

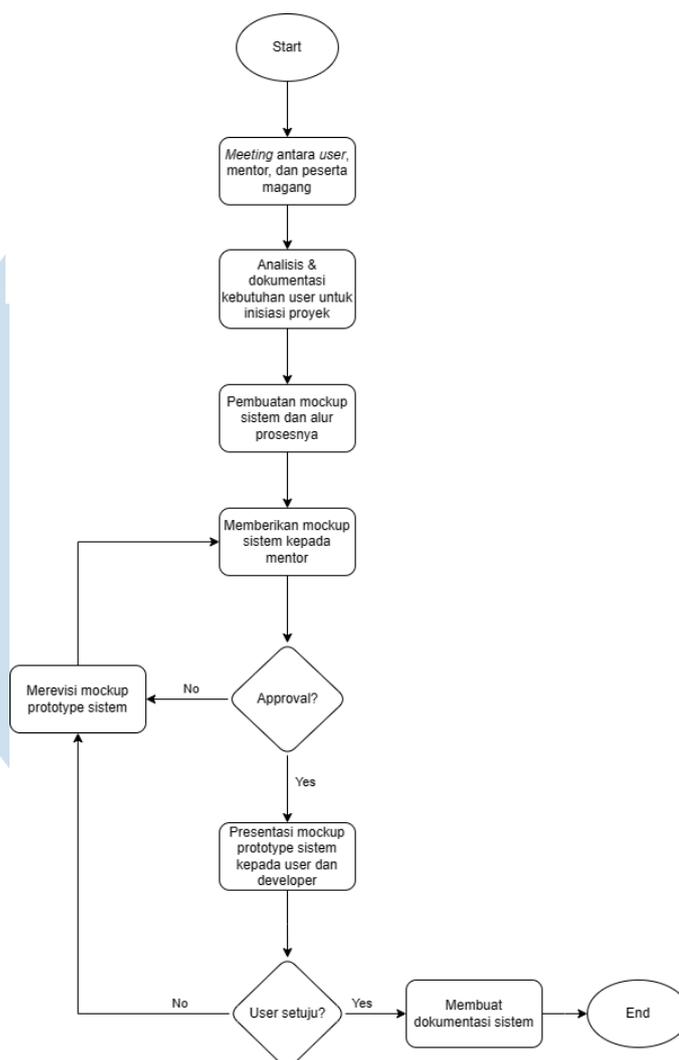
BAB III

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Peserta magang dengan posisi *junior business analyst* pada departemen *IT Business Process and Project* di PT XYZ memiliki kedudukan sebagai bawahan dari senior manager atau kepala departemen *IT Business Process and Project* itu sendiri. Dalam posisi ini, tugas peserta magang dengan posisi *junior business analyst* adalah untuk membantu tim dari departemen *IT Business Process and Project* dalam proyek - proyek IT yang ada, terutama untuk menganalisis kebutuhan bisnis di dalam proyek tersebut, kemudian mendokumentasikan kebutuhannya, dan membantu dalam perancangan arsitektur sistem, tampilan *prototype* sistem serta alur prosesnya agar sesuai dengan kebutuhan bisnis, yang diawasi langsung oleh kepala beserta timnya dari departemen *IT Business Process and Project* selaku mentor magang. Berada di bawah kepemimpinan Kepala Departemen *IT Business Process and Project*, peserta magang diberikan peluang untuk memperoleh pengetahuan mendalam mengenai proses bisnis dalam lingkup perusahaan media. Sebagai *junior business analyst*, peserta magang akan berpartisipasi secara langsung dalam berbagai proyek IT yang ada, untuk berkontribusi dalam pembuatan prototipe sistem dan pembuatan dokumentasinya yang sesuai dengan standar yang telah ditentukan. Peserta magang juga akan terlibat dalam pertemuan dengan pengguna dan tim pengembang untuk membahas kebutuhan bisnis, mulai dari tahap awal inisiasi proyek hingga tahap akhir untuk mempresentasikan prototipe sistem kepada pengguna maupun tim pengembang.

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 3.1 Alur kerja peserta magang

Gambar 3.1 merupakan alur kerja dari *junior business analyst*. Alur kerja ini mencakup dua tahapan atau fase, yaitu fase pertama untuk inisiasi proyek dan fase kedua untuk pelaksanaan proyek. Dalam tahap inisiasi proyek, hal pertama yang dilakukan adalah melakukan meeting antara *user*, mentor, dan peserta magang. Peserta magang *junior business analyst* diikutsertakan didalam meeting ini agar dapat ikut menganalisa dan memahami kebutuhan *user* untuk pembuatan suatu sistem. Pada tahap ini dilakukan *user requirement gathering* dimana *user* mengutarakan permasalahan dan sistem yang diinginkan untuk mengatasi permasalahan bisnis yang dimiliki. Mentor akan memberikan contoh solusi sistem kepada *user*, dan setelah *user* menyetujuinya, maka *junior business analyst* akan membuat dokumentasi awal sebagai landasan untuk membuat tampilan sistem yang

sesuai. Setelah itu, mentor akan memberikan penugasan untuk membuat tampilan awal atau *mockup* dari sistem yang dibutuhkan, serta alur proses berjalannya sistem tersebut, dan skenario – skenario penggunaan sistem berdasarkan kebutuhan *user*. Tampilan awal atau *mockup* sistem dibuat menggunakan *tool* Figma.

Pada tahap selanjutnya dilakukan pengerjaan pembuatan tampilan sistem. Selama proses pengerjaan ini, koordinasi antara mentor dan peserta magang selalu dijalankan agar proyek dapat selesai dengan benar dan tidak ada keterlambatan. Koordinasi dan komunikasi dilakukan melalui beberapa *platform*, seperti WhatsApp dan Microsoft Teams. WhatsApp berfungsi sebagai saluran komunikasi dengan mentor dan dengan anggota tim *junior business analyst*, sedangkan Microsoft Teams digunakan sebagai sarana untuk melakukan meeting mingguan internal yang dilakukan setiap hari jumat untuk berbagi update terkini mengenai proyek yang sedang dikerjakan.

Setelah proses pengerjaan pembuatan tampilan sistem beserta alur dan skenario penggunaannya sudah selesai, maka akan diberikan dan dipresentasikan kepada mentor untuk persetujuan dan dievaluasi ulang. Apabila masih ada bagian yang belum sesuai, maka revisi akan dilakukan untuk memperbaikinya. Setelah mentor mengevaluasi tampilan *mockup prototype* sistem, maka presentasi final akan dilakukan untuk mempresentasikan tampilan tersebut kepada *user* untuk menjelaskan cara penggunaan sistem dan alur prosesnya. Selain dipresentasikan kepada *user*, tampilan *prototype* sistem juga akan dipresentasikan kepada tim *developer* atau pengembang. Tujuannya adalah untuk mengukur kapabilitas tim pengembang dalam mengembangkan dan mengimplementasikan sistem yang akan dibuat. Pada tahap ini, apabila *user* belum setuju dengan sistem yang diinginkan, maka umpan balik dari *user* atau pengguna akan dicatat untuk didiskusikan kembali bersama mentor. Setelah *user* setuju dengan sistemnya dan tim pengembang dapat memahami sistem yang akan dibuat, maka pembuatan dokumentasi sistem akan dikerjakan. Dokumentasi yang dibuat dibagi menjadi dua tipe, yaitu *Functional Specification Document (FSD)* dan *Change Request Document (CR)*. *Functional Specification Document* merupakan dokumentasi yang dibuat untuk pembuatan sistem baru yang belum ada sebelumnya, sedangkan *Change Request Document*

dibuat untuk penambahan, perubahan, atau pembaruan fitur dari sistem yang sudah ada sebelumnya.

3.2 Tugas dan Uraian Kerja Magang

junior business analyst bertanggung jawab untuk menganalisis, memahami, dan mendokumentasikan kebutuhan bisnis, serta merancang arsitektur desain/topologi sistem dan alur proses dari sistem yang akan dibuat. Selain itu, *junior business analyst* juga bertanggung jawab dalam pembuatan tampilan prototipe atau blueprint sistem dan/atau aplikasi yang akan dikembangkan, serta mempresentasikan hasil rancangan sistem kepada pengguna dan pengembang. Tujuan utama dari tanggung jawab ini adalah untuk memastikan keselarasan antara sistem dan aplikasi yang dibangun dengan kebutuhan bisnis pengguna. Selama menjalani magang di PT XYZ, peserta magang dari posisi *junior business analyst* diberikan sejumlah aktivitas dan tanggung jawab yang harus dijalankan. Berikut ini adalah ringkasan umum dari uraian beberapa aktivitas dan tugas yang dilaksanakan selama proses magang berlangsung:

Tabel 3.1 Tabel uraian kerja magang

No.	Aktivitas	Periode Mulai	Periode Selesai	Minggu Ke-
1	Pengenalan lingkungan kerja perusahaan	16 Februari 2024	19 Februari 2024	1-2
2	Mengerjakan proyek integrasi e-career	20 Februari 2024	4 Maret 2024	1-3
3	Mengerjakan proyek sistem <i>inventory movement</i>	5 Maret 2024	27 Maret 2024	3-6
4	Mengerjakan proyek <i>blast email</i>	28 Maret 2024	26 April 2024	6-10
5	Mengerjakan proyek IT Management System Dashboard	29 April 2024	27 Mei 2024	11-14

Berdasarkan Tabel 3.1, berikut merupakan penjelasan mengenai uraian pekerjaan magang secara lebih rinci:

3.2.1 Pengenalan Lingkungan Kerja Perusahaan (Minggu 1 - 2)

Pada awal periode magang, para peserta magang di PT XYZ menjalani acara On-Boarding yang dilakukan secara daring selama dua hari. Acara ini diadakan oleh PT XYZ untuk memberikan pemahaman yang mendalam tentang lingkungan kerja dan budaya perusahaan kepada peserta magang. Selama dua hari, terdapat beberapa sesi yang diselenggarakan, seperti sesi sambutan dari divisi *Human Resources Development* yang memberikan sambutan bagi para peserta magang yang lolos, dan pemaparan mengenai divisi HRD dalam pengembangan karyawan dan peserta magang, yang dilanjutkan dengan sesi penjelasan ekosistem dan gambaran bisnis yang memberikan gambaran luas tentang ekosistem bisnis perusahaan dalam industri media beserta entitas bisnis lainnya yang dinaungi oleh PT XYZ. Sesi berikutnya dilanjutkan oleh sesi yang membahas cara mengelola beban kerja dan memaksimalkan pengalaman *mentoring* antara peserta magang dan mentor atau supervisi magang selama periode magang, dan sesi yang menjelaskan secara detail tentang program magang merdeka, tujuan, manfaat, dan harapan perusahaan terhadap para peserta magang. Sesi terakhir adalah sesi untuk menjelaskan atau memberikan gambaran dari semua divisi di perusahaan, sehingga memberikan pemahaman yang lebih baik kepada peserta magang tentang bagaimana setiap divisi bekerja sama untuk mencapai tujuan perusahaan. Acara ini penting dan wajib diikuti oleh seluruh peserta magang PT XYZ untuk memastikan para peserta magang memahami dan siap untuk berkontribusi dalam lingkungan kerja perusahaan.

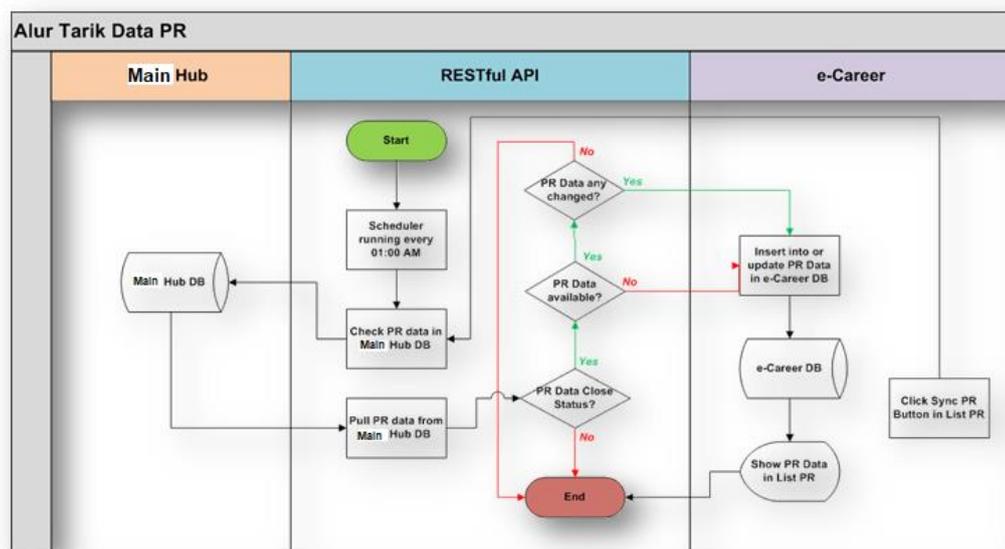
3.2.2 Mengerjakan Proyek Integrasi E-Career (Minggu 1 - 3)

Proyek pertama yang dikerjakan dalam periode magang ini adalah proyek untuk Integrasi *Website* E-Career dengan Sistem Main Hub yang dimiliki oleh PT XYZ. Proyek Integrasi *Website* E-Career dengan Sistem Main Hub merupakan proyek perubahan atau penambahan fitur dari sistem yang sudah ada sebelumnya pada aplikasi *Super Admin* PT XYZ. Saat ini, sistem *Super Admin* yang ada belum memiliki fitur untuk menarik data *Personal Request* (PR) dan mengirimkan data *Applicants* yang diterima ke Sistem Main Hub perusahaan dari menu *Vacancies* pada sub menu *Talent Acquisition Applicants*. Selain itu, sistem juga belum

memiliki audit log untuk mencatat aktivitas penambahan, perubahan, atau penghapusan data. Proyek ini diinisiasi berdasarkan kebutuhan *user* yang berasal dari permasalahan bisnis pada divisi *Human Resource Development* (HRD).

3.2.2.1 Pemahaman Perubahan Sistem Yang Diusulkan

Tujuan utama dari proyek ini adalah untuk mengintegrasikan dua sistem, yaitu Sistem Main Hub perusahaan pada modul *Personal Request* dengan e-career menggunakan sebuah *API*. Sehingga hasil yang diharapkan oleh *user* dari integrasi ini adalah untuk memungkinkan penarikan data PR dari Sistem Main Hub perusahaan ke *website* e-career dan pengiriman data pelamar yang diterima di menu *Vacancies* dari e-career ke Sistem Main Hub perusahaan melalui media perantara aplikasi *Super Admin*. Proses integrasi antar sistem yang diusulkan digambarkan melalui *flowchart* pada gambar 3.2.

