

BAB 3

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Dalam pelaksanaan magang di PT Bonum Mata Asia, posisi yang dijalankan adalah sebagai anggota *Development Team*. Divisi ini bertanggung jawab atas aspek teknis pengembangan perangkat lunak, meliputi perancangan komponen sistem, implementasi fitur, integrasi dengan modul lain, serta pengujian untuk memastikan kesesuaian fungsi dengan kebutuhan yang telah ditetapkan. *Development Team* berperan dalam merealisasikan hasil analisis kebutuhan dengan menerjemahkan spesifikasi fungsional menjadi solusi teknis yang optimal. Selama magang, tugas utama berfokus pada proses implementasi dan penyempurnaan fitur dalam aplikasi manajemen proyek.

Koordinasi selama magang dilakukan secara daring bersama mentor dari *Development Team*. Komunikasi harian terutama menggunakan *WhatsApp* untuk pengiriman dan penerimaan tugas, diskusi singkat terkait kendala teknis, klarifikasi instruksi, serta pelaporan progres pekerjaan. Untuk pembahasan yang lebih mendalam, seperti penanganan masalah kompleks, penjelasan teknis detail, atau sesi demo dan *screen sharing*, digunakan *Google Meet*. Kombinasi metode komunikasi ini mendukung alur kerja yang efisien, mempercepat penyelesaian masalah, serta memungkinkan proses bimbingan yang terstruktur meskipun magang dilaksanakan secara jarak jauh.

3.2 Tugas yang dilakukan

Selama periode magang di PT Bonum Mata Asia, kegiatan difokuskan pada perancangan dan pengembangan aplikasi manajemen proyek berbasis web untuk salah satu klien perusahaan. Aplikasi tersebut dibangun menggunakan *platform Scriptcase* dengan tujuan untuk mendigitalisasi alur bisnis pada sektor konstruksi. Sistem yang dikembangkan mencakup beberapa fitur utama, antara lain pembuatan kontrak berdasarkan templat yang telah ditetapkan, pengelolaan Rencana Anggaran Biaya (RAB), pencatatan laporan harian dan kemajuan (*progress*) proyek, serta pengaturan versi untuk dokumen dan gambar proyek. Melalui kegiatan ini, dihasilkan solusi perangkat lunak yang terintegrasi guna meningkatkan efisiensi, akurasi, dan transparansi dalam proses pengelolaan proyek di lingkungan klien.

3.3 Uraian Pelaksanaan Magang

Rangkaian tugas yang telah diuraikan pada bagian sebelumnya kemudian dilaksanakan melalui tahapan pengembangan yang sistematis, sebagaimana dijelaskan pada bagian berikut.



Gambar 3.1. Alur proses pengerjaan proyek di PT Bonum Mata Asia

Pelaksanaan kegiatan magang di PT Bonum Mata Asia dilakukan melalui serangkaian tahapan yang sistematis. Tahapan tersebut mencakup perencanaan proyek hingga proses pengembangan dan evaluasi teknis. Seluruh aktivitas magang berpusat pada proyek pengembangan aplikasi manajemen proyek berbasis web untuk salah satu klien perusahaan. Proyek ini memberikan kesempatan untuk mengikuti seluruh siklus pengembangan perangkat lunak, mulai dari analisis kebutuhan, perancangan sistem, hingga implementasi dan penyempurnaan fitur.

Tahap pertama dimulai dengan pertemuan bersama pihak klien untuk melakukan identifikasi kebutuhan sistem secara menyeluruh. Melalui sesi diskusi tersebut, diperoleh pemahaman mengenai alur bisnis yang berjalan di klien serta permasalahan yang dihadapi dalam pengelolaan proyek konstruksi. Setelah kebutuhan utama (*user requirements*) terkumpul, kegiatan berlanjut dengan pertemuan lanjutan bersama tim internal untuk membahas aspek teknis, termasuk rancangan arsitektur sistem, pembagian modul, serta penentuan teknologi yang akan digunakan. Hasil pembahasan internal ini kemudian dipresentasikan kembali kepada klien untuk mencapai kesepakatan bersama dan memastikan bahwa rancangan sistem sesuai dengan kebutuhan pengguna. Pada tahap ini, proses negosiasi dan penyesuaian juga dilakukan untuk menghasilkan keputusan yang saling menguntungkan sebelum pembuatan kontrak dimulai.

