe* halaman *Leave Management Approval*. Pada halaman ini ditampilkan data karyawan yang mengajukan izin, yang kemudian akan ditinjau untuk disetujui (*accept*) atau ditolak (*reject*).



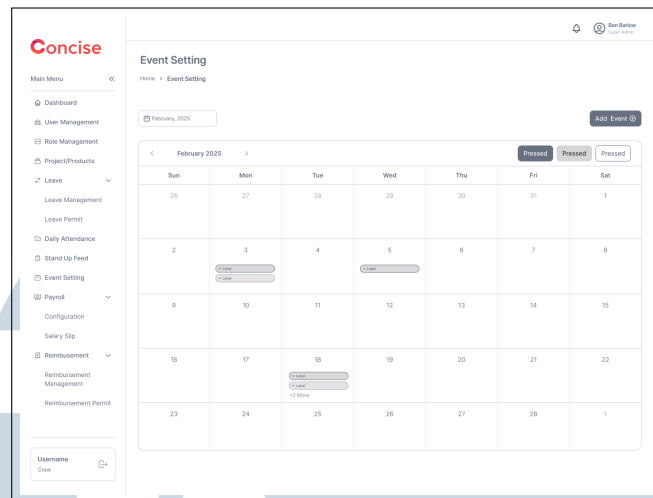
Gambar 3.26. Wireframe Leave Management Approval Modal Image

Gambar 3.26 menunjukkan *wireframe* tampilan berkas (*view file*) pada halaman *Leave Approval*. Pada halaman ini terdapat modal yang muncul ketika pengguna melihat berkas yang dipilih. Pengguna dapat memperbesar maupun memperkecil tampilan berkas tersebut.



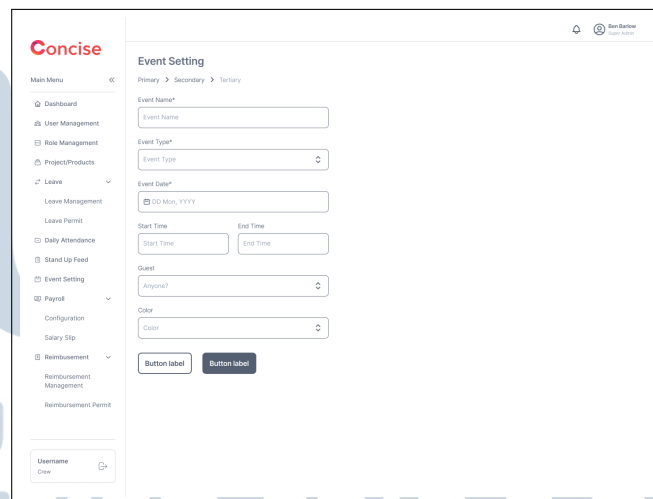
Gambar 3.27. Wireframe People Report Summary Attendance

Gambar 3.27 menunjukkan *wireframe* halaman *People Report*. Pada halaman ini ditampilkan daftar nama karyawan beserta filter berdasarkan *Job Title*. Selain itu, terdapat ringkasan (*summary*) kehadiran bulanan yang mencakup jumlah kehadiran, ketidakhadiran, dan total hari kerja dalam satu bulan. Di bagian bawah, terdapat tabel yang menampilkan daftar absensi secara rinci.



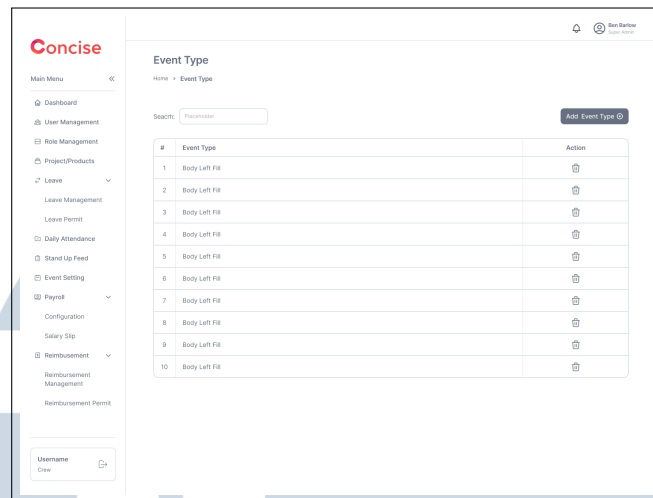
Gambar 3.28. Wireframe Event Setting Calendar