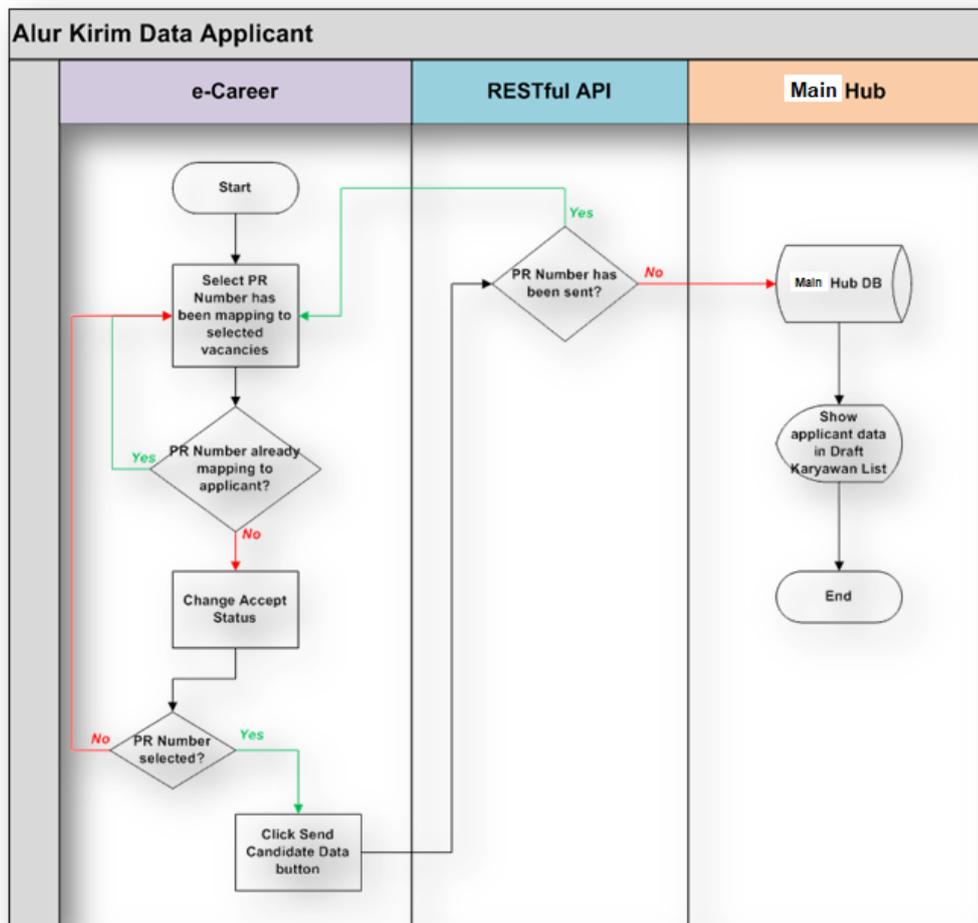


Gambar 3.2 *Flowchart* alur menarik data *Personal Request*

Untuk memahami pembuatan tampilan sistem, maka *junior business analyst* perlu untuk memahami alur atau proses data yang akan berjalan dalam integrasi sistem. *Flowchart* pada gambar 3.2 menggambarkan proses otomatis yang melibatkan Main Hub perusahaan, *API*, dan sistem e-career. Proses ini dimulai dengan scheduler di Main Hub perusahaan yang berjalan untuk memeriksa dan menarik data PR dari *database* Main Hub perusahaan. Data ini kemudian dikirim

ke *API* untuk memeriksa apakah ada perubahan. Jika ada perubahan atau data PR sebelumnya tidak tersedia, data tersebut akan dimasukkan atau diperbarui di *database* e-career. Jika tidak ada perubahan atau status penutupan data PR tidak terpenuhi, proses berakhir. Sehingga, proses ini memastikan bahwa data PR selalu up-to-date dan dapat diakses oleh pengguna di sistem e-career.

Dari alur sistem yang sudah dibuat sebelumnya, setiap pembuatan *Vacancies* atau lowongan kerja mewajibkan pengisian nomor PR, dimana nomor PR ini akan berfungsi sebagai penanda untuk setiap lowongan dan pelamar yang diterima untuk pengiriman data ke sistem Main Hub perusahaan. Proses pengiriman data pelamar digambarkan melalui *flowchart* pada gambar 3.3.



N U S A N T A R A
Gambar 3.3 *Flowchart* alur mengirim data pelamar

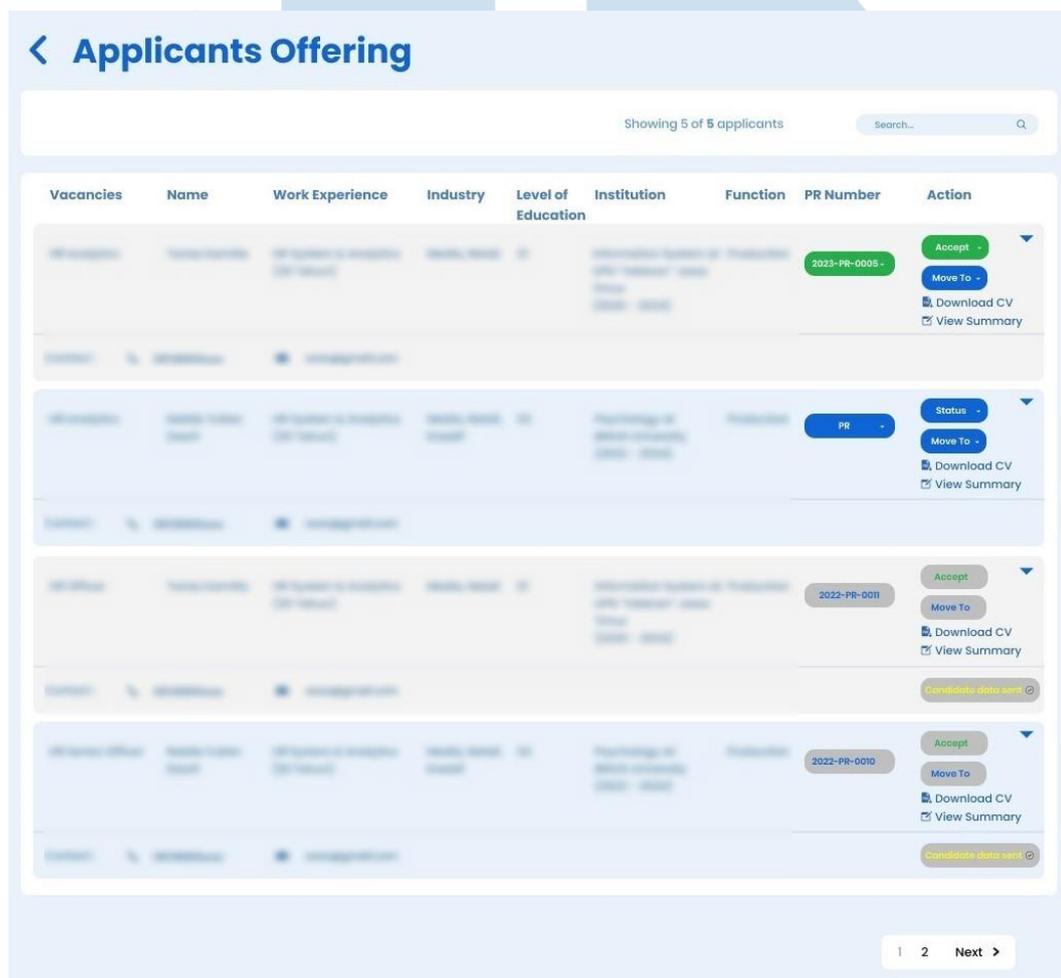
Flowchart pada gambar 3.3 menggambarkan proses pengiriman data pelamar dari e-career ke Main Hub Perusahaan. Proses ini dimulai di e-career dengan memilih Nomor PR yang telah dipetakan atau di-mapping ke lowongan yang dipilih. Jika Nomor PR sudah dipetakan ke pelamar, status diubah menjadi terima, dan jika Nomor PR dipilih, data kandidat akan dikirim. Selanjutnya, melalui *API*, akan dilakukan pengecekan apakah Nomor PR telah dikirim. Jika ya, data akan berakhir di *database* Main Hub Perusahaan, jika tidak, data pelamar ditampilkan dalam Draft Karyawan List.

3.2.2.2 Pengerjaan Tampilan pada sistem *Super Admin* dan dokumentasi

Setelah memahami alur proses di dalam sistem, maka masuk di dalam tahap pengerjaan. Dalam proyek ini, *junior business analyst* ditugaskan untuk membantu melanjutkan pembuatan tampilan sistem atau *mockup* sistem dari sistem *Super Admin* yang sudah ada sebelumnya. Fokus pembuatan tampilan yang dimaksud adalah penambahan fitur untuk mengirimkan *email* kepada pelamar yang lolos seleksi (diterima) berdasarkan data PR yang sudah di mapping sebelumnya dengan tujuan untuk mempermudah pemantauan data pelamar pada sistem *Super Admin* perusahaan yang akan dikirimkan ke sistem Main Hub perusahaan. Selain dari tugas pembuatan penambahan desain pada *mockup* sistem yang telah ada, *junior business analyst* juga bertanggung jawab dalam melakukan dokumentasi yang mendalam dan rinci. Tugas ini mencakup penulisan alur sistem dengan jelas dan penjelasan detail dari tampilan antarmuka yang akan pengguna. Dokumentasi ini akan menjadi referensi kunci yang dapat digunakan oleh tim pengembangan sebagai panduan dalam mengembangkan fitur tersebut.

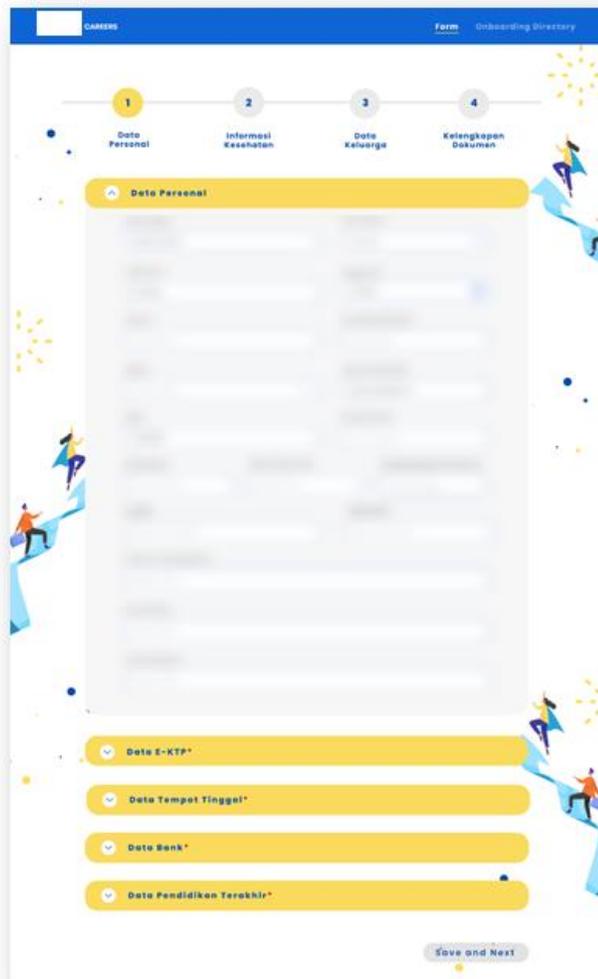
Pembuatan tampilan untuk fitur pengiriman *email* kepada pelamar dimulai pada menu *Applicants Offering* seperti pada gambar 3.4. Pada menu ini, terdapat beberapa menu *header*, yaitu *Vacancies*, *Name*, *Work Experience*, *Industry*, *Level of Education*, *Institution*, dan *Function* dimana data pada menu *header* ini diambil dari data pada *website* e-career. Sedangkan untuk *header* menu *PR Number* dan *Action* merupakan data yang ter-generate di dalam sistem *Super Admin*. Perubahan tampilan yang dilakukan disini adalah pada *header Action*, dimana saat *user* ingin

mengirimkan *email* kepada pelamar, maka *user* dapat melakukan aksinya melalui tombol hijau dengan merubah status *offering* pelamar menjadi *Accept*. Dengan begitu, sistem *Super Admin* dapat mengirimkan *email* kepada pelamar yang berisikan *link* untuk melengkapi proses pendaftaran pelamar.



Gambar 3.4 Tampilan menu *applicants offering*

Setelah pelamar menerima *email* dan membuka *link* yang diberikan pada *email* tersebut, maka pelamar akan diarahkan kepada *website* e-career untuk mengisi kelengkapan data lanjutan. Tampilan dari *Website* e-career ditunjukkan pada gambar 3.5.



Gambar 3.5 Tampilan *website* e-career

Gambar 3.5 menunjukkan tampilan dari *website* e-career yang dibuat. *Website* ini merupakan situs untuk mengisi kelengkapan data pelamar. Kelengkapan data pelamar yang perlu untuk diisikan dalam situs ini mencakup data personal, informasi kesehatan, data keluarga, dan kelengkapan dokumen tambahan. Permasalahan yang ada pada *website* e-career ini adalah tampilan ini belum memiliki dokumentasi yang lengkap. Oleh karena itu, *junior business analyst* bertugas untuk membuat dokumentasi pada *website* ini dengan fokus pada penjelasan *Graphical User Interface* (GUI) dan setiap *field* yang ada di dalamnya secara detail. Ringkasan dokumentasi dari setiap *field* dapat dilihat pada tabel 3.2.

Tabel 3.2 Detail keterangan masing-masing *field*

Jenis <i>Field</i>	Keterangan	Validasi
<i>Text Field</i>	<ul style="list-style-type: none"> • Mandatory • Non Mandatory atau Optional 	<ul style="list-style-type: none"> • Validasi pilihan jenis kelamin • Validasi pilihan agama • Validasi nomor telepon yang valid • Validasi alamat <i>email</i> yang valid • Validasi untuk pilihan <i>dropdown</i>
<i>Date Picker</i>		
<i>Dropdown</i>		

Berdasarkan tabel 3.2, dokumentasi setiap *field* pada halaman *website* e-career dibagi ke dalam tiga jenis kategori, yaitu *Text Field*, *Date Picker*, dan *Dropdown*. *Text Field* merupakan *field* kosong yang dapat diisi oleh teks dengan cara diketik oleh pengguna. *Date Picker* merupakan *field* yang menampilkan tanggal secara detail, mulai dari hari, bulan, dan waktu dimana pengguna dapat memilih tanggal pada *field* ini. Sedangkan *Dropdown* merupakan *field* yang berisikan daftar beberapa pilihan, dimana pengguna hanya bisa memilih satu pilihan dari daftar pilihan yang ada. Selain jenis *field*, dokumentasi setiap *field* juga mencakup keterangan dari masing-masing *field* yang ada. Keterangan dibagi menjadi dua kategori yaitu Mandatory dan Non-Mandatory. *Field* dengan keterangan Mandatory memiliki arti bahwa *field* tersebut wajib diisi oleh pelamar. *Field* ini ditandai dengan tanda bintang berwarna merah (*). Sedangkan *field* dengan keterangan Non-Mandatory memiliki arti bahwa *field* tersebut tidak wajib atau opsional untuk diisi. Terkadang, keterangan *field* dapat berubah tergantung kondisi pengisian form oleh pelamar. Terdapat beberapa *field* yang sebelumnya bersifat Non-Mandatory akan berubah menjadi Mandatory, bergantung dengan pemilihan *field dropdown* sebelumnya. Terakhir, dokumentasi setiap *field* juga mencakup validasi dari setiap *field* yang ada. Beberapa validasi diperlukan agar pengisian *field* dapat mengikuti aturan validasi yang sudah diberlakukan, sehingga data yang diisi tidak akan salah. Beberapa validasi yang diterapkan adalah validasi pilihan jenis kelamin, yaitu hanya pria dan wanita pada *field* jenis kelamin, validasi

pilihan agama, validasi nomor telepon yang valid dengan minimal digit yang harus ditulis, validasi alamat *email* yang valid dengan domain *email* yang benar, dan validasi untuk pilihan *dropdown* yang berisikan daftar pilihan yang dapat dipilih oleh pengguna. Contoh dokumentasi *field* pada dokumen *Change Request* (CR) digambarkan pada gambar 3.6.

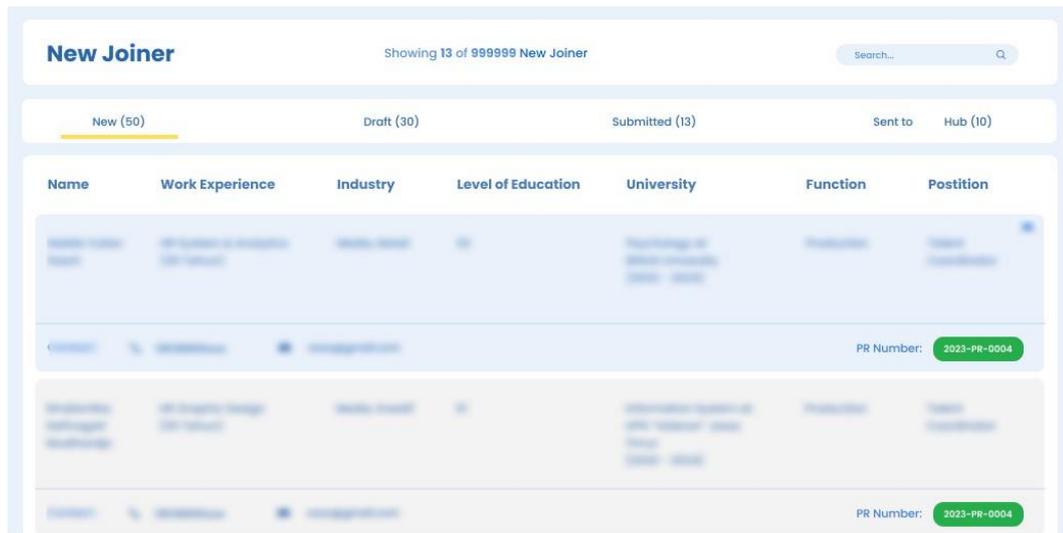
Change Request

Berikut detail dari GUI form Data Personal:

Field	Jenis	Keterangan
Nama Lengkap	TextField	<ul style="list-style-type: none"> • Mandatory
Jenis Kelamin	Dropdown	<ul style="list-style-type: none"> • Mandatory • Validasi: Pilihan hanya "Laki-Laki" atau "Perempuan"
Tempat Lahir	TextField	<ul style="list-style-type: none"> • Mandatory
Tanggal Lahir	DatePicker	<ul style="list-style-type: none"> • Mandatory
Anak Ke	TextField	<ul style="list-style-type: none"> • Mandatory • Validasi: Harus diisi dengan angka
Dari Berapa Bersaudara	TextField	<ul style="list-style-type: none"> • Mandatory • Validasi: Harus diisi dengan angka
Agama	Dropdown	<ul style="list-style-type: none"> • Mandatory • Validasi: Pilihan hanya dari daftar yang tersedia: Islam / Protestan / Katolik / Hindu / Buddha / Konghucu / Others (harus diisi dengan Text)
Alamat E-Mail Pribadi	TextField	<ul style="list-style-type: none"> • Mandatory • Validasi: Harus diisi dengan format email yang valid (contoh: example@email.com)
No HP	TextField	<ul style="list-style-type: none"> • Mandatory • Validasi: Harus diisi dengan format nomor telepon yang valid (contoh: 08123456789)
No Telpn Rumah	TextField	<ul style="list-style-type: none"> • Optional • Validasi: Harus diisi dengan format nomor telepon yang valid (contoh: 08123456789)

Gambar 3.6 Contoh tabel dokumentasi *field website* e-career pada dokumen

Setelah pembuatan dokumentasi untuk *website* e-career, selanjutnya *junior business analyst* bertugas untuk membuat penambahan menu baru pada *Super Admin*. Menu baru ini bernama *New Joiner*, dimana menu ini berfungsi sebagai menu yang menampung data lanjutan pelamar yang sudah mengisi kelengkapan form melalui *website* e-career.