Setelah tahap perencanaan dan validasi kebutuhan selesai, kegiatan berlanjut ke fase teknis pengembangan aplikasi. Tahap ini diawali dengan proses

penyiapan lingkungan kerja (*environment setup*), yang mencakup konfigurasi server, pembuatan basis data awal, serta penyesuaian pengaturan pada *platform* Scriptcase. Pengembangan sistem dilakukan secara bertahap, dimulai dengan pembuatan modul utama atau *master data* yang menjadi fondasi bagi fitur-fitur lainnya.

Selama proses pengembangan, dilakukan komunikasi rutin dengan pembimbing magang untuk memastikan arah implementasi sesuai dengan standar perusahaan. Diskusi dilakukan hampir setiap hari melalui pertemuan daring guna membahas kendala teknis, perbaikan rancangan, atau penambahan fitur baru. Pendekatan ini serupa dengan metode pengembangan *Agile*, di mana progres pekerjaan terus dievaluasi dan disesuaikan berdasarkan kebutuhan yang muncul selama proses berlangsung, meskipun tidak diterapkan secara ketat dengan jadwal *sprint* atau pelaporan formal.

Dalam praktiknya, beberapa penyesuaian teknis dilakukan seiring perkembangan proyek. Misalnya, terdapat perubahan struktur basis data seperti penambahan tabel atau kolom baru untuk menyesuaikan kebutuhan sistem yang berkembang. Setiap perubahan tersebut dikonsultasikan terlebih dahulu dengan pembimbing untuk memastikan konsistensi data dan kestabilan sistem. Melalui proses kerja yang iteratif dan kolaboratif ini, pengembangan aplikasi dapat berjalan secara dinamis sekaligus tetap terarah, menghasilkan solusi yang sesuai dengan kebutuhan klien dan standar pengembangan perangkat lunak di lingkungan PT Bonum Mata Asia. Rincian pekerjaan yang dilakukan setiap minggunya selama periode magang didokumentasikan dalam Tabel 3.1 berikut.

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

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

Minggu Ke-	Pekerjaan yang dilakukan
1	Briefing awal, pengenalan, persiapan magang, instalasi Scriptcase dan PostgreSQL, serta pengenalan struktur project perusahaan.
2	Pengujian project Scriptcase, pembuatan website sederhana, penerapan <i>template</i> , dan pengembangan website.
3	Evaluasi website, penyelesaian website, pembelajaran server hosting dan database, serta diskusi flow awal dengan klien.
4	Pembuatan dan revisi database awal, memulai project utama, serta pengembangan website untuk project lain.
5	Presentasi website, pengisian konten dan aset, serta pengembangan lanjutan aplikasi.
6	Pengembangan flow master, pembuatan <i>template</i> dokumen HTML, serta pengerjaan awal <i>security</i> dan halaman <i>login</i> .
7	Penyempurnaan flow master dan <i>security</i> , perbaikan bug, serta pengerjaan awal modul <i>project management</i> .
8	Pengembangan lanjutan aplikasi <i>project management</i> dan perbaikan masalah <i>login</i> .
9	Penyesuaian project terhadap perubahan database dan pengembangan fitur utama aplikasi.
10	Penanganan bug, pembelajaran server, serta pengembangan project sesuai hasil meeting.
11	Pembahasan masalah aplikasi dengan klien dan penyesuaian project berdasarkan hasil meeting.
12	Pengembangan lanjutan aplikasi, perubahan aset, serta perbaikan halaman depan dan <i>login</i> .
13	Meeting perubahan struktur database dan pembuatan project <i>dummy</i> sebagai persiapan lanjutan project utama.
14	<i>Push</i> aplikasi ke server, pengujian dan standarisasi fitur, serta pembaruan database termasuk <i>mass insert</i> .
15	Meeting pengembangan fitur, penerimaan masukan, pembaruan aplikasi, dan <i>push</i> ke server.
16	Meeting pasca-deployment, <i>trial and error</i> fitur produksi, serta penyempurnaan fitur aplikasi.

3.3.1 Requirements Analysis

Tahap analisis kebutuhan (*requirements analysis*) merupakan langkah awal yang sangat penting dalam proses pengembangan sistem. Pada tahap ini, dilakukan pengumpulan informasi secara menyeluruh melalui diskusi dan pertemuan dengan pihak klien untuk memahami alur bisnis, kebutuhan operasional, serta permasalahan yang dihadapi dalam pengelolaan proyek konstruksi.

A Kebutuhan Fungsional (*Functional Requirements*)

Kebutuhan fungsional merupakan fitur atau kemampuan yang harus disediakan oleh sistem agar dapat mendukung proses bisnis klien secara optimal.