Gambar 3.28 menunjukkan *wireframe* halaman *Event Setting Calendar*. Pada halaman ini terdapat tampilan kalender besar (*big calendar*) yang dapat dilihat dalam format bulanan, mingguan, maupun harian. Pengguna dapat menambahkan berbagai jenis acara (*event*), seperti jadwal rapat maupun hari libur. Jadwal selain hari libur akan terlihat oleh karyawan yang namanya telah diinput dalam acara tersebut.



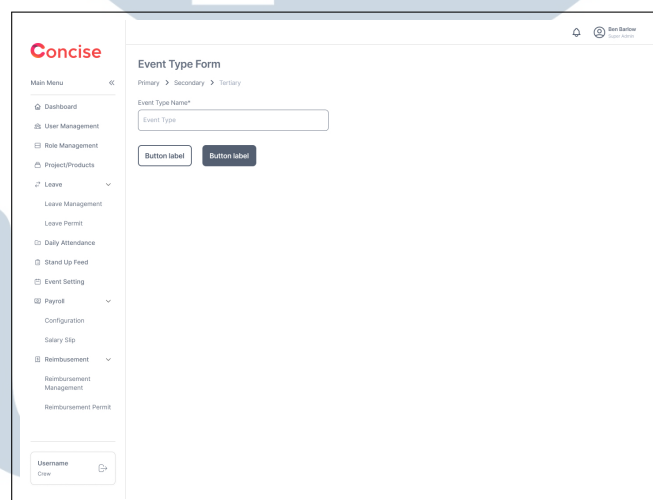
Gambar 3.29. Wireframe Event Setting Input

Gambar 3.29 menunjukkan *wireframe* halaman *Event Setting Input*. Pada halaman ini terdapat beberapa *form fields*, yaitu *Event Name*, *Event Type*, *Event Date*, *Event Time*, *Guest*, dan *Color*. Selain itu, tersedia tombol *Back* untuk kembali ke halaman sebelumnya, serta tombol *Submit* untuk mengirim data ke *database*.



Gambar 3.30. Wireframe Event Type List

Gambar 3.30 menunjukkan wireframe halaman *Event Type List*. Pada halaman ini terdapat tabel yang memuat kolom *Event Type* dan *Action*. Selain itu, tersedia tombol untuk menambahkan *event type* baru.

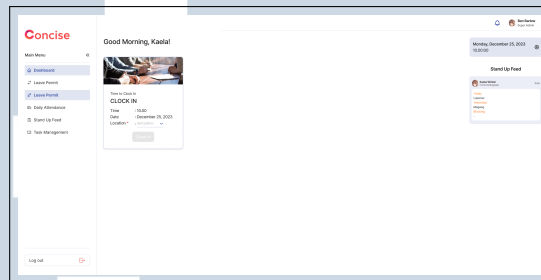


Gambar 3.31. Wireframe Event Type Form

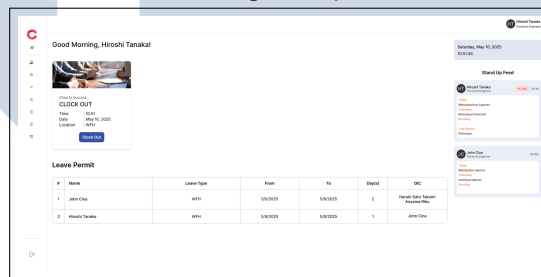
Gambar 3.31 menunjukkan wireframe halaman *Event Type Form*. Pada halaman ini terdapat form field, yaitu *Event Type Name*. Selain itu, tersedia tombol *Back* untuk kembali ke halaman sebelumnya, serta tombol Submit untuk mengirim data ke *database*.