Gambar 3.7 Tampilan menu new joiner

Gambar 3.7 menampilkan menu *new joiner*. Pembuatan tampilan menu *new joiner* ini mempertimbangkan proses kelengkapan data pelamar yang diisi oleh pelamar pada *website* e-career. Oleh karena itu, pada menu ini dibuat empat status berdasarkan status nyata dari proses pengisian data yang sedang atau sudah diisi oleh pelamar. Detail dari setiap status dijelaskan dalam tabel 3.3.

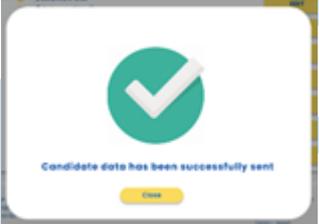
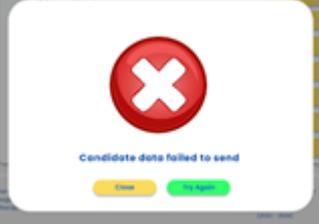
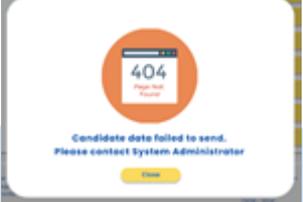
Tabel 3.3 Keterangan status di dalam menu *new joiner*

Status	Keterangan
<i>New</i>	Berisikan daftar pelamar yang baru membuka form data lengkap tetapi belum mengisi.
<i>Draft</i>	Berisikan daftar pelamar yang baru mengisi sebagian data atau belum selesai mengisi pada form data lengkap.
<i>Submitted</i>	Berisikan daftar pelamar yang sudah mengisi data secara lengkap data dan melakukan submit. Pada halaman ini, admin dapat mengecek ulang data pelamar agar sesuai dengan kriteria yang diperlukan.
<i>Sent to Main Hub</i>	Berisikan daftar pelamar yang datanya sudah divalidasi secara menyeluruh oleh admin dan sudah dikirimkan ke sistem Main Hub PT XYZ.

Terakhir, data pelamar yang sudah lengkap diverifikasi akan dikirim kepada sistem Main Hub perusahaan. Namun, proses pengiriman data yang menggunakan

API ini bisa saja tidak selalu berjalan lancar. Oleh karena itu, penting untuk membuat suatu antarmuka *error handling* yang dapat menampilkan kode *error* untuk setiap jenis *error* yang mungkin terjadi. Beberapa tampilan untuk *error handling* yang dibuat dijelaskan pada tabel berikut:

Tabel 3.4 Keterangan *error handling* saat pengiriman data

Tampilan Pop Up Error Handling	Keterangan
 <p data-bbox="323 835 820 869">Gambar 3.8 Tampilan pengiriman data sukses</p>	<p data-bbox="847 611 1310 719">Proses pengiriman data sukses atau respon REST API 200 (responseCode: 200 OK)</p>
 <p data-bbox="323 1126 820 1160">Gambar 3.9 Tampilan pengiriman data gagal</p>	<p data-bbox="847 902 1334 1010">Proses pengiriman data gagal dengan respon REST API 400 (responseCode: 400 Bad request)</p>
 <p data-bbox="323 1395 820 1462">Gambar 3.10 Tampilan pengiriman data gagal karena data tidak ditemukan</p>	<p data-bbox="847 1193 1334 1328">Proses pengiriman data gagal dengan respon REST API 404 (responseCode: 404 Resource not found)</p>
 <p data-bbox="323 1697 820 1765">Gambar 3.11 Tampilan pengiriman data gagal karena server error</p>	<p data-bbox="847 1496 1334 1630">Proses pengiriman data gagal dengan respon REST API 500 (responseCode: 500 Internal server error)</p>

3.2.3 Mengerjakan Proyek Sistem *Inventory Movement* (Minggu 3 - 6)

Proyek kedua yang dikerjakan adalah proyek pembuatan sistem *inventory movement*. Proyek sistem *inventory movement* merupakan proyek baru yang dibuat berdasarkan kebutuhan *user* yang berasal dari permasalahan bisnis pada divisi *warehouse* dari salah satu entitas bisnis yang dinaungi oleh PT XYZ.

3.2.3.1 *User Requirement Gathering*

Karena proyek ini merupakan pembuatan sistem yang baru, maka untuk memulai proyek ini, dilakukan *User Requirement Gathering* terlebih dahulu. Proses pengumpulan kebutuhan pengguna atau *user requirement gathering* adalah proses di mana kita berusaha untuk memahami pengguna secara lebih mendalam, sehingga dapat mengidentifikasi kebutuhan pengguna yang belum tercapai, dan menentukan kebutuhan mana yang dapat diakomodasi. Proses ini sangat penting dalam menciptakan *user experience* yang baik pada sistem yang dihasilkan, karena memungkinkan kita untuk menyesuaikan sistem atau produk yang kita buat agar selaras dengan persyaratan kebutuhan dan tujuan pengguna [6]. Dengan demikian, sistem atau produk tersebut akan menjadi lebih bermanfaat bagi pengguna dalam mencapai tujuan mereka.

Tahap *user requirements gathering* pada proyek ini dilakukan melalui meeting secara luring bersama dengan *user*. Selain mentor utama, peserta magang *junior business analyst* juga dilibatkan di dalam meeting ini. Dari diskusi dan pembahasan yang dilakukan melalui meeting ini, diketahui permasalahan bisnis yang sedang dialami oleh *user* yaitu belum adanya sistem yang dapat melacak dan memantau perpindahan aset dari satu gudang ke gudang lainnya, dimana aset-aset ini banyak digunakan untuk keperluan sebagai properti hingga sebagai alat renovasi.

Berdasarkan hasil diskusi yang dilakukan, kebutuhan *user* yang dapat disimpulkan dari permasalahan ini adalah dibutuhkan sebuah sistem manajemen perpindahan aset yang berbasis web dan dapat digunakan dengan tampilan *mobile* yang bersifat *mobile-friendly*. Sistem ini harus dapat mengelola berbagai pencatatan dan perpindahan aset dengan detail seperti nama, kategori, lokasi, jumlah, foto,

status, dan kode aset yang merujuk kepada kode SAP. Kategori aset mencakup alat pertukangan, multimedia, dan properti. Status aset dapat berupa bagus, rusak, atau perlu diperbaiki. Lokasi aset mencakup gudang-gudang dimana aset sering berpindah tangan. Sistem ini juga harus memiliki fitur master barang dan transaksi peninjauan masuk dengan detail program, PIC barang, durasi peminjaman, dan lokasi peminjaman. Sistem ini juga harus dapat mengirim konfirmasi peminjaman dan *email approval* untuk konfirmasi peminjaman, dimana setiap aset yang dipinjamkan harus memiliki dokumentasi.

3.2.3.2 Pembuatan Mockup *Inventory Movement*

Berdasarkan kebutuhan *user* yang sudah dijabarkan sebelumnya, *junior business analyst* bertugas untuk membuat tampilan awal atau *mockup* sistem *Inventory Movement* sebelum dikembangkan menjadi aplikasi yang dapat digunakan. Rancangan awal untuk sistem ini berfokus pada tiga menu utama yang akan digunakan oleh pengguna, yaitu menu *Master Asset*, *Movement Management*, dan *Audit Log*.

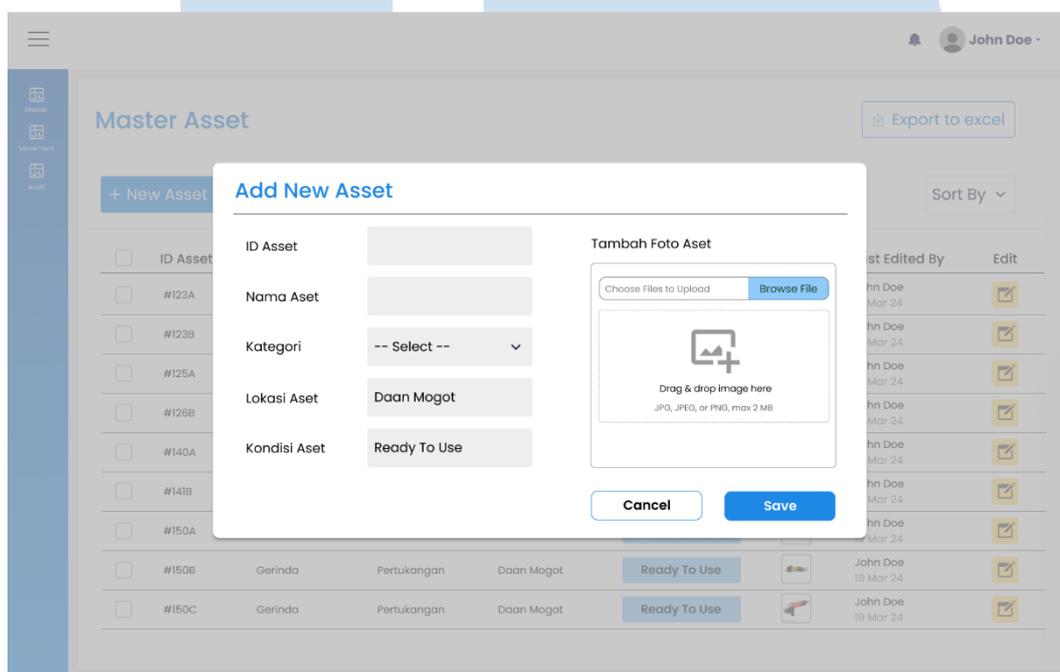
The screenshot shows a web application interface for 'Master Asset'. It includes a sidebar with navigation icons for 'Master', 'Movement', and 'Audit'. The main content area has a title 'Master Asset', an 'Export to excel' button, a '+ New Asset' button, a search bar, and a 'Sort By' dropdown. Below is a table listing assets with columns: ID Asset, Nama, Kategori, Lokasi, Kondisi, Foto, Last Edited By, and Edit.

<input type="checkbox"/>	ID Asset	Nama	Kategori	Lokasi	Kondisi	Foto	Last Edited By	Edit
<input type="checkbox"/>	#123A	Waterpass	Pertukangan	Daan Mogot	Ready To Use		John Doe 19 Mar 24	
<input type="checkbox"/>	#123B	Waterpass	Pertukangan	Daan Mogot	Ready To Use		John Doe 19 Mar 24	
<input type="checkbox"/>	#125A	Genset	Pertukangan	Daan Mogot	Ready To Use		John Doe 19 Mar 24	
<input type="checkbox"/>	#126B	Genset	Pertukangan	Daan Mogot	Ready To Use		John Doe 19 Mar 24	
<input type="checkbox"/>	#140A	Table Saw	Pertukangan	Daan Mogot	Ready To Use		John Doe 19 Mar 24	
<input type="checkbox"/>	#141B	Table Saw	Pertukangan	Daan Mogot	Ready To Use		John Doe 19 Mar 24	
<input type="checkbox"/>	#150A	Gerinda	Pertukangan	Daan Mogot	Ready To Use		John Doe 19 Mar 24	
<input type="checkbox"/>	#150B	Gerinda	Pertukangan	Daan Mogot	Ready To Use		John Doe 19 Mar 24	
<input type="checkbox"/>	#150C	Gerinda	Pertukangan	Daan Mogot	Ready To Use		John Doe 19 Mar 24	

Gambar 3.12 Tampilan menu *master asset*

Gambar 3.12 merupakan tampilan utama dari menu *master asset* yang dibuat berdasarkan kebutuhan *user* sebelumnya. Menu ini berguna sebagai menu

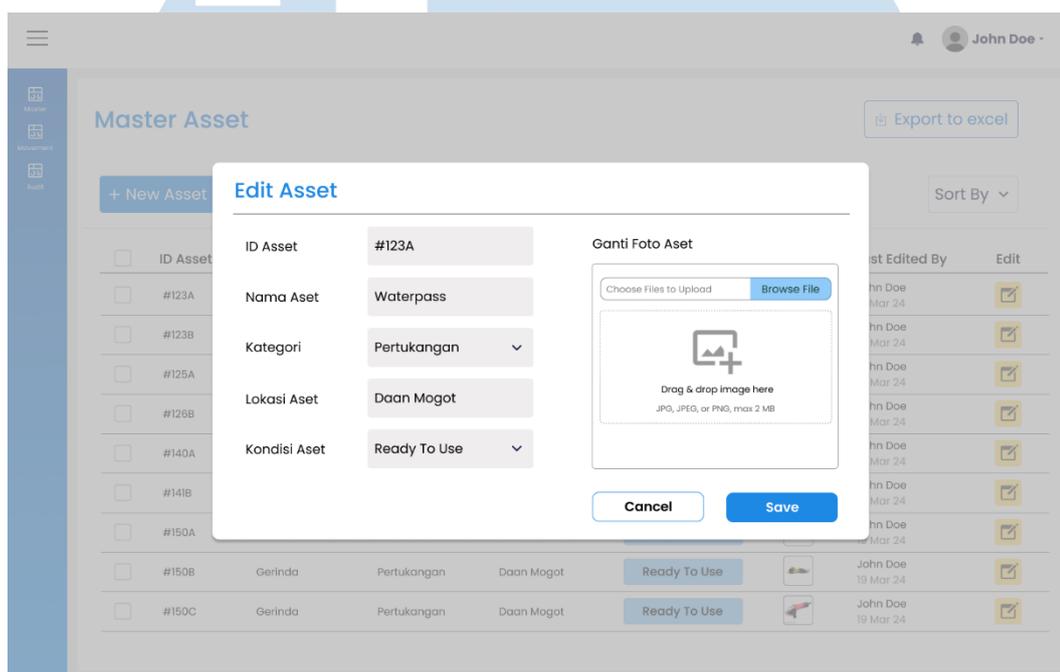
utama, dimana pada menu ini pengguna dapat melihat dan memantau setiap aset yang dimiliki beserta informasi penting lainnya, terutama status atau kondisi aset saat ini. Informasi penting lainnya pada daftar aset ini adalah ID Aset, Nama, Kategori, Lokasi, Kondisi, Foto terkini barang, dan *Last Edited By*. Di dalam menu ini, pengguna dapat menambahkan aset baru melalui tombol *New Asset*, melakukan pencarian aset, mengurutkan aset berdasarkan kategori, kondisi, maupun lokasi aset, dan meng-*edit* masing-masing aset.