1. **Pembuatan Kontrak Berbasis Templat**

Sistem harus menyediakan fitur untuk membuat dokumen kontrak secara otomatis berdasarkan templat yang telah ditentukan oleh perusahaan.

2. **Pengelolaan Rencana Anggaran Biaya (RAB)**

Sistem harus mampu mencatat, menyimpan, dan mengelola data RAB, termasuk perhitungan biaya satuan, total biaya, dan pembaruan data material atau pekerjaan.

3. **Pencatatan Laporan Harian Proyek**

Sistem harus menyediakan antarmuka bagi pengguna untuk mencatat aktivitas harian proyek, seperti progres pekerjaan, jumlah tenaga kerja, dan kondisi lapangan.

4. **Pelacakan Kemajuan Proyek (*Progress Tracking*)**

Sistem harus mampu menampilkan perkembangan proyek berdasarkan data laporan harian, sehingga manajemen dapat melakukan pemantauan secara *real-time*.

5. **Pengaturan Versi Dokumen dan Gambar Proyek (*Version Control*)**

Sistem harus menyediakan mekanisme pengaturan versi agar perubahan pada dokumen atau gambar proyek dapat terdokumentasi dengan baik.

6. **Visualisasi Proyek melalui *Gantt Chart* dan *S-Curve***

Aplikasi menyediakan fitur visualisasi progres proyek yang terdiri atas *Gantt Chart* dan *S-Curve* untuk menampilkan rencana dan realisasi pekerjaan secara

terstruktur, sehingga memudahkan pemantauan jadwal serta perbandingan antara progres terencana dan aktual.

7. Manajemen Pengguna dan Hak Akses (*User Management & Access Control*)

Sistem harus menyediakan fitur untuk mengelola akun pengguna serta menetapkan hak akses tertentu sesuai peran masing-masing.

B Kebutuhan Non-Fungsional (*Non-Functional Requirements*)

Kebutuhan non-fungsional berfokus pada kualitas, keamanan, dan performa sistem.

1. Keamanan Data (*Data Security*)

Sistem harus menjamin keamanan data melalui mekanisme autentikasi, otorisasi, dan perlindungan dari akses tidak sah.

2. Kemudahan Penggunaan (*Usability*)

Antarmuka pengguna harus mudah dipahami, intuitif, dan mendukung efisiensi kerja pengguna dengan berbagai tingkat literasi teknologi.

3. Kinerja Sistem yang Stabil (*Performance & Stability*)

Sistem harus mampu berjalan secara stabil, menangani jumlah data besar, serta memberikan waktu respons yang cepat.

4. Ketersediaan Sistem (*Availability*)

Sistem harus dapat diakses kapan saja sesuai kebutuhan operasional perusahaan, terutama bagi pengguna lapangan.

5. Skalabilitas (*Scalability*)

Sistem harus dapat dikembangkan lebih lanjut jika perusahaan membutuhkan penambahan modul atau peningkatan kapasitas data.

C Analisis Kelayakan (*Feasibility Analysis*)

Seluruh kebutuhan di atas dianalisis secara internal oleh tim pengembang untuk menilai kelayakan dari sisi teknis, waktu, serta ketersediaan sumber daya. Hasil analisis kebutuhan tersebut menjadi landasan utama dalam penyusunan rancangan arsitektur dan komponen sistem yang dibahas pada bagian berikutnya.

3.3.2 System Design

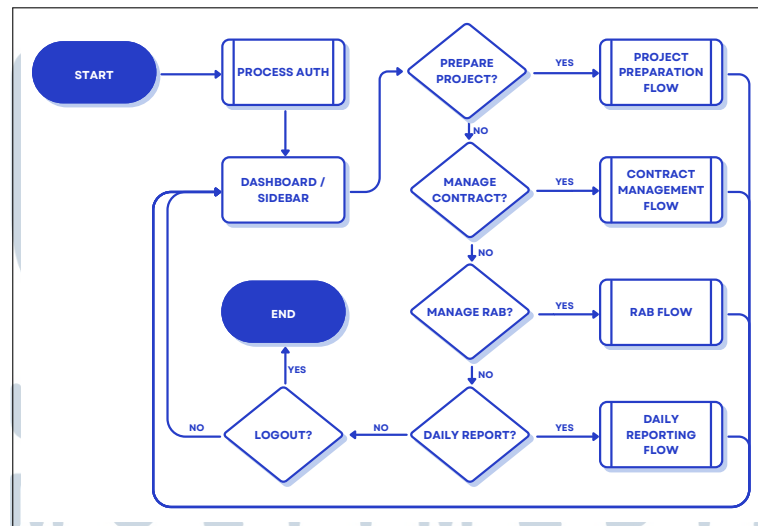
Tahap perancangan sistem (*system design*) bertujuan untuk menerjemahkan hasil analisis kebutuhan ke dalam bentuk rancangan teknis yang lebih terstruktur.