3.5 Implementasi

Human Resource Internal System telah berhasil dikembangkan dan diimplementasikan pada mode *production*, serta telah digunakan oleh seluruh karyawan PT Ganda Visi Jayatama. Berikut ini merupakan tampilan akhir dari *Human Resource Internal System* yang telah selesai dikembangkan:



(a) Tampilan *Before*



(b) Tampilan *After*

Gambar 3.32. Tampilan halaman *Dashboard Before* dan *After*

Gambar 3.32 (a) menunjukkan tampilan halaman *Dashboard* sebelum dilakukan pengembangan. Pada tampilan sebelumnya, halaman hanya menampilkan *card* absensi dan masih terdapat banyak ruang kosong (*white space*) yang belum dimanfaatkan untuk menampilkan informasi tambahan. Selain itu, pada *stand-up card* belum tersedia badge keterlambatan (*late*), sehingga karyawan lain tidak dapat mengetahui siapa saja yang mengalami keterlambatan beserta alasannya.

Gambar 3.32 (b) menunjukkan tampilan halaman *Dashboard* setelah dilakukan pengembangan. Pada tampilan baru ini, ditambahkan tabel *Leave Permit* untuk menampilkan daftar karyawan yang mengajukan cuti pada minggu ini beserta informasi mengenai petugas pengganti (*Officer In Charge*). Selain itu, pada *stand-up card* juga ditambahkan *badge* keterlambatan (*late*) beserta alasannya, sehingga memudahkan pemantauan keterlambatan karyawan.

#	Leave Type	From	From	To	Days	Status	Details
1	Sick	10/01/2024	10/01/2024	10/01/2024	1	Pending	
2	Sick	10/01/2024	10/01/2024	10/01/2024	1	Accepted	
3	Sick	10/01/2024	10/01/2024	10/01/2024	1	Pending	
4	Annual	10/01/2024	10/01/2024	10/01/2024	1	Pending	

(a) Tampilan *Before*

#	Leave Type	From	To	Days	DCC	Status	Details
1	Annual	10/01/2024	10/01/2024	1	John Doe	Accepted	
2	WHF Permit	10/01/2024	10/01/2024	1	John Doe	Pending	

(b) Tampilan *After*

Gambar 3.33. Tampilan halaman *Leave Permit Before* dan *After*

Gambar 3.33 (a) menunjukkan tampilan halaman *Leave Permit* sebelum dilakukan *revamp*. Pada tampilan sebelumnya, terdapat inkonsistensi antara daftar data *leave* dengan tombol pembuatan permintaan *leave* dibandingkan dengan halaman-halaman lainnya. Selain itu, penggunaan dua tab secara bersamaan untuk mengajukan permintaan dan menampilkan daftar *leave* membuat antarmuka menjadi kurang intuitif, sehingga menyulitkan pengguna baru dalam mengoperasikan sistem. Oleh karena itu, dilakukan perbaikan untuk meningkatkan konsistensi tampilan dan mempermudah pengalaman pengguna.

Gambar 3.33 (b) menunjukkan tampilan halaman *Leave Permit* setelah dilakukan *revamp* pada bagian daftar (*list*). Pada tampilan baru ini, halaman difokuskan hanya untuk menampilkan daftar permintaan cuti yang telah diajukan. Apabila pengguna ingin membuat permintaan cuti baru, tersedia tombol yang akan mengarahkan ke halaman *Leave Permit Form* untuk melakukan pengajuan.

Gambar 3.34. Tampilan *After Leave Permit Form*

Gambar 3.34 menunjukkan tampilan halaman *Leave Permit* setelah dilakukan *revamp* pada bagian formulir (*form*). Pada tampilan baru ini, pengajuan cuti memiliki halaman tersendiri. Penambahan kolom *Officer In Charge* memudahkan karyawan untuk mengetahui siapa yang akan bertanggung jawab menggantikan tugas selama masa cuti berlangsung.