Gambar 3.13 Tampilan fitur tambah aset

Gambar 3.13 merupakan tampilan dari fitur untuk menambah aset baru. Pengguna dapat menambahkan ID Aset, Nama Aset, Kategori Aset, Lokasi Aset, Kondisi Aset, dan foto aset. *Field* kategori aset memiliki tipe *dropdown*, dikarenakan kategori dari setiap barang yang akan dimasukkan ke dalam menu ini sudah ditentukan dari awal, sehingga pengguna sistem hanya dapat memilih sesuai dengan barang yang dimaksud dan tidak dapat menambahkan kategori secara mandiri. Selain itu, dikarenakan aksi ini terhitung sebagai aksi untuk menambah barang baru, maka *field* kondisi aset akan terisi secara otomatis dengan status *ready to use* dengan asumsi barang yang dimasukkan ke dalam menu ini merupakan barang yang tidak rusak dan siap dipakai. Jika sudah selesai, maka pengguna dapat

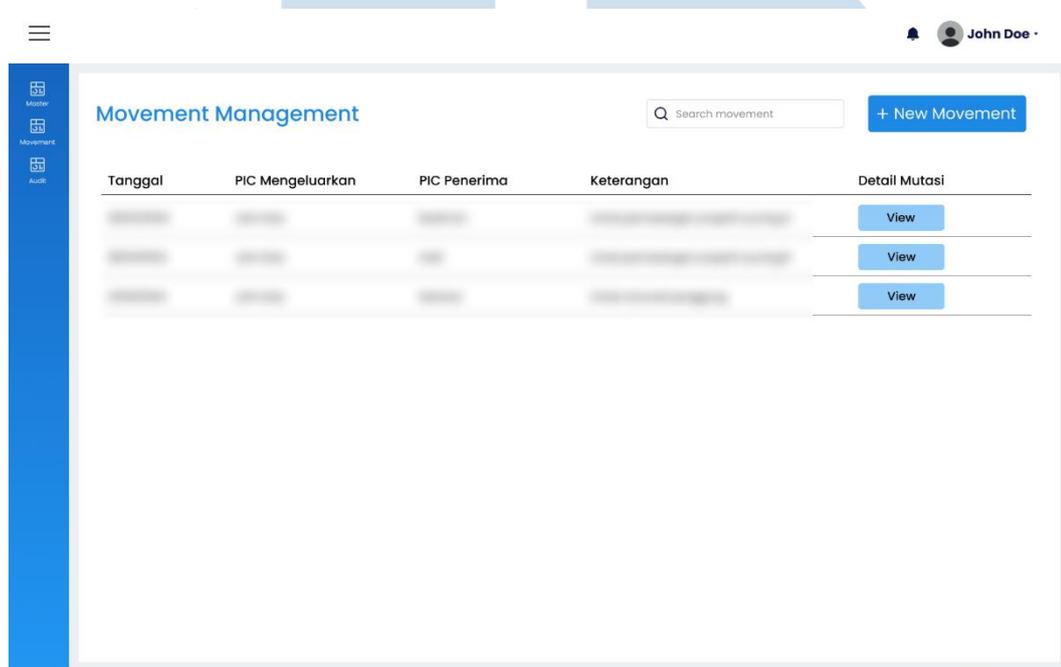
melakukan *save* untuk menyimpan aset tersebut. Selain menambahkan barang atau aset baru, pengguna juga dapat melakukan *edit* pada masing-masing aset yang sudah ditambahkan ke dalam sistem dengan menekan tombol *edit* berwarna kuning pada ujung sebelah kanan setiap aset, dan sistem akan menampilkan *pop up* untuk melakukan *edit* aset.



Gambar 3.14 Tampilan fitur *edit* aset

Gambar 3.14 merupakan tampilan dari fitur untuk mengedit aset yang sudah ada. Pengguna dapat mengganti ID Aset, Nama Aset, Kategori Aset, Lokasi Aset, Kondisi Aset, dan mengganti foto aset. Jika pada fitur menambahkan aset *field* kondisi aset akan terisi secara otomatis dengan status *ready to use*, maka pada menu ini kondisi aset tersebut dapat diubah oleh pengguna. Maka dari itu, *field* ini berubah tipe menjadi *field dropdown*. Pilihan status kondisi aset yang dapat diubah oleh pengguna adalah *damaged*, *under repair*, dan *ready to use*. Status *damaged* memiliki arti bahwa barang sedang dalam status rusak dan tidak dapat digunakan atau dipinjam. Status *under repair* memiliki arti bahwa barang sedang dalam tahap perbaikan. Dengan begitu, barang atau aset dengan status *damaged* atau *under repair* tidak bisa dibuatkan untuk aksi peminjaman pada menu *movement management*.

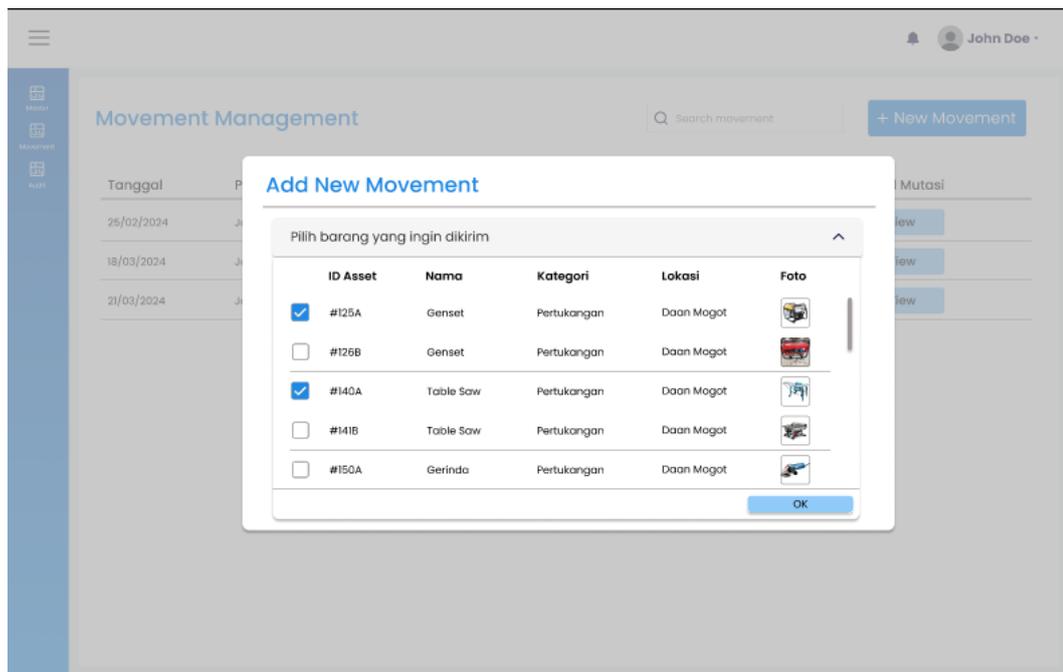
Apabila sudah selesai, maka pengguna dapat melakukan *save* untuk menyimpan aset tersebut. Setelah tampilan menu master aset dan alur proses untuk masing-masing fitur pada menu master aset selesai dibuat, dilanjutkan dengan tampilan menu *movement management*.



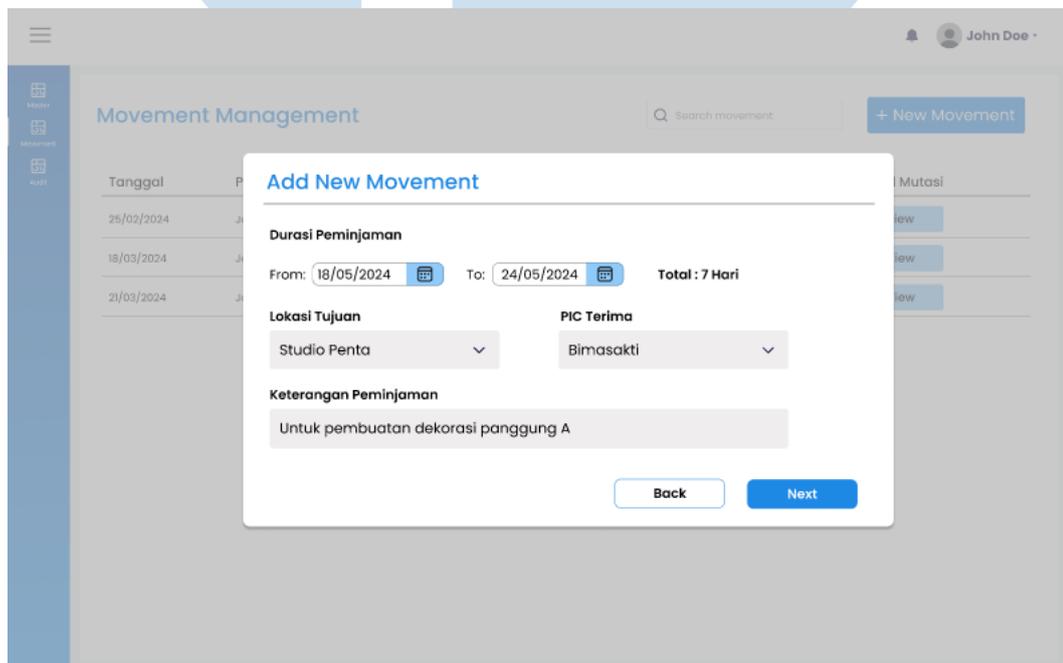
Gambar 3.15 Tampilan menu *movement management*

Gambar 3.15 merupakan tampilan utama dari menu *movement management* yang dibuat. Menu ini berguna sebagai menu untuk membuat peminjaman aset dan melacak pergerakan atau mutasi dari aset tersebut. Pada menu ini pengguna dapat membuat dan memantau detail dari setiap mutasi yang dibuat beserta informasi penting seperti tanggal dibuatnya mutasi, PIC yang bertanggung jawab, keterangan dari mutasi, dan detail mutasi. Untuk membuat peminjaman baru, pengguna dapat menekan tombol *New Movement*, dan akan diarahkan kepada tampilan pada gambar 3.16 dan 3.17.

UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.16 Tampilan pemilihan aset yang akan dipinjam



Gambar 3.17 Tampilan pengisian detail peminjaman aset

Gambar 3.16 dan 3.17 menunjukkan tampilan pemilihan aset yang akan dilakukan peminjaman. *Pop up* ini berfungsi untuk membuat peminjaman baru, dimana pengguna harus memilih barang atau aset yang akan dipinjam dari daftar barang yang ditampilkan dengan memilih pada tombol *checkbox* dan menekan

tombol OK. Daftar barang yang ditampilkan pada halaman ini merupakan daftar barang yang statusnya siap digunakan pada halaman menu master aset. Selanjutnya, pengguna harus mengisi detail peminjaman seperti durasi peminjaman, lokasi tujuannya, PIC yang akan menerima aset tersebut, serta keterangan peminjaman. *Field* durasi peminjaman bertipe *date picker*, Lokasi Tujuan dan PIC Penerima bertipe *dropdown*, dan terakhir *field* Keterangan Peminjaman bertipe *text field*. PIC Penerima bertipe *dropdown* dikarenakan PIC tersebut sudah dipetakan dari lokasi yang dipilih, sehingga jika pengguna memilih lokasi tertentu, maka daftar PIC yang ditampilkan adalah PIC yang bertugas di lokasi tersebut. Setelah pengguna menyelesaikan pengisian detail peminjaman aset, maka sistem yang dibuat akan menampilkan rangkuman dari seluruh detail peminjaman dan konfirmasi untuk pengiriman aset, yang ditunjukkan pada gambar 3.18 dan 3.19.

The screenshot shows a 'Movement Management' interface with a modal window titled 'Add New Movement Summary'. The modal contains the following information:

Add New Movement Summary

Detail Aset

No.	ID Asset	Nama	Kategori	Lokasi Aset	Foto	Qty
1	#125A	Genset	Pertukangan	Daan Mogot		x1
2	#140A	Table Saw	Pertukangan	Daan Mogot		x1

Detail Peminjaman

From : 18/05/2024
 To : 24/05/2024
 Durasi Peminjaman : 7 Hari

Detail Lokasi & PIC

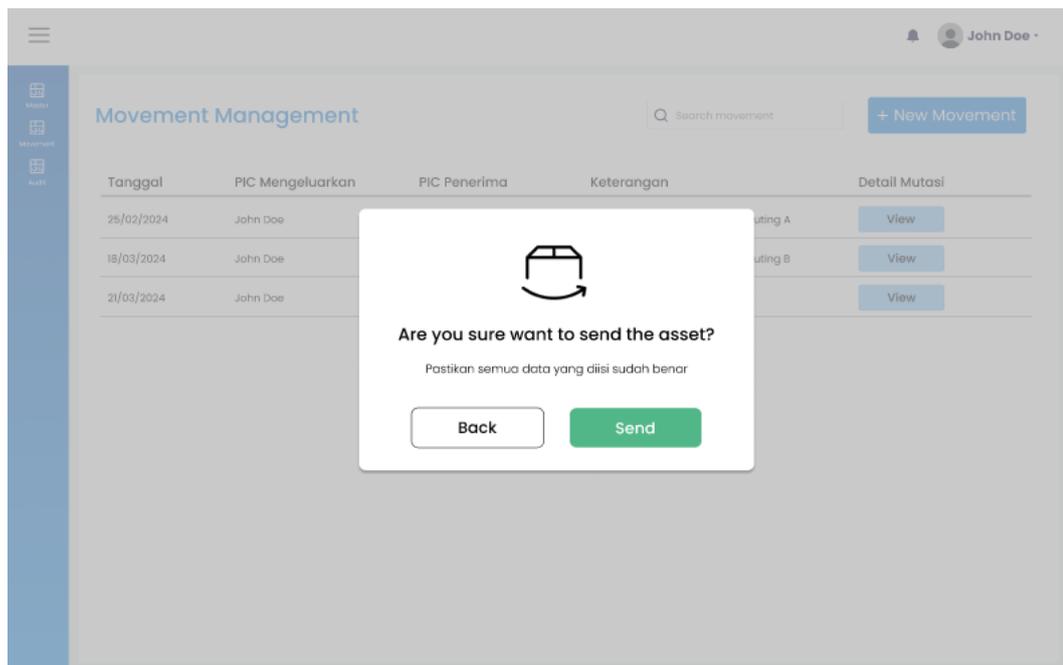
Lokasi Tujuan : **Studio Penta**
 PIC Terima : **Bimasakti**
 PIC Mengeluarkan : **John Doe**

Keterangan:

Buttons:

Gambar 3.18 Tampilan rangkuman detail peminjaman

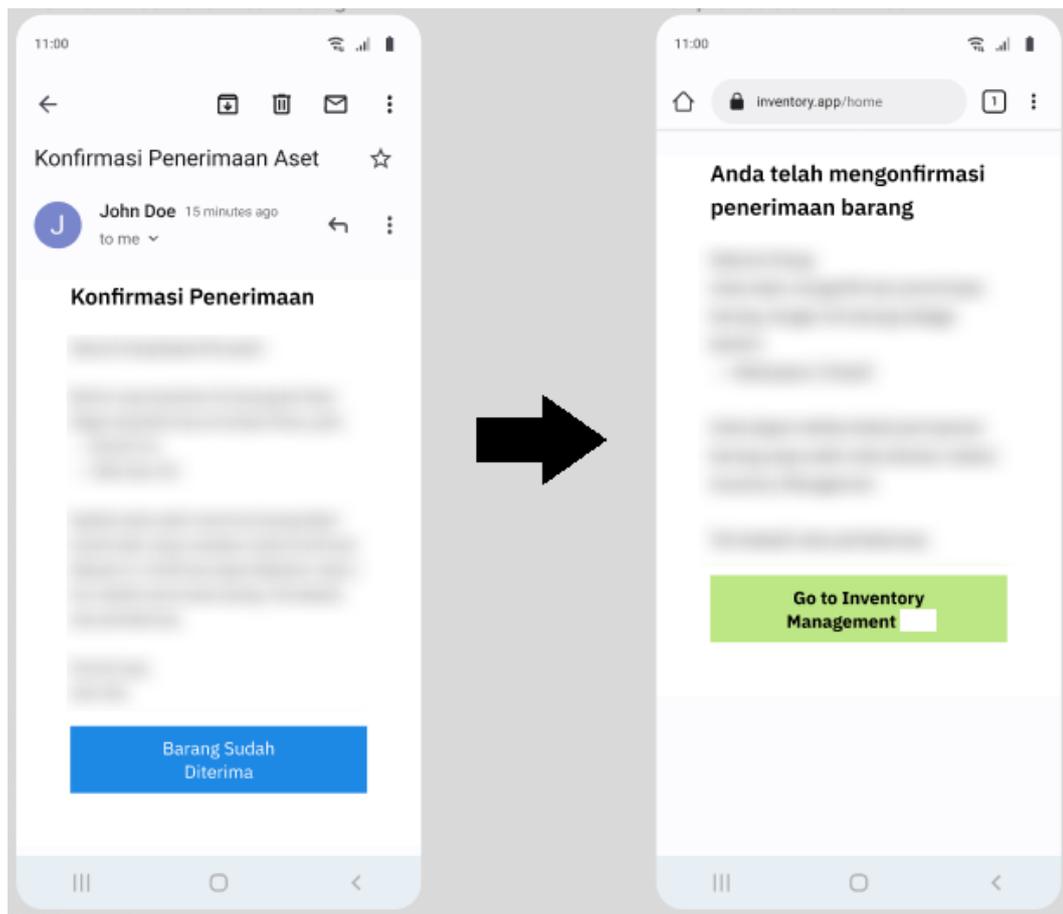
U N I V E R S I T A S
 M U L T I M E D I A
 N U S A N T A R A