A Flowchart

Flowchart pada bagian ini menjelaskan alur logika utama dalam aplikasi manajemen proyek konstruksi, dimulai dari tahap autentikasi pengguna hingga penyajian atau visualisasi data. Setiap rangkaian proses dijabarkan secara rinci sebagai berikut:

1. Alur Utama Sistem (*Main System Flow*)

Gambar 3.2 menggambarkan alur kerja aplikasi secara keseluruhan. Sistem ini menggunakan navigasi berbasis *sidebar* yang selalu tampil di sisi layar. Setelah berhasil *login*, pengguna akan langsung diarahkan ke halaman *Dashboard* sebagai tampilan awal. Lalu, pengguna bebas berpindah ke fitur lain, seperti manajemen proyek, kontrak, atau laporan dengan menggunakan *sidebar*. Dengan menggunakan *sidebar*, pengguna dapat bernavigasi dengan cepat dan mudah.

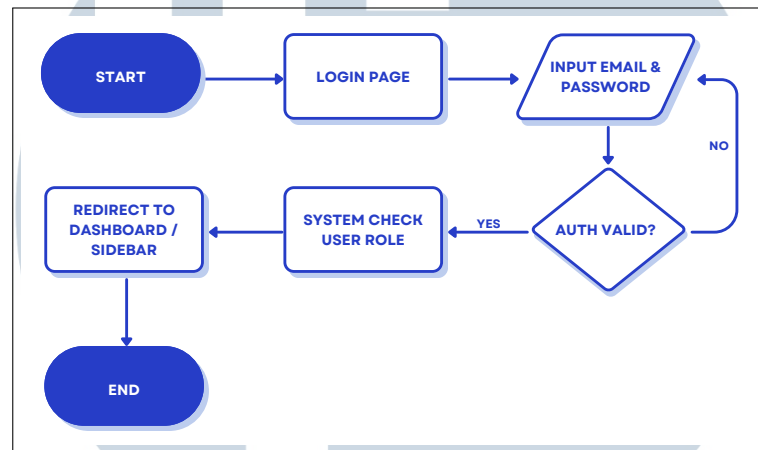


Gambar 3.2. *Flowchart* Utama Sistem

2. Alur Autentikasi (*Authentication Flow*)

Detail proses *login* ditunjukkan pada Gambar 3.3. Alur dimulai ketika pengguna membuka aplikasi dan diminta mengisi kredensial berupa *email*

serta *password*. Sistem kemudian melakukan pengecekan dan validasi terhadap data yang dimasukkan. Apabila kredensial dinyatakan benar, sistem akan melanjutkan dengan mengidentifikasi peran (*user role*) yang dimiliki pengguna. Setelah proses verifikasi selesai, pengguna secara otomatis diarahkan menuju *dashboard* utama.