#	Leave Type	From	To	Days	Status	Details
1	Sick	16/05/2024	16/05/2024	1	Pending	
2	Sick	16/05/2024	16/05/2024	1	Approved	
3	Sick	16/05/2024	16/05/2024	1	Pending	
4	Annual	16/05/2024	16/05/2024	1	Pending	

(a) Tampilan *Before*

#	Name	Leave Type	From	To	Days	Status	Details
1	John Doe	Annual	16/05/2024	16/05/2024	1	Pending	

(b) Tampilan *After*

Gambar 3.35. Tampilan halaman *Leave Management Before* dan *After*

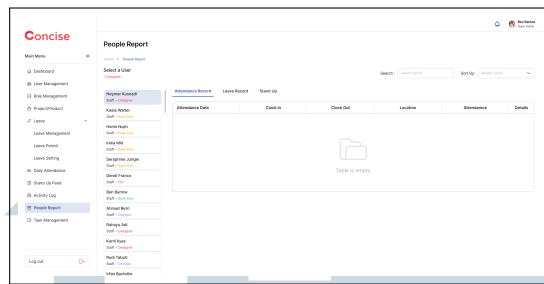
Gambar 3.35 (a) menunjukkan tampilan halaman *Leave Management* sebelum dilakukan *revamp*. Pada tampilan sebelumnya, proses peninjauan pengajuan dipisahkan antara cuti (*leave*) dan kerja dari rumah (*WFH*), serta terdapat

tab tambahan untuk melihat daftar riwayat pengajuan. Hal ini menyebabkan tampilan menjadi kurang intuitif. Selain itu, proses persetujuan (*approval*) dilakukan langsung melalui tabel dengan menggunakan *checkbox*, di mana pengguna harus memilih apakah pengajuan akan diterima atau ditolak. Mekanisme ini meningkatkan risiko ketidaktepatan dalam membaca detail pengajuan, serta memungkinkan terjadinya kesalahan dalam memilih *checkbox* saat melakukan persetujuan.

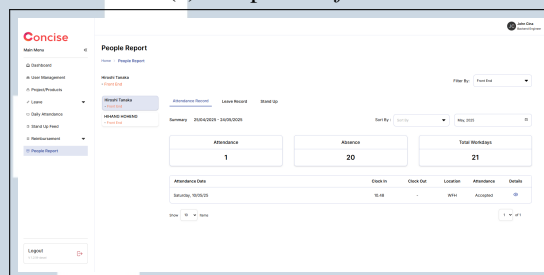
Gambar 3.35 (b) menunjukkan tampilan halaman *Leave Management* setelah dilakukan *revamp* pada bagian daftar (*list*). Pada tampilan baru ini, halaman difokuskan untuk menampilkan daftar pengajuan cuti karyawan. Daftar ini dikelola langsung oleh *supervisor* masing-masing karyawan berdasarkan struktur hirarki organisasi, sehingga dapat mengurangi penumpukan tugas pengelolaan pengajuan cuti oleh *superadmin*. Untuk melakukan persetujuan (*approval*), pengguna dapat menekan tombol *Detail* yang akan mengarahkan ke halaman *Leave Management Approval* untuk melakukan tinjauan lebih lanjut terhadap pengajuan tersebut.

Gambar 3.36. Tampilan *After Leave Management Approval*

Gambar 3.36 menunjukkan tampilan halaman *Leave Management* setelah dilakukan *revamp* pada bagian (*approval*). Pada tampilan baru ini, halaman difokuskan untuk melakukan peninjauan terhadap pengajuan cuti secara lebih teliti sebelum diberikan keputusan *accept* atau *reject*, sehingga dapat meminimalisir terjadinya kesalahan dalam proses persetujuan.