Gambar 3.19 Pop up konfirmasi pengiriman barang kedalam sistem

Apabila proses pembuatn peminjaman barang sudah selesai, maka sistem dapat mengirimkan *email* kepada PIC penerima aset untuk melakukan konfirmasi, dimana proses ini juga merupakan implementasi dari kebutuhan *user* untuk membuat konfirmasi atas setiap perpindahan aset yang dilakukan. Maka dari itu, pembuatan tampilan *email* untuk proses konfirmasi penerimaan barang dibuat seperti pada gambar 3.20.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.20 Tampilan proses konfirmasi penerimaan aset

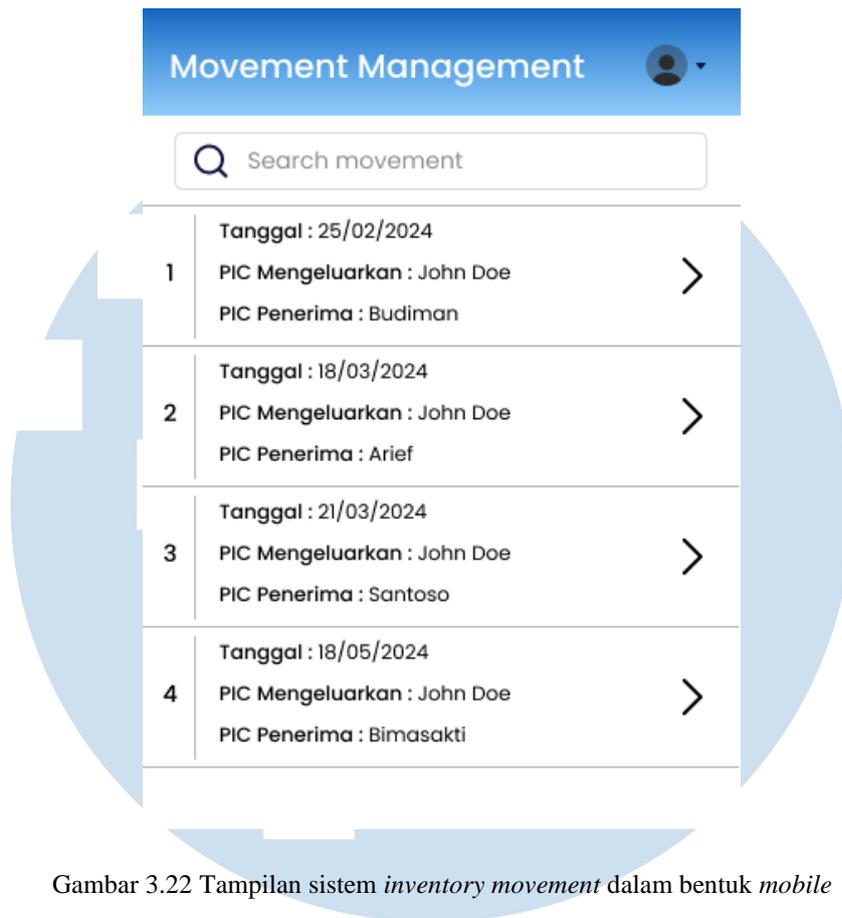
Tampilan menu terakhir yang dibuat adalah menu audit log. Menu ini merupakan menu yang penting karena menu ini dapat menampilkan catatan aktivitas penambahan, perubahan, atau penghapusan data yang dilakukan pada sistem. Tampilan menu yang dibuat digambarkan pada gambar 3.21.

U M W N
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

The screenshot shows a web interface for an Audit Log. On the left is a blue sidebar with navigation icons for 'Account', 'Movement', and 'Audit'. The main content area is titled 'Audit Log' and features a search bar labeled 'Search audit log'. Below the search bar is a table with the following columns: Activity, Name, Platform, Module, Data/Field, Data Before, Data After, Description, IP Address, User Agent, and Created At. The table contains three rows of data, which are blurred for privacy. The user's name 'John Doe' is visible in the top right corner of the interface.

Gambar 3.21 Tampilan *audit log* sistem

Gambar 3.21 menunjukkan tampilan dari *audit log*. Beberapa informasi penting yang perlu dicatat oleh sistem adalah jenis aktivitas yang dilakukan (*Insert/Update/Delete*), nama pelaku, *platform* dilakukannya aktivitas, *field* yang diubah, data sebelum dan setelah diubah, deskripsi aksi (*Success/Fail*), alamat IP, *User Agent*, dan tanggal serta waktu aktivitas dilakukan. Selain tampilan menu sistem yang menggunakan tampilan desktop, peserta magang *junior business analyst* juga membuat tampilan *mobile* untuk sistem ini, yang dikhususkan pada bagian *movement management*.



Gambar 3.22 Tampilan sistem *inventory movement* dalam bentuk *mobile*

Gambar 3.22 menunjukkan tampilan sistem *inventory movement* dalam bentuk *mobile*. Tampilan ini ditujukan untuk penggunaan sistem di tempat yang tidak ada komputer desktop, terutama bagu pengguna sistem yang bertugas sebagai PIC penerima, sehingga lebih mudah diakses melalui *browser* pada *smartphone*. Fitur yang terdapat pada tampilan *mobile* ini adalah pengembalian aset sebelum tenggat waktu dan sesuai tenggat waktu. Berikut adalah tampilan yang dibuat:

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

< Pengembalian Aset

Detail Pengembalian Aset

No.	ID	Nama	Lokasi
1	#125A	Genset	Studio Penta

Alasan pengembalian aset sebelum tenggat waktu:

- Aset Rusak
- Kesalahan Request
- Perubahan Kebutuhan

Tuliskan alasan lebih rinci disini

Tambahkan Foto Aset

Drop image here
or
[Browse File](#)
JPG, JPEG, or PNG, max 2 MB

Return

Gambar 3.23 Tampilan pengembalian aset sebelum tenggat waktu

< Pengembalian Aset

Detail Aset

No.	ID	Nama	Lokasi
1	#125A	Genset	Studio Penta
2	#140A	Table Saw	Studio Penta

Tambahkan Foto Aset

Drop image here
or
[Browse File](#)
JPG, JPEG, or PNG, max 2 MB

Return

Gambar 3.24 Tampilan pengembalian aset sesuai tenggat waktu

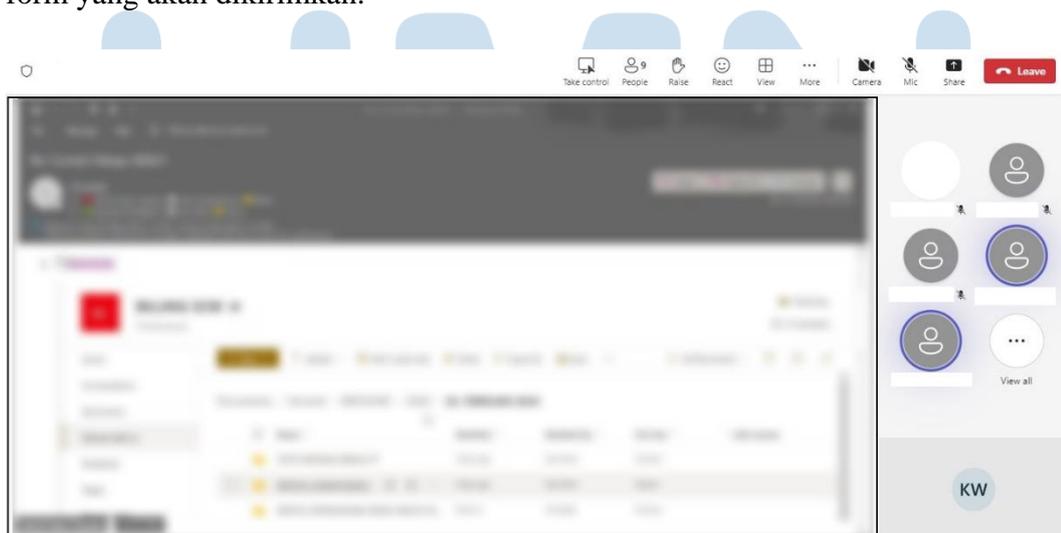
Keseluruhan tampilan sistem ini dibuat berdasarkan kebutuhan *user* dimana tampilan sistem ini sudah memberikan gambaran cakupan kemampuan sistem untuk dapat mengelola berbagai pencatatan dan perpindahan aset dengan detail seperti nama, kategori, lokasi, jumlah, foto, status, dan kode aset yang merujuk kepada kode SAP. Tampilan sistem yang dibuat juga dapat mengelola berbagai pencatatan dan perpindahan aset dengan detail. Tahap selanjutnya setelah pembuatan tampilan sistem adalah presentasi dan pembuatan dokumentasi dari sistem yang akan dibuat. Akan tetapi, dikarenakan terdapat proyek baru yang lebih penting untuk diselesaikan terlebih dahulu, maka peserta magang *junior business analyst* dialihkan untuk membantu pembuatan proyek yang lain.

3.2.4 Mengerjakan Proyek Blast Email (Minggu 6 - 10)

Proyek ketiga yang dikerjakan adalah proyek pembuatan sistem *blast email*. Proyek sistem *blast email* merupakan proyek baru yang dibuat berdasarkan kebutuhan *user* yang berasal dari permasalahan bisnis pada divisi marketing dalam mengirimkan *email* yang berisikan tautan form evaluasi sales kepada banyak klien.

3.2.4.1 Meeting User Requirement

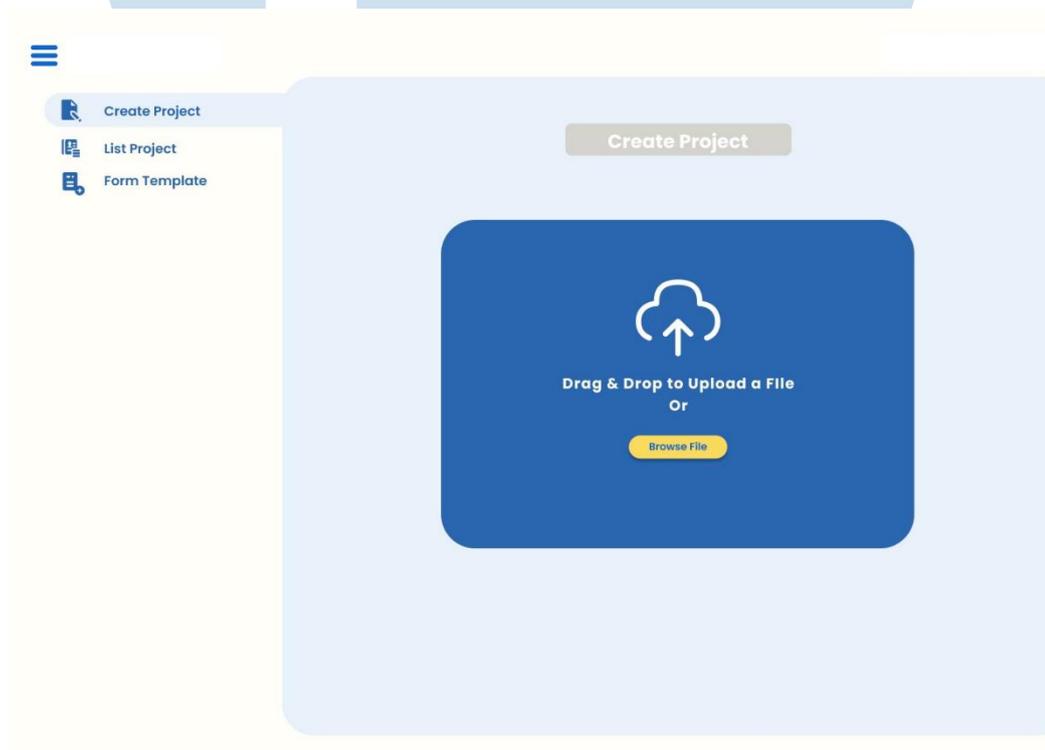
Sama seperti proyek yang dilakukan sebelumnya sebelumnya, untuk memulai proyek ini diadakan meeting antara mentor, peserta magang *junior business analyst*, dan *user* dari divisi marketing. Tahap meeting *user requirement* ini dilakukan melalui meeting secara daring bersama dengan *user*. Dari diskusi dan pembahasan yang dilakukan melalui meeting ini, diketahui permasalahan bisnis yang sedang dialami oleh *user* yaitu belum adanya sistem yang dapat mengambil data dari *file excel* yang dimiliki dan mengirimkan *email* secara bulk atau dalam jumlah besar kepada banyak klien, dimana *email* yang dikirim dapat berisikan lampiran *file* dan *link* form untuk mengisi survei kepuasan pelanggan terhadap kinerja sales. Dari sini, kebutuhan pengguna adalah dibutuhkan sebuah sistem yang dapat melakukan *blast email*, selain itu sistem ini juga harus dapat mengambil data dari *file* dengan format *.xlsx* dan juga dapat membuat *template* pertanyaan untuk form yang akan dikirimkan.



Gambar 3.25 Meeting user requirements

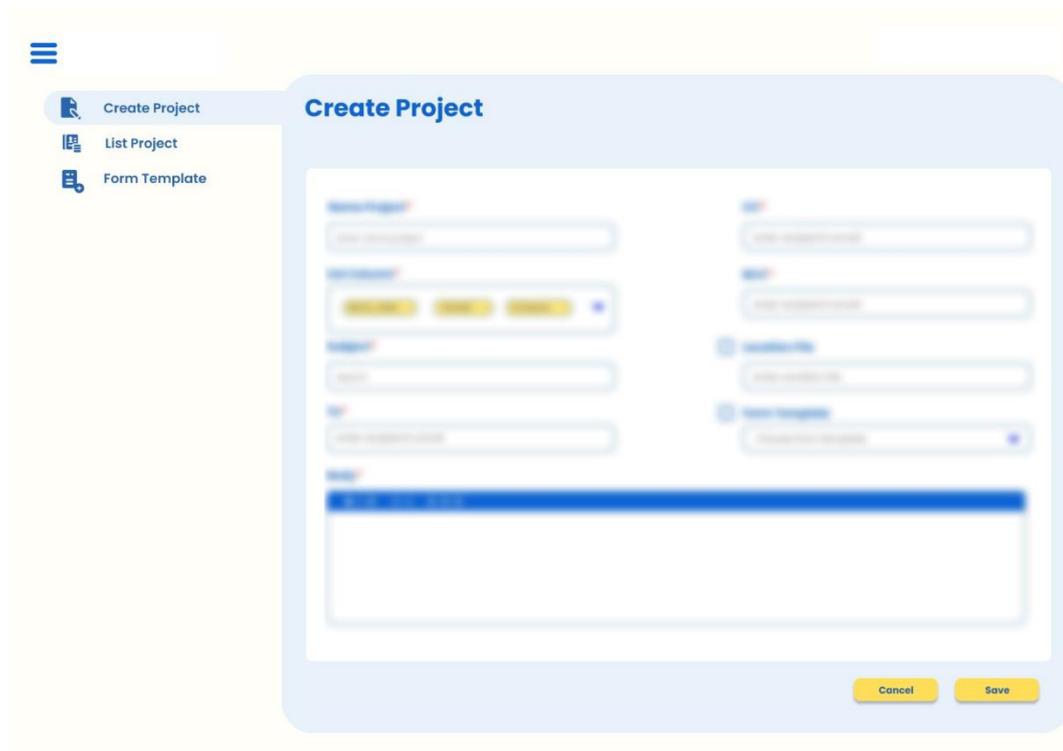
3.2.4.2 Pembuatan Tampilan Sistem Blast Email

Berdasarkan kebutuhan pengguna yang telah didefinisikan sebelumnya, *junior business analyst* bertanggung jawab untuk merancang dan mendokumentasikan antarmuka awal atau *mockup* sistem *blast email*. Rancangan awal sistem ini terdapat pada tiga menu utama, yaitu menu *Create Project*, *List Project*, dan *Form Template*. Menu *Create Project* berfungsi sebagai menu untuk membuat proyek pengiriman *email*, dimana satu *file excel* yang diunggah hanya dapat digunakan untuk satu proyek. Menu *List Project* berfungsi sebagai menu yang menyimpan daftar dari proyek-proyek yang sudah dibuat dan aksi terhadap masing-masing proyek tersebut. Sedangkan menu *Form Template* berfungsi sebagai tempat untuk membuat daftar pertanyaan dan menyimpan daftar pertanyaan tersebut.