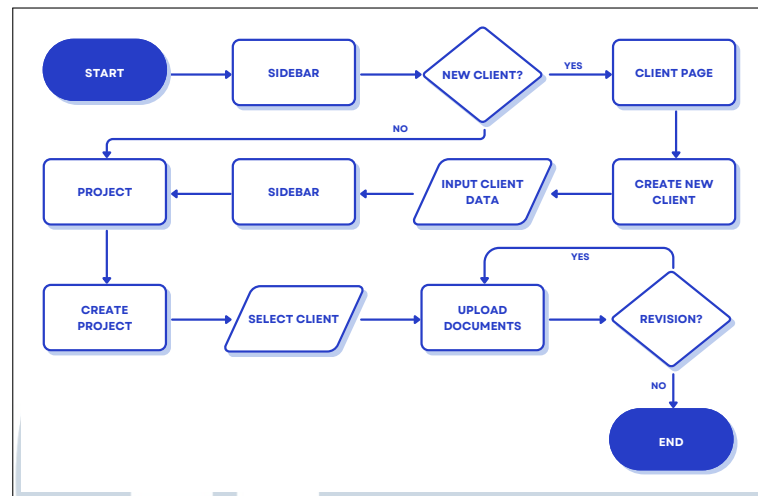


Gambar 3.3. *Flowchart Authentication*

3. Alur Pembuatan Proyek Baru (*Project Initiation Flow*)

Gambar 3.4 menunjukkan bagaimana proses pembuatan proyek baru berjalan. Langkah pertama adalah pengecekan data klien; jika klien belum terdaftar, sistem akan meminta pengguna untuk mendaftarkan klien terlebih dahulu. Setelah data klien siap, pengguna bisa mulai membuat proyek dengan mengisi informasi dasar serta mengunggah fail pendukung. Terdapat proses validasi di mana dokumen yang belum sesuai harus direvisi sebelum data proyek bisa disimpan ke dalam sistem.

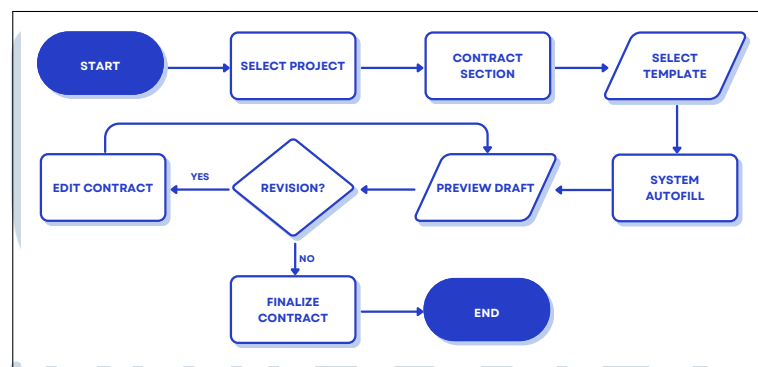
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.4. Flowchart Pembuatan Proyek Baru

4. Alur Pengelolaan Kontrak (*Contract Management Flow*)

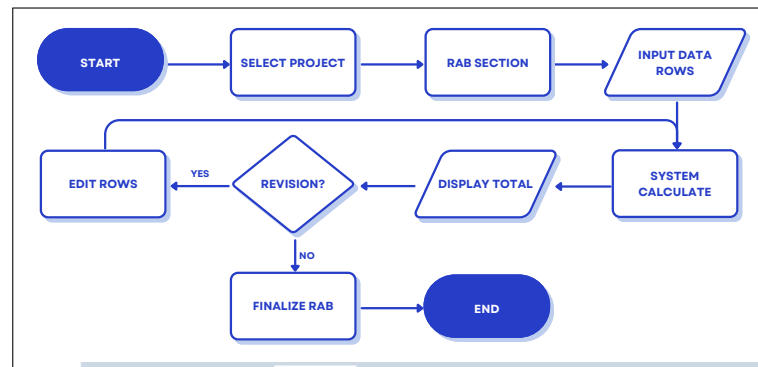
Proses pembuatan kontrak digital dapat dilihat pada Gambar 3.5. Alur ini dimulai ketika pengguna memilih proyek aktif, lalu masuk ke menu kontrak untuk memilih templat. Sistem akan otomatis mengisi draf kontrak (*auto-fill*) menggunakan data proyek. Pengguna dapat melihat pratinjau (*preview*) dan mengedit pasal jika diperlukan. Jika draf sudah sesuai, kontrak akan difinalisasi dan statusnya dikunci.



Gambar 3.5. Flowchart Fitur Kontrak Proyek

5. Alur Rencana Anggaran Biaya (*RAB Flow*)

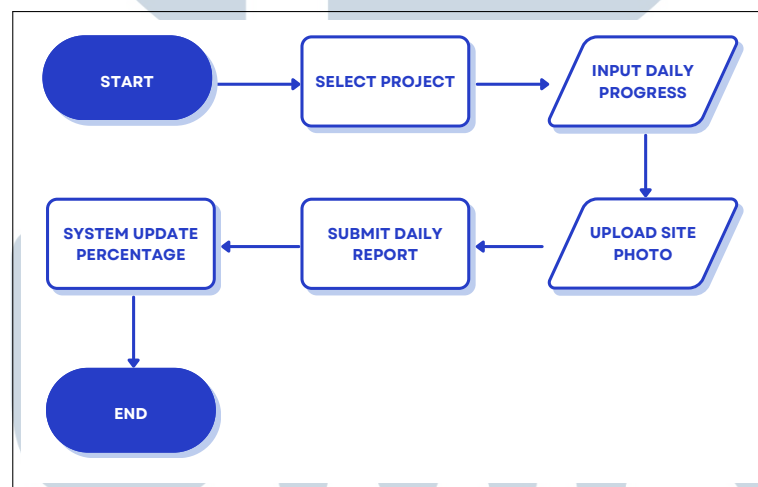
Pada Gambar 3.6, alur penyusunan RAB dimulai dengan memilih proyek dan memasukkan rincian komponen biaya. Sistem kemudian menghitung total estimasi secara otomatis. Pengguna bisa mengecek ulang total anggaran dan merevisi *item* jika terjadi kesalahan input. Jika angka sudah disetujui, RAB disimpan sebagai acuan utama anggaran proyek.