(a) Tampilan *Before*

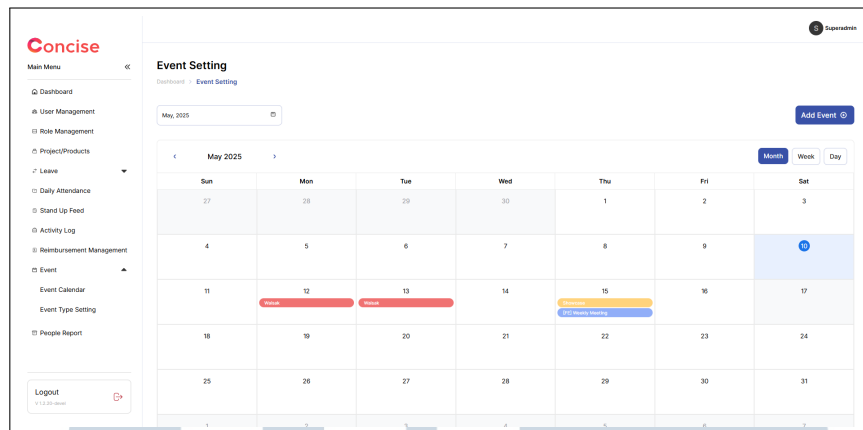


(b) Tampilan *After*

Gambar 3.37. Tampilan halaman *People Report Before* dan *After*

Gambar 3.37 (a) menunjukkan tampilan halaman *People Report* sebelum dilakukan pengembangan. Pada tampilan sebelumnya, belum tersedia fitur filter berdasarkan *job title*, sehingga menyulitkan dalam mencari nama karyawan sesuai bidang yang dituju. Selain itu, *attendance record* hanya menampilkan daftar absensi tanpa adanya *recap summary* bulanan, yang dapat menyulitkan karyawan dengan status gaji harian dalam melakukan perhitungan jumlah kehadiran.

Gambar 3.37 (b) menunjukkan tampilan halaman *People Report* setelah dilakukan pengembangan. Pada tampilan baru ini, tersedia fitur filter berdasarkan *job title*, sehingga memudahkan dalam mencari nama karyawan sesuai bidang yang dituju. Selain itu, *attendance record* kini dilengkapi dengan *recap summary* absensi bulanan yang mencakup jumlah kehadiran yang telah disetujui, jumlah ketidakhadiran, serta total hari kerja yang seharusnya dalam satu bulan.

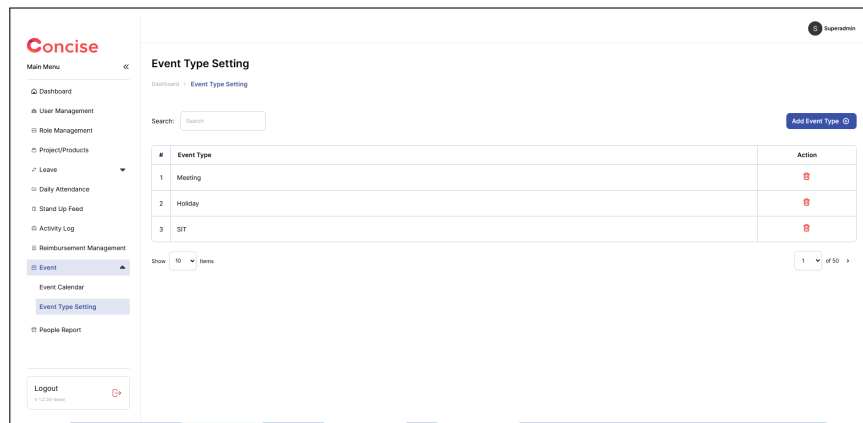


Gambar 3.38. Tampilan *Event Calendar*

Gambar 3.38 menunjukkan tampilan halaman *Event Calendar*. Pada fitur baru ini, terdapat *big calendar* yang dapat ditampilkan dalam tampilan bulanan, mingguan, maupun harian. Kalender ini memuat informasi terkait kegiatan seperti *meeting* dan jadwal libur. Tersedia tombol untuk membuat *event* baru yang akan mengarahkan pengguna ke halaman *Event Setting Form*. Selain itu, kegiatan *meeting* hanya dapat dilihat oleh karyawan yang namanya telah didaftarkan pada *event* tersebut.