Gambar 3.26 Tampilan fitur *upload* pada menu *create project*

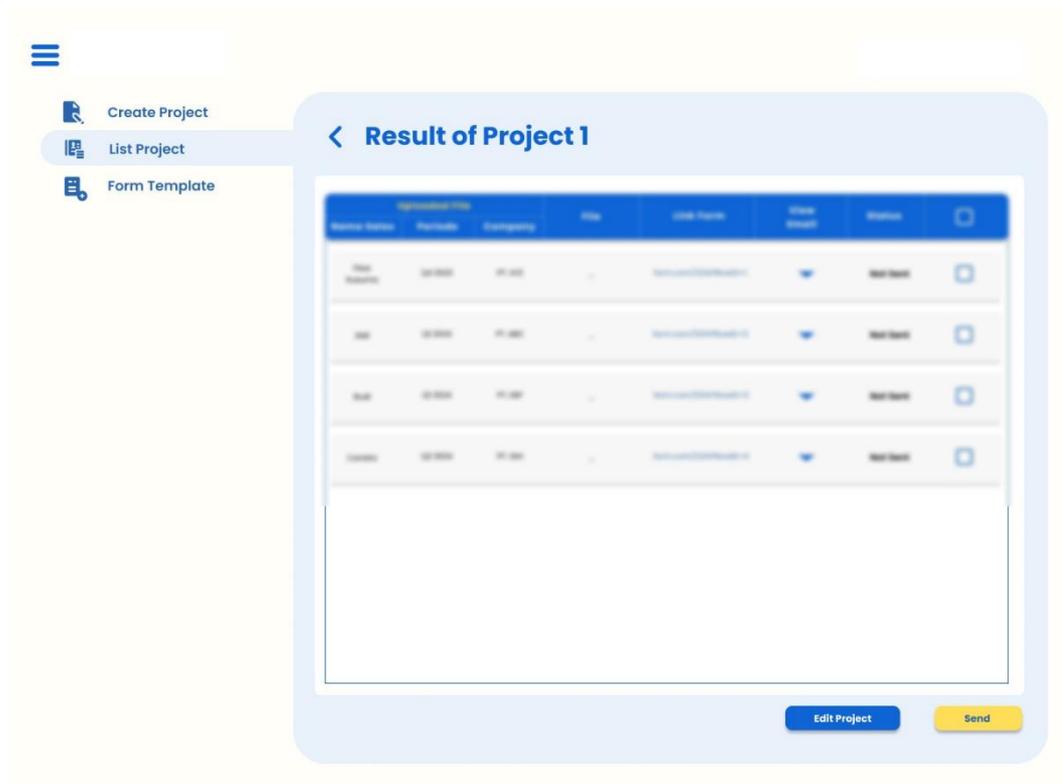
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.27 Tampilan menu *create project*

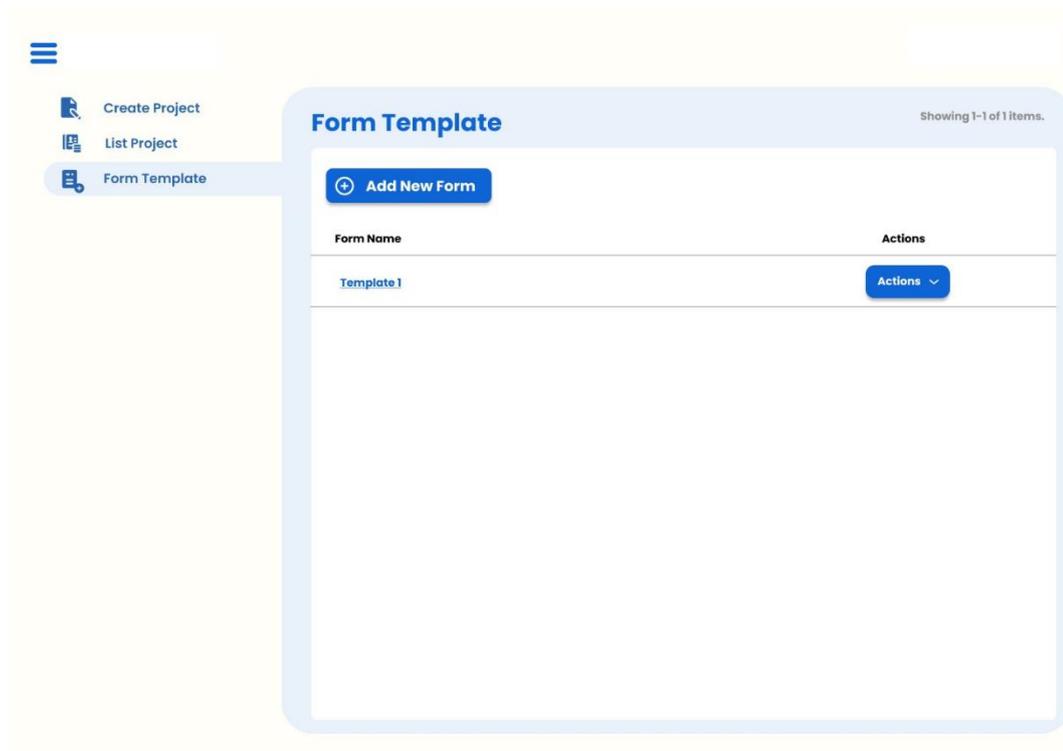
Gambar 3.26 menunjukkan tampilan menu untuk mengunggah *file* dengan format *.xlsx* yang akan digunakan dalam pembuatan proyek pengiriman *email*. Selanjutnya, gambar 3.27 menunjukkan tampilan menu untuk pembuatan proyek pengiriman *email*. Pada menu ini, *file* dengan format *.xlsx* yang sudah diunggah sebelumnya akan dibaca oleh sistem dan disimpan dalam bentuk variabel, dimana variabel ini dapat digunakan dalam penulisan alamat *email* tujuan, subject, dan body *email*. Pada menu ini juga beberapa informasi penting harus diisi seperti nama proyek dan pemilihan form pertanyaan yang akan digunakan di dalam *email*.

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



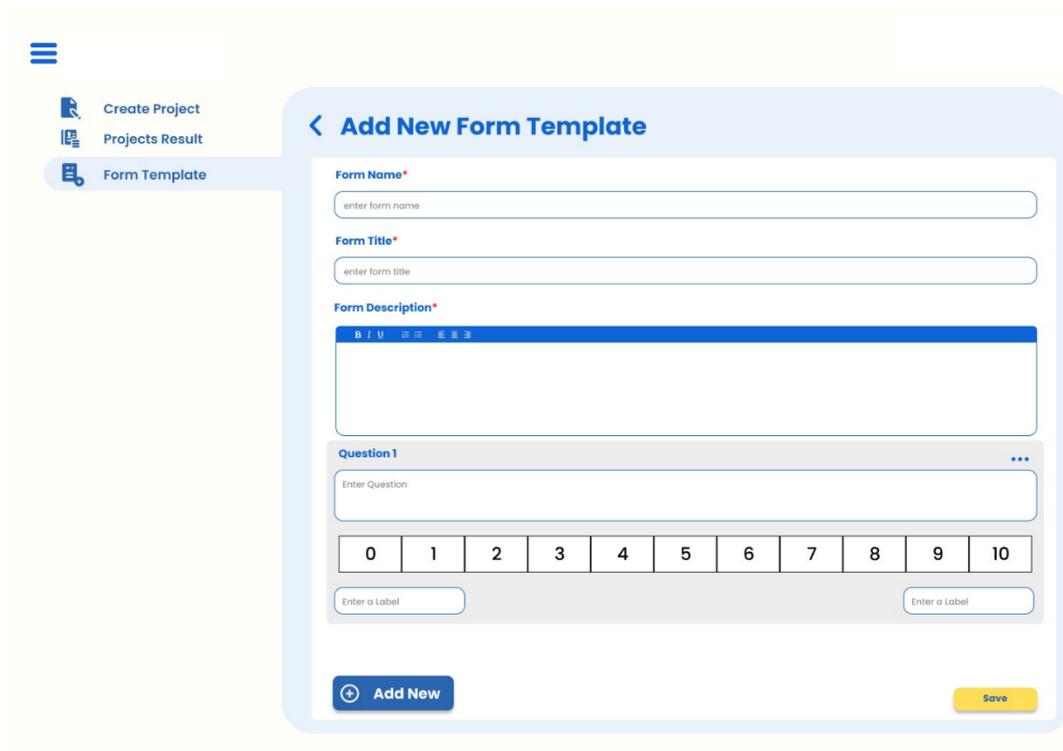
Gambar 3.28 Tampilan menu *list project*

Gambar 3.28 menunjukkan tampilan menu *list project* yang berisikan daftar dari *project* yang sudah dibuat. Pada menu ini, daftar *project* yang sudah dibuat dapat dilakukan pengecekan ulang sebelum dilakukan pengiriman *email*. Setiap *email* yang akan di *blast* memiliki status pengirimannya masing-masing. Status pengiriman *email* yang dibuat dapat bertuliskan *Not Sent*, *Sent*, dan *Sent (n-x)*. Status *Not Sent* memiliki arti bahwa *email* tersebut belum pernah dikirimkan sama sekali. Status *Sent* memiliki arti bahwa *email* sudah berhasil dikirimkan dan juga berarti bahwa *email* tersebut baru dikirimkan satu kali. Dan status *Sent (n-x)* memiliki arti bahwa *email* tersebut sudah dikirimkan ulang sebanyak *n* kali kepada penerima. Masing-masing *email* dapat dikirimkan secara terpisah maupun secara bersamaan dengan menggunakan fitur *checkbox* yang sudah dibuat.



Gambar 3.29 Tampilan menu *form template*

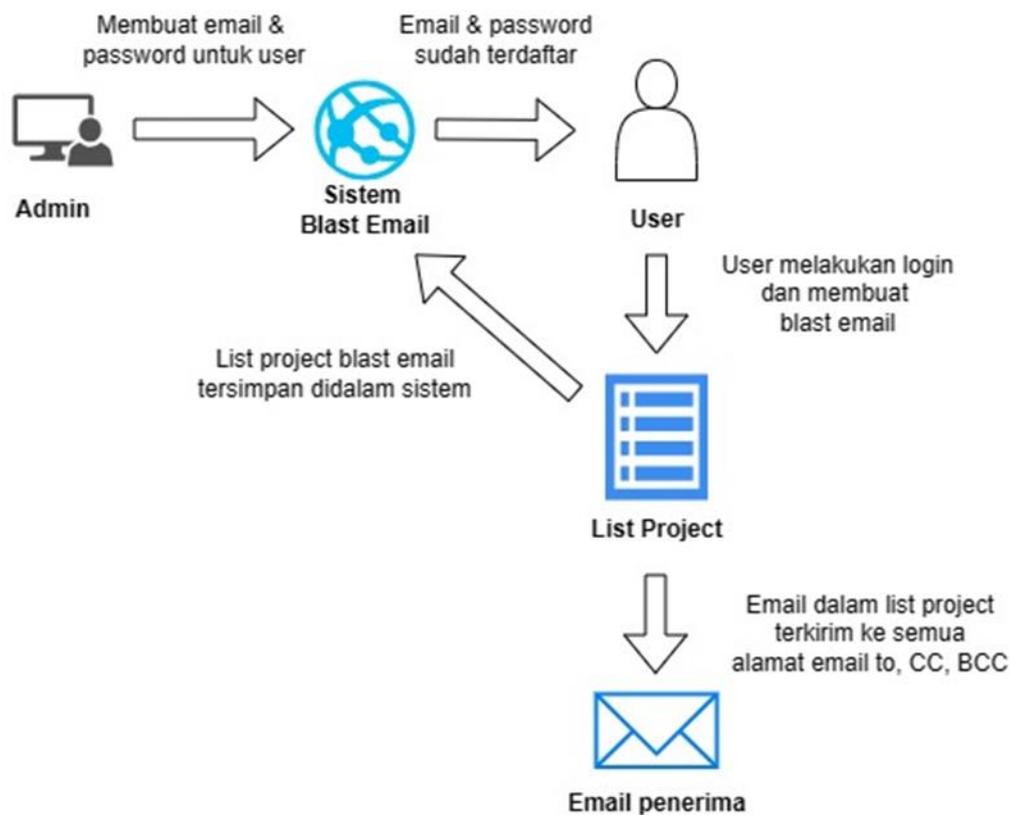
Gambar 3.29 menunjukkan tampilan menu *form template* yang berisikan daftar *template* pertanyaan yang sudah dibuat. Pada menu ini, masing-masing *template* memiliki aksi yang dapat dilakukan, seperti *edit*, *delete*, *preview form*, dan *view submission*. Aksi *edit* berguna untuk mengganti pertanyaan yang sudah ada di dalam *template*. Aksi *delete* berguna untuk menghapus *template* yang sudah dibuat termasuk semua pertanyaan yang ada di dalamnya. Aksi *preview form* berguna untuk melihat daftar pertanyaan yang sudah dibuat pada *template* tersebut. Dan aksi *view submission* berguna untuk melihat daftar jawaban yang sudah dikumpulkan oleh penerima *email* bagi penerima yang mengisi pertanyaan untuk *template* tersebut. Selain itu, pada menu ini, pengguna juga dapat menambahkan *template* pertanyaan baru melalui tombol *Add New Form*. Tampilan dari halaman pembuatan pertanyaan baru ditunjukkan pada gambar 3.30.



Gambar 3.30 Tampilan halaman pembuatan form

3.2.4.3 Pembuatan Dokumentasi Sistem

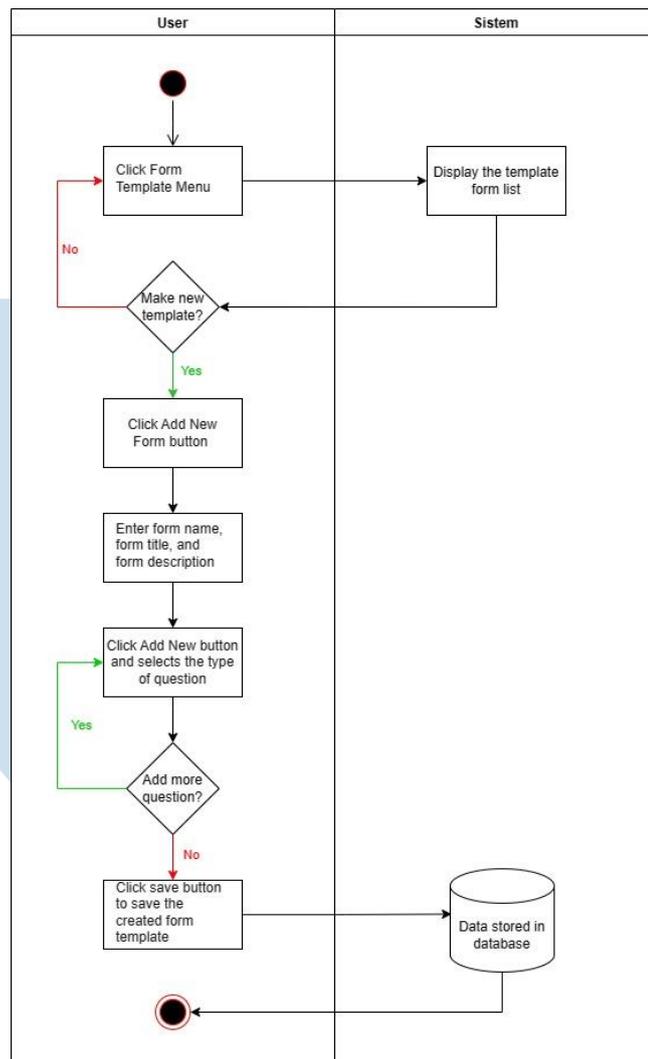
Setelah membuat tampilan atau *mockup* sistem, maka tugas selanjutnya adalah mendokumentasikan sistem *blast email* tersebut. Dokumentasi yang dibuat adalah *Functional Specification Document* (FSD). Dokumentasi dengan jenis FSD digunakan karena sistem yang dibuat merupakan sistem baru yang belum ada sebelumnya. Pembuatan FSD mencakup informasi sistem saat ini dan informasi sistem yang diinginkan, serta rencana implementasi sistem. Peserta magang *junior business analyst* memiliki fokus pengerjaan pada dokumentasi informasi sistem saat ini dan informasi sistem yang diinginkan. Dokumentasi informasi sistem mencakup arsitektur desain atau topologi sistem, *flow* sistem *website*, *map view website*, dan design *website*. Informasi sistem saat ini adalah masih belum adanya sistem yang dapat mengirimkan *blast email* secara otomatis dan masih menggunakan cara manual. Maka dari itu, pembuatan sistem *blast email* dilakukan, dengan arsitektur desain yang diusulkan oleh *junior business analyst* dan digambarkan pada gambar 3.31.