Gambar 3.6. *Flowchart* Fitur RAB Baru

6. Alur Laporan Harian (*Daily Reporting Flow*)

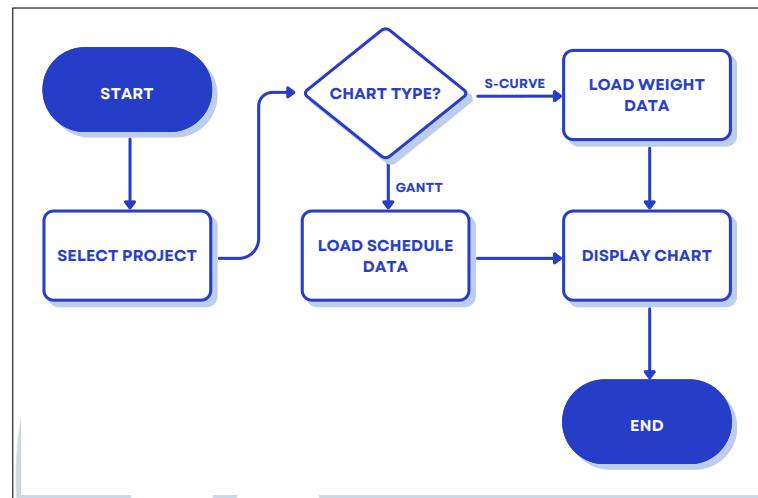
Fitur pelaporan lapangan bisa dilihat pada Gambar 3.7. Proses dimulai saat supervisor lapangan memilih proyek, kemudian menginput progres harian dan mengunggah foto dokumentasi. Begitu laporan dikirim (*submit*), sistem langsung memperbarui data persentase aktual proyek tersebut di database.



Gambar 3.7. *Flowchart* Fitur Laporan Harian

7. Alur Visualisasi Progres (*Visualization Flow*)

Terakhir, Gambar 3.8 menjelaskan alur visualisasi data. Pengguna memulainya dengan masuk ke halaman proyek dan memilih proyek yang ingin dipantau. Di dalam menu tersebut, tersedia pilihan tampilan *Gantt Chart* untuk mengecek jadwal ataupun *S-Curve* untuk membandingkan rencana dengan realisasi. Sistem akan menarik data dari modul RAB dan Laporan Harian, lalu menyajikannya dalam bentuk grafik interaktif.