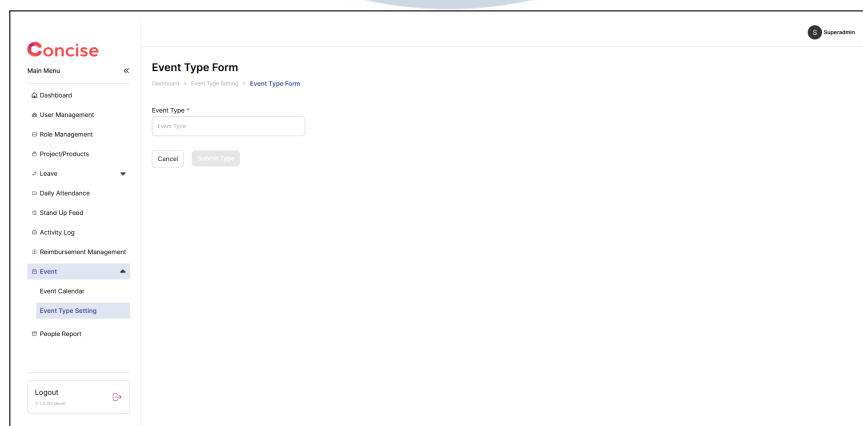
Gambar 3.39. Tampilan *Event Setting Form*

Gambar 3.39 menunjukkan tampilan halaman *Event Setting Form*. tersedia beberapa *form fields*, yaitu *Event Name*, *Event Type*, *Event Date*, *Event Time*, *Guest*, dan *Color*. Apabila *Event Type* yang dipilih adalah *Holiday*, maka *form fields* yang ditampilkan hanya *Event Name* dan *Event Date* saja. Selain itu, halaman ini juga berfungsi untuk mengedit *event* apabila diperlukan perubahan terhadap data *event* yang telah dibuat.



Gambar 3.40. Tampilan *Event Type List*

Gambar 3.40 menunjukkan tampilan halaman *Event Type* pada bagian daftar (*list*). Pada fitur baru ini, halaman menampilkan daftar *Event Type Name* yang hanya dapat dikelola oleh *Superadmin*. *Superadmin* dapat melakukan *action* seperti menghapus *Event Type* apabila sudah tidak diperlukan. Selain itu, tersedia tombol untuk menambahkan *Event Type* baru yang akan mengarahkan pengguna ke halaman *Event Type Form*.



Gambar 3.41. Tampilan *Event Type Form*

Gambar 3.41 menunjukkan tampilan halaman *Event Type Form* pada bagian formulir (*form*). Pada halaman ini tersedia *form field* untuk mengisi *Event Type Name*. Setiap *Event Type Name* harus bersifat unik, sehingga pengguna tidak dapat membuat nama *Event Type* yang sama dengan yang sudah ada.

3.6 Kendala dan Solusi yang Ditemukan

Beberapa kendala yang dialami selama melakukan magang di PT Ganda Visi Jayatama adalah sebagai berikut:

1. Permasalahan komunikasi antara tim *backend* dan tim *frontend* pada awal masa kerja magang menyebabkan ketidaksesuaian informasi, sehingga diperlukan penyesuaian terhadap perubahan API.
2. Kurangnya pengetahuan mengenai React Typescript pada awal pelaksanaan magang, sehingga membutuhkan waktu lebih pada pengerjaan tahap awal.

Solusi yang ditemukan untuk menghadapi masalah tersebut adalah sebagai berikut:

1. Meningkatkan komunikasi dan koordinasi antara tim *backend* dengan *frontend* untuk menyesuaikan informasi.
2. Mempelajari dan memahami penggunaan React Typescript secara mendalam, serta melakukan pencarian informasi tambahan untuk mengetahui *best practice* yang tepat dalam pengembangan aplikasi menggunakan React Typescript