Gambar 3.31 Topologi sistem *blast email*

Berdasarkan topologi pada gambar 3.31, proses dimulai dengan admin yang membuat *email* dan *password* untuk *user* pada *website* yang kemudian akan disimpan pada *database*. Langkah berikutnya adalah jika *email* dan *password* sudah terdaftar maka *user* dapat masuk kedalam aplikasi untuk membuat daftar dari *email* yang akan dilakukan *blast*, kemudian data tersebut akan otomatis tercatat pada *website* yang kemudian akan disimpan pada *database*. *Blast email* yang sudah dibuat akan tersimpan di dalam menu *list project*, dan terakhir *email* dari *list project* akan dikirim ke semua alamat penerima yang ditentukan, termasuk 'To', 'CC', dan 'BCC'.

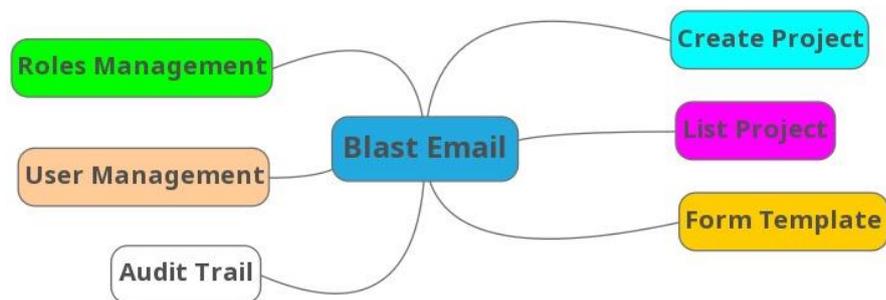
Berikutnya, *flow* sistem *website* dibuat untuk menunjukkan alur penggunaan sistem kepada *user* dan juga tim pengembang. Salah satu contoh *flow* sistem yang dibuat adalah alur pada menu *form template*, yang digambarkan pada gambar 3.32.



Gambar 3.32 Alur sistem menu *form template*

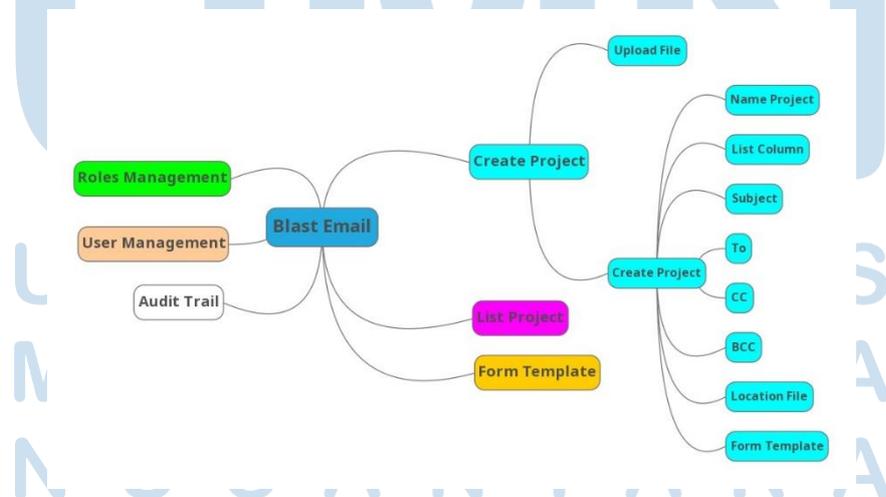
Gambar 3.32 menunjukkan alur sistem pada menu *form template*. Alur sistem ini menjelaskan bagaimana alur pada menu *form template*. Dalam proses pembuatan *template* form, pertama-tama, *user* memilih menu *form template* dan sistem akan menampilkan daftar *template* yang tersedia. Jika *user* ingin membuat *template* baru, mereka dapat menekan tombol *Add New Form*. Setelah itu, *user* akan diarahkan ke halaman pembuatan *template* form dimana mereka dapat mengisi nama *form*, judul *form*, dan deskripsi *form*. Untuk menambahkan pertanyaan baru ke dalam form, *user* dapat menekan tombol *Add New* dan memilih jenis pertanyaan yang diinginkan. *User* kemudian dapat mengisi pertanyaan tersebut sesuai kebutuhan. Proses ini dapat diulangi jika *user* ingin menambahkan lebih banyak pertanyaan. Setelah selesai menambahkan pertanyaan, *user* dapat menekan tombol

Save untuk menyimpan *template* form yang telah dibuat. Sistem kemudian akan menyimpan *template* form baru ini ke dalam *database* dan mengarahkan *user* kembali ke halaman menu Form Template. Ini adalah rangkuman dari proses pembuatan *template* form. Setelah alur sistem dijelaskan, maka *map view* sistem akan dibuat untuk menunjukkan posisi *field* di dalam masing-masing menu dengan lebih detail.



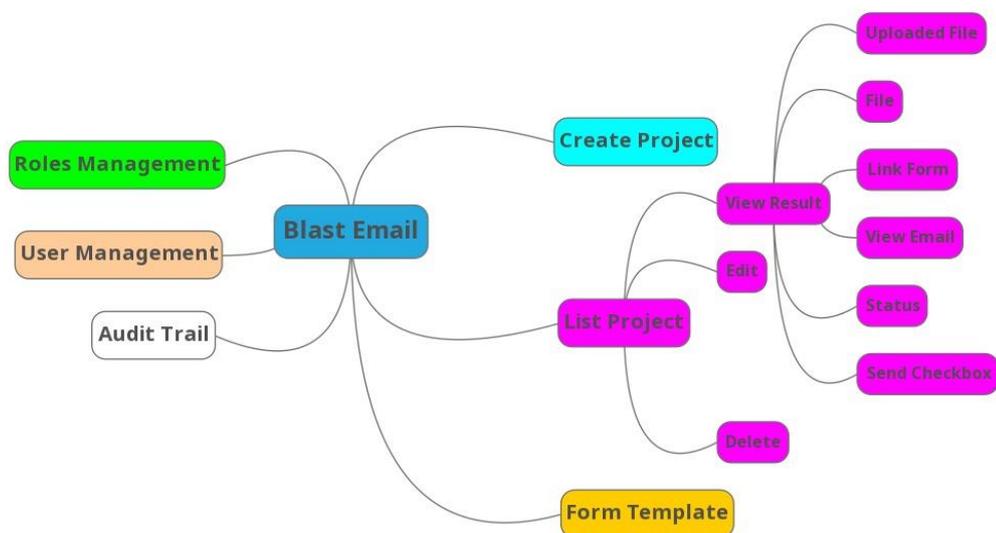
Gambar 3.33 General map view

Gambar 3.33 menunjukkan *general map view* dari sistem *blast email*. Dari *map view* yang sudah dibuat, dapat terlihat bahwa sistem *blast email* secara umum memiliki enam fungsi atau modul utama yang dibuat sebagai menu di dalam sistem, yaitu *create project*, *list project*, *form template*, *roles management*, *user management*, dan *audit trail*. Tiga menu utama yang akan dijabarkan adalah menu *create project*, *list project*, dan *form template*.



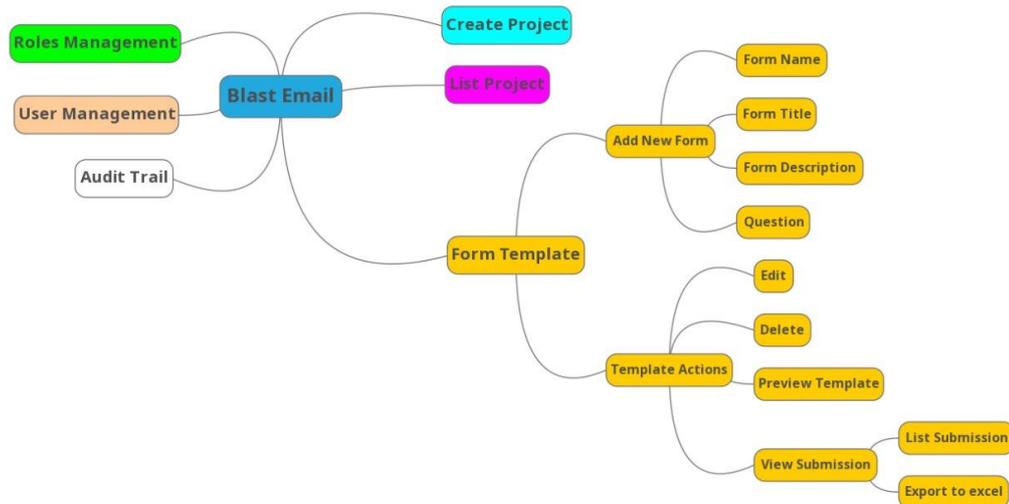
Gambar 3.34 Create project map view

Gambar 3.34 menunjukkan general map view dari menu *create project*. Menu *create project* memiliki dua fungsi utama, yaitu *upload file* dan *create project*. Pada fungsi atau fitur *create project*, *field* yang ada di dalamnya adalah nama proyek, daftar dari kolom yang dibaca pada *file* yang diunggah pengguna, *subject email* yang akan dituliskan, alamat penerima yaitu *To*, *CC*, dan *BCC*, lokasi *file* untuk menambahkan *attachment file*, dan *form template* yang digunakan untuk memilih pertanyaan mana yang akan digunakan pada *email* yang akan di *blast*.



Gambar 3.35 List project map view

Gambar 3.35 menunjukkan general map view dari menu *list project*. Menu *list project* memiliki tiga fungsi utama, yaitu *view result*, *edit*, dan *delete*. Pada fungsi atau fitur *view result*, *field* yang ada di dalamnya adalah *uploaded file* yang berisikan kolom yang dibaca sistem dan sudah digunakan dari *file* yang sudah diunggah, kemudian *file* yang berisikan *file* yang akan dijadikan *attachment* pada *email*, *link form* yang berisikan tautan untuk pengisian form berdasarkan *form template* yang sudah dipilih sebelumnya, *view email* untuk melihat bentuk dari *email* yang akan dikirimkan kepada penerima, *status* yang berisikan status pengiriman *email*, dan *send checkbox* yang berguna untuk mengirimkan *email* baik secara parsial maupun keseluruhan. Fungsi *edit* berguna untuk meng-*edit* proyek yang sudah dibuat pada daftar proyek, dan fungsi *delete* berguna untuk menghapus proyek yang sudah dibuat dari daftar proyek.



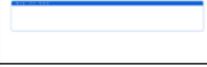
Gambar 3.36 *Form template map view*

Gambar 3.36 menunjukkan general *map view* dari menu *form template*. Menu *form template* memiliki dua fungsi utama, yaitu *add new form* dan *template actions*. Pada fitur *add new form*, *field* yang ada di dalamnya adalah nama form, judul form, deskripsi form, dan daftar pertanyaan yang dibuat. Pada fitur *template action*, *field* yang ada di dalamnya adalah *edit* yang berguna untuk meng-*edit* pertanyaan yang sudah dibuat pada *template*, *delete* yang berguna untuk menghapus *template* pertanyaan yang sudah dibuat dari daftar *template*, *preview template* yang berguna untuk melihat pertanyaan, dan *view submission*. *Field view submission* sendiri berisikan fitur untuk melihat *list submission* yang berguna untuk melihat daftar jawaban dan *export to excel* yang berguna untuk mengubah tabel *submission* menjadi *format file .xlsx*.

Setelah pembuatan *map view* dilakukan, maka komponen FSD terakhir yang akan dibuat adalah penjelasan design *website*. Contoh penjelasan dari design *website* untuk dokumentasi digambarkan pada gambar 3.37.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Tabel 2. Informasi Kondisi pada form Add New Template

Item	Jenis	Image/Kolom	Keterangan	Kondisi
Input Form Name	TextField		Field untuk menginputkan nama Form	<ul style="list-style-type: none"> • Mandatory • Textfield Nama Form • Type Data (Varchar 256)
Input Form Title	TextField		Field untuk menginputkan judul Form	<ul style="list-style-type: none"> • Mandatory • Textfield Title Form • Type Data (Varchar 256)
Input Form Description	TextField		Field untuk menginputkan deskripsi Form	<ul style="list-style-type: none"> • Mandatory • Textfield Form Description • Type Data (Varchar 256)
Button Add New	List		Daftar pilihan jenis jawaban untuk pertanyaan yang dapat dimasukkan di form	<ul style="list-style-type: none"> • Mandatory • Button Add New Dapat diklik lalu akan menampilkan List dengan GUI sebagai berikut: <div style="text-align: center;">  <p>Gambar 60. GUI Button Add New</p> </div> <ul style="list-style-type: none"> • Jika salah satu pilihan diklik muncul tampilan dengan GUI sebagai berikut:

Gambar 3.37 Contoh dokumentasi *design website*

Pada gambar 3.37 contoh dokumentasi dari desain *website* secara detail yang dibuat dalam bentuk tabel pada dokumen FSD. Tabel di dalam dokumentasi memiliki beberapa header seperti *header nama item* yang menjelaskan nama komponen yang ada di dalam menu yang bersangkutan, *header jenis* yang menunjukkan jenis komponen tersebut, *header image/kolom* yang menunjukkan *Graphical User Interface* dari komponen tersebut, *header keterangan* yang menunjukkan penjelasan kegunaan dari komponen tersebut, dan *header kondisi* yang menunjukkan validasi yang diperlukan dalam penggunaan komponen tersebut.

Perancangan dari keseluruhan tampilan sistem ini sudah dibuat berdasarkan kebutuhan pengguna, dengan antarmuka yang mencerminkan fungsionalitas dari sistem. Sistem ini memiliki kapabilitas untuk mengolah data dari *file* Excel dan melakukan pengiriman *email* massal kepada sejumlah besar klien. Email yang dikirimkan dapat mencakup lampiran *file* dan tautan ke form survei, yang memungkinkan pengumpulan umpan atau feedback balik pelanggan tentang kinerja tim penjualan.

Selain itu, sistem ini juga memiliki kemampuan untuk mengambil data dari *file* dengan format *.xlsx* dan merancang *template* pertanyaan untuk form survei yang akan dikirimkan. Dengan begitu, sistem ini dapat memberikan solusi yang efisien dan efektif untuk mengelola pengiriman *email* secara bulk atau massal dan untuk mengumpulkan feedback dari klien.

3.2.5 Mengerjakan Proyek IT Management System Dashboard (Minggu 11 - 14)

Proyek keempat yang dikerjakan adalah proyek pembuatan *dashboard* untuk sistem manajemen IT. Proyek sistem *dashboard* ini merupakan proyek pembuatan sistem baru yang diinisiasi oleh mentor dan berdasarkan kebutuhan *user* yang berasal dari permasalahan bisnis pada divisi internal audit PT XYZ. Dalam konteks proyek ini, divisi audit internal telah merancang konsep dasar pembuatan *dashboard* yang diharapkan dan menyampaikan konsep tersebut kepada tim business analyst. Sehingga, tugas utama yang dilakukan dalam proyek ini adalah menginterpretasikan dan mentransformasikan konsep dasar yang telah disediakan menjadi sebuah *dashboard* yang lengkap, mendalam, sederhana, dan visualnya menarik, sehingga mudah untuk dipahami dan dianalisis.