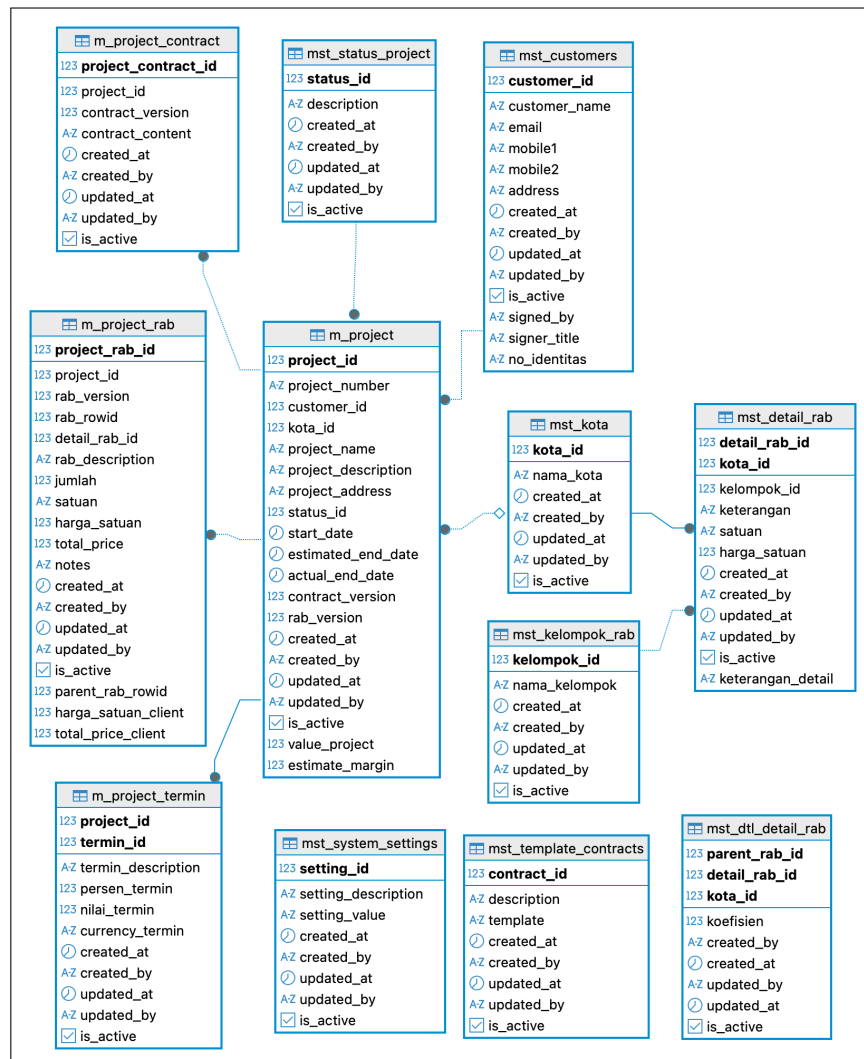


Gambar 3.8. *Flowchart* Fitur Visualisasi Progres

B Entity Relationship Diagram

Gambar 3.9 menampilkan *Entity Relationship Diagram* (ERD) yang digunakan dalam aplikasi manajemen proyek. Struktur basis data dirancang untuk menghubungkan data master, seperti data klien dan proyek, dengan data transaksional yang mencakup kontrak, RAB, dan laporan harian proyek. Relasi antar tabel disusun untuk memastikan keterkaitan data tetap konsisten dan mendukung proses pengolahan data secara terintegrasi.

Selama proses pengembangan, struktur basis data mengalami beberapa penyesuaian sesuai dengan kebutuhan sistem yang berkembang. Penyesuaian tersebut meliputi perubahan relasi antar tabel, penambahan tabel baru, serta modifikasi atribut data. Perlu diketahui bahwa ERD yang ditampilkan belum merupakan rancangan akhir. Hal ini disebabkan oleh proses pengembangan aplikasi yang masih berlangsung. Oleh karena itu, struktur basis data masih bersifat dinamis dan dapat mengalami perubahan atau penyesuaian lebih lanjut sesuai dengan kebutuhan sistem yang berkembang. ERD ini merepresentasikan kondisi basis data sejauh tahap pengembangan yang telah dicapai selama periode magang.



Gambar 3.9. ERD Aplikasi Manajemen Proyek

3.3.3 Implementation and Development

Tahap implementasi dan pengembangan dilakukan untuk menerapkan rancangan sistem ke dalam bentuk aplikasi yang dapat digunakan oleh klien. Pada tahap ini, aspek arsitektur, pemilihan teknologi, lingkungan pengembangan, serta konfigurasi server menjadi fokus utama untuk memastikan sistem berjalan secara optimal dan sesuai kebutuhan operasional perusahaan.

A Arsitektur Sistem

Sistem dikembangkan dengan menerapkan arsitektur *client-server*. Pada pola ini, pengguna berinteraksi dengan aplikasi melalui *web browser* di sisi

client, sedangkan seluruh pemrosesan logika aplikasi, pengelolaan data, serta penyimpanan dilakukan terpusat di sisi *server*. Pendekatan ini dipilih karena mampu mendukung skalabilitas, mempermudah proses pemeliharaan, serta menjamin kompatibilitas penggunaan di berbagai jenis perangkat. Secara garis besar, arsitektur sistem mencakup:

1. **Client**

Web browser yang digunakan oleh pengguna di lapangan maupun di kantor untuk mengakses antarmuka aplikasi tanpa memerlukan instalasi khusus.

2. **Application Server**

Aplikasi berbasis *PHP* yang dihasilkan oleh platform Scriptcase, yang bertanggung jawab terhadap pemrosesan *business logic*, mekanisme autentikasi, *routing* permintaan, serta pelaksanaan operasi CRUD terhadap data.

3. **Database Server**

PostgreSQL sebagai sistem manajemen basis data utama yang menyimpan informasi proyek, kontrak, RAB, laporan harian, serta berbagai parameter dan konfigurasi sistem.