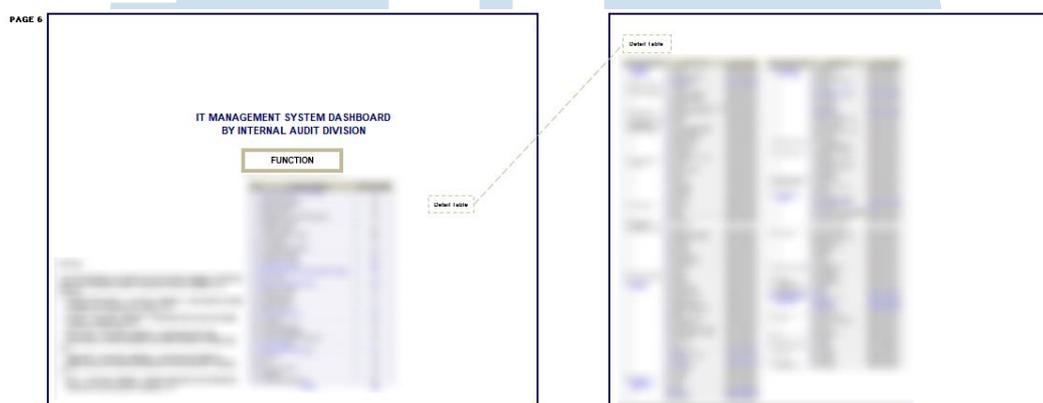
3.2.5.1 User Requirement Gathering

Proyek ini bermula dari penugasan yang diberikan oleh mentor, yang bertindak sebagai kepala proyek. Pada proyek ini, tim junior business analyst tidak bertemu dan berinteraksi langsung dengan divisi audit internal, tetapi divisi tersebut telah memberikan gambaran kebutuhan mereka kepada tim. Oleh karena itu, diskusi dilakukan dengan mentor untuk memahami kebutuhan pengguna dalam proyek ini. Selain itu, diskusi juga dilakukan bersama rekan dari tim junior business analyst. Dari hasil diskusi, didapatkan kebutuhan dimana *user* ingin memiliki sistem *dashboard* yang dapat mengupload *file database* mentah yang dimiliki dan menampilkan data yang sudah diolah. Selain itu, *user* juga ingin sistem yang dibuat dapat memudahkan interpretasi dan analisis data, karena selain akan digunakan oleh divisi internal audit, sistem ini juga akan digunakan oleh pimpinan perusahaan. Berdasarkan kebutuhan yang dibahas sebelumnya, maka sistem ini memerlukan fitur untuk mengunggah *file database* dalam format tertentu, kemudian menyimpan

data tersebut dalam sistem untuk diolah dan ditampilkan dalam bentuk *dashboard*. Sistem ini juga harus dilengkapi dengan grafik yang menarik dan sesuai dengan data yang ditampilkan untuk memudahkan analisis data.

3.2.5.2 Pembuatan Dashboard

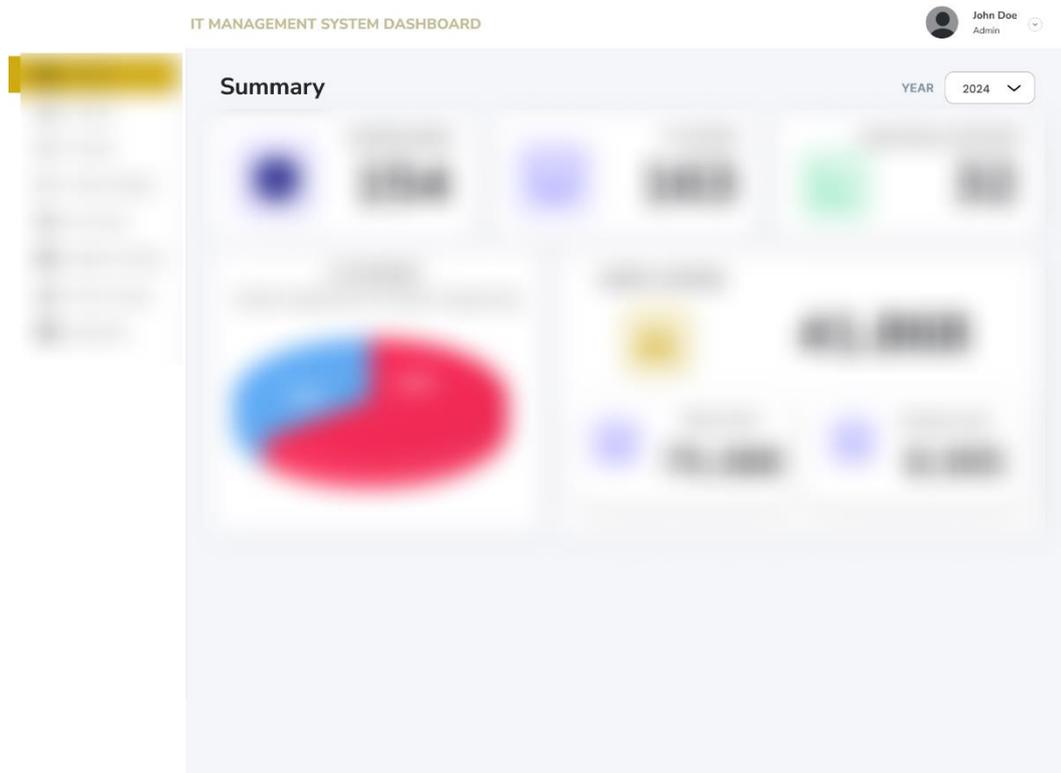
Pembuatan *dashboard* dimulai dengan pemahaman akan gambaran sistem yang diinginkan oleh *user*. Divisi internal audit memberikan beberapa gambaran dengan detail halaman yang akan dibuat menjadi menu di dalam *dashboard* terkait.



Gambar 3.38 Contoh gambaran sistem yang diberikan oleh divisi audit internal

Berdasarkan gambar 3.38, *dashboard* yang diinginkan oleh *user* adalah *dashboard* dengan antarmuka yang dapat menampilkan ringkasan dari data yang diolah pada halaman utamanya. Setelah itu, apabila *user* ingin mengeksplorasi detail dari masing-masing ringkasan, maka *user* dapat melakukannya dengan memilih ringkasan yang relevan. Untuk memudahkan interpretasi dan analisis data, maka halaman utama akan diisi dengan grafik yang sesuai dengan ringkasan data yang ada.

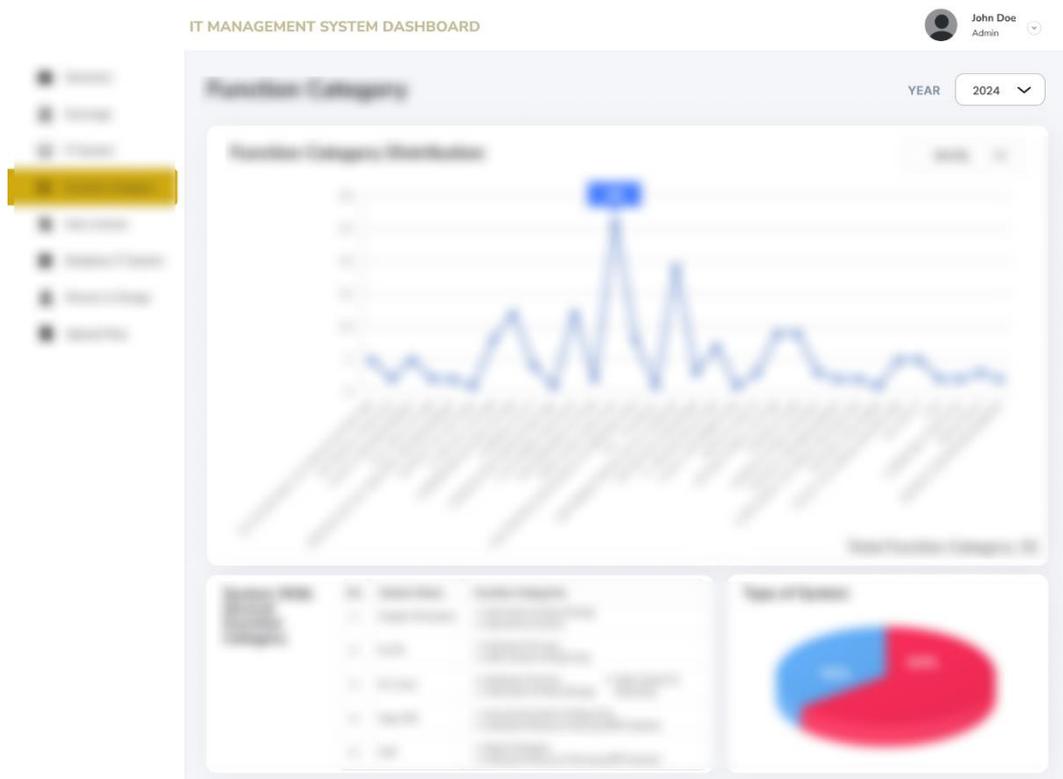
user ke dalam sistem sebelumnya. Daftar *file database* yang sudah diunggah dapat menampilkan beberapa informasi seperti aktor yang mengunggah *file*, detail waktunya, dan dapat melakukan preview untuk melihat kembali isi dari data yang diunggah untuk memastikan data tersebut sudah benar.



Gambar 3.41 Tampilan halaman utama *dashboard*

Gambar 3.41 menunjukkan tampilan halaman utama dari sistem *dashboard* yang dibuat. Halaman utama atau halaman home merupakan rangkuman dari beberapa halaman *dashboard* yang sudah dibuat. Pada halaman ini, beberapa jenis visualisasi data digunakan, seperti card dan pie chart.

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 3.42 Contoh tampilan halaman kategori sistem IT

Selain halaman utama, *dashboard* ini juga memiliki beberapa halaman lainnya. Gambar 3.42 menunjukkan tampilan halaman kategori sistem IT yang digunakan oleh perusahaan. Halaman ini menggunakan visualisasi data berupa grafik garis atau line chart untuk menampilkan distribusi dari kategori sistem yang ada. Visualisasi ini dapat diurutkan secara ascending maupun descending. Selain itu, terdapat tabel di sisi kiri bawah yang menunjukkan beberapa sistem IT yang tergolong ke dalam dua kategori atau lebih. Visualisasi terakhir yang digunakan adalah pie chart untuk menunjukkan perbandingan antara tipe dari sistem IT yang ada. Hasil proyek *IT management system dashboard* yang dibuat adalah sebuah sistem *dashboard* yang dapat mengolah dan menampilkan data secara efisien dengan memanfaatkan *file database* yang sudah diunggah sebelumnya. Dengan demikian hasil *dashboard* ini dapat digunakan oleh *user* untuk menganalisis data dan setelah dilanjutkan diharapkan *dashboard* ini dapat memberikan manfaat dalam pengambilan keputusan yang lebih baik.

3.3 Kendala yang Ditemukan

Selama menjalani program kerja magang di PT XYZ sebagai junior business analyst, terdapat sejumlah tantangan yang dihadapi oleh mahasiswa sebagai peserta magang. Secara umum, tantangan yang umum muncul selama proses magang untuk setiap fase adalah sebagai berikut:

3.3.1 Kendala Dalam Pemahaman Sistem dan Kebutuhan *User*

Pada tahap pemahaman sistem, peserta magang sebagai junior business analyst mengalami kendala dari kurangnya pemahaman yang mendalam akan kebutuhan *user*. Sebagai mahasiswa yang baru masuk ke dalam lingkungan nyata dunia kerja, terdapat beberapa kebutuhan *user* yang lebih kompleks dari biasanya. Kebutuhan yang lebih kompleks dan penyampaian dari perspektif *user* yang berbeda menjadi hambatan utama bagi peserta magang dalam memperoleh pemahaman yang memadai terkait sistem yang akan dibuat, yang mengakibatkan pembuatan *prototype* sistem menjadi kurang tepat sehingga perlu dilakukan pengecekan ulang dan pemberian masukan oleh mentor.

3.3.2 Kendala Dalam Penyusunan Dokumentasi

Selama proses penyusunan dokumentasi sistem, kendala utama yang dihadapi adalah kekurangan pemahaman tentang bagaimana menyusun dokumentasi sistem secara rinci dan mendalam. Penyusunan dokumentasi untuk beberapa sistem yang dikerjakan selama periode magang memerlukan tingkat detail yang tinggi, dengan begitu dokumentas yang dibuat memungkinkan tim pengembang yang bertanggung jawab atas pembuatan sistem untuk memahami desain dan alur sistem yang akan di kembangkan dengan lebih efektif. Selain itu, kurangnya pemahaman tentang proses bisnis secara luas terkadang menjadi kendala dalam pembuatan diagram alur proses dan topologi dari sistem yang sedang dikerjakan.

3.3.3 Adaptasi Dengan Lingkungan Kerja Perusahaan

Menghadapi tantangan dalam beradaptasi dan menyesuaikan diri dengan lingkungan kerja yang baru di suatu perusahaan menjadi kendala umum bagi peserta magang. Hal ini disebabkan oleh adanya beberapa kebijakan dan budaya

kerja yang baru dan berbeda dari apa yang biasanya mereka hadapi. Dalam konteks profesional, peserta magang seringkali dihadapkan pada tantangan untuk beradaptasi dan menyesuaikan diri dengan lingkungan kerja yang baru di suatu organisasi. Hal ini dapat menjadi hambatan, mengingat adanya berbagai kebijakan dan budaya kerja yang mungkin belum pernah ditemui sebelumnya. Kebijakan-kebijakan tersebut dapat mencakup berbagai aspek, mulai dari etika kerja, hingga tata cara berkomunikasi di tempat kerja. Sementara itu, budaya kerja dapat mencakup nilai-nilai dan perilaku yang diharapkan dari setiap karyawan di perusahaan tersebut. Selain itu, peserta magang juga perlu memahami dan menerima bahwa setiap organisasi memiliki cara kerja dan sistem yang unik. Oleh karena itu, diperlukan pembelajaran untuk memahami cara kerja dan sistem tersebut untuk dapat berkontribusi secara efektif, dimana hal ini tentunya membutuhkan waktu dan usaha yang tidak sebentar.

3.4 Solusi atas Kendala yang Ditemukan

3.4.1 Solusi Dalam Mengatasi Kendala Pemahaman Sistem

Solusi utama untuk mengatasi kendala dalam pemahaman sistem dan kebutuhan *user* adalah melakukan diskusi dengan intens bersama mentor. Diskusi ini diperlukan karena mentor memiliki pengalaman, pengetahuan, dan perspektif yang dapat membantu dalam memahami kebutuhan *user* dengan lebih baik. Selain itu, melalui diskusi ini, dapat dilakukan tanya jawab kepada mentor dan rekan kerja lainnya, terutama dengan mentor lainnya yang memiliki pengalaman dalam proyek dalam sistem yang serupa. Mentor juga selalu memberikan arahan tentang proses bisnis dan studi kasus yang sudah dikerjakan, yang dapat memperdalam pemahaman tentang proses bisnis. Solusi yang dilakukan tidak hanya bersama mentor dan rekan lainnya, akan tetapi juga dilakukan secara mandiri melalui eksplorasi dalam hal yang belum terlalu dipahami seperti proses bisnis melalui internet. Dengan demikian, pemahaman tentang kebutuhan pengguna dan proses bisnis menjadi lebih baik, dan pembuatan antarmuka sistem menjadi lebih tepat dan efektif.

3.4.2 Solusi Dalam Mengatasi Kendala Penyusunan Dokumentasi

Sejalan dengan permasalahan sebelumnya, solusi utama untuk mengatasi kendala dalam penyusunan dokumentasi adalah aktif bertanya kepada mentor dan selalu mencari referensi dokumentasi yang sudah dibuat sebelumnya. Dengan bertanya kepada mentor, maka peserta magang dapat memahami apa yang masih kurang atau perlu ditambah dari dokumentasi sistem yang dibuat. Mentor, dengan pengalaman mereka, seringkali memberikan masukan berharga dalam proses pembuatan dokumentasi. Di sisi lain, dengan mencari referensi dari dokumentasi yang telah ada, peserta magang dapat memahami struktur, pola, dan alur penulisan dokumentasi, serta mendapatkan gambaran tentang tingkat detail yang diperlukan dalam penulisan dokumentasi sistem yang tepat.

3.4.3 Solusi Dalam Beradaptasi Dengan Lingkungan Kerja Perusahaan

Untuk mengatasi kendala dalam beradaptasi dan menyesuaikan diri dengan lingkungan kerja baru di perusahaan, peserta magang memiliki sikap yang terbuka dan proaktif dalam belajar. Sikap proaktif yang dapat ditunjukkan yakni berani bertanya dan mencari informasi yang dibutuhkan untuk memahami kebijakan dan budaya kerja di perusahaan tersebut. Dengan demikian, peserta magang dapat memahami ekspektasi dan tuntutan yang ada, serta cara kerja yang efektif dalam lingkungan tersebut. Selain itu, mentor atau supervisor berperan penting sebagai sumber informasi dan bimbingan. Mentor atau supervisor biasanya memiliki pengetahuan dan pengalaman yang luas tentang perusahaan dan dapat memberikan arahan dan saran yang berguna untuk membantu peserta magang beradaptasi.

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