Dengan pemisahan peran antara client, *application server*, dan *database server*, aliran data dapat dikontrol dengan lebih baik dan perubahan pada sisi server dapat dilakukan tanpa mengganggu cara pengguna mengakses aplikasi.

B Lingkungan dan Teknologi Pengembangan

Lingkungan pengembangan menggunakan platform Scriptcase sebagai kerangka utama untuk membangun modul *form*, *grid*, *control*, serta manajemen keamanan. Bahasa pemrograman yang digunakan adalah *PHP*, dengan PostgreSQL sebagai sistem manajemen basis data relasional.

Perangkat keras dan perangkat lunak yang digunakan untuk mendukung proses pengembangan adalah sebagai berikut:

1. **Perangkat Komputer:** MacBook Pro (Intel Core i5, 8 GB RAM, 256 GB SSD)
2. **Sistem Operasi:** macOS Monterey

3. **Tools Pengembangan:** Scriptcase, PostgreSQL, Web Browser (Google Chrome/Safari)

Perangkat tersebut dimanfaatkan untuk menjalankan lingkungan server lokal, merancang antarmuka modul, serta mengelola struktur basis data.

C Pembangunan dan Penyesuaian Modul Aplikasi

Proses implementasi dimulai dengan pembangunan *master data* sebagai fondasi utama aplikasi. Pada tahap ini dilakukan pembuatan struktur tabel beserta relasi antar-entitas, konfigurasi awal pada Scriptcase untuk menghasilkan modul dasar, serta pembuatan *form* input dan *grid* untuk kebutuhan pengelolaan data. Setelah modul dasar terbentuk, pengembangan dilanjutkan ke fitur-fitur utama, seperti pembuatan templat dokumen kontrak, pengelolaan Rencana Anggaran Biaya (RAB), pencatatan laporan harian, tampilan *progress* proyek secara visual, serta pengaturan versi dokumen dan gambar proyek. Penyesuaian antarmuka juga dilakukan agar tampilan lebih intuitif dan sesuai kebutuhan pengguna.

Selama proses tersebut, aspek keamanan sistem diterapkan melalui fitur bawaan Scriptcase, meliputi manajemen pengguna, pengaturan *role* dan hak akses, pembatasan akses modul berdasarkan peran, serta implementasi halaman login yang terintegrasi. Setiap modul diuji secara berkala guna memastikan tidak terdapat celah keamanan dan seluruh fungsi berjalan sesuai ketentuan.

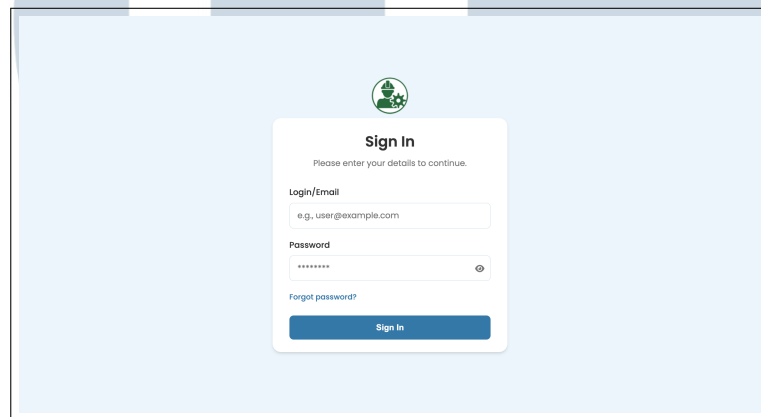
Selain itu, struktur basis data mengalami beberapa penyesuaian untuk mengikuti perkembangan kebutuhan sistem. Perubahan ini mencakup penambahan tabel baru, modifikasi kolom dan tipe data, pembaruan relasi, serta optimalisasi indeks untuk meningkatkan performa. Setiap perubahan dikonsultasikan dengan pembimbing magang agar integritas data tetap terjaga dan implementasi berjalan konsisten.

D Tampilan Antarmuka Aplikasi

Subbagian ini menampilkan beberapa contoh tampilan antarmuka (*user interface*) dari aplikasi manajemen proyek yang dikembangkan selama kegiatan magang. Tampilan antarmuka difokuskan pada peran setiap halaman dalam mendukung proses operasional sistem serta memastikan alur kerja pengguna berjalan secara jelas dan terstruktur. Setiap halaman disesuaikan dengan modul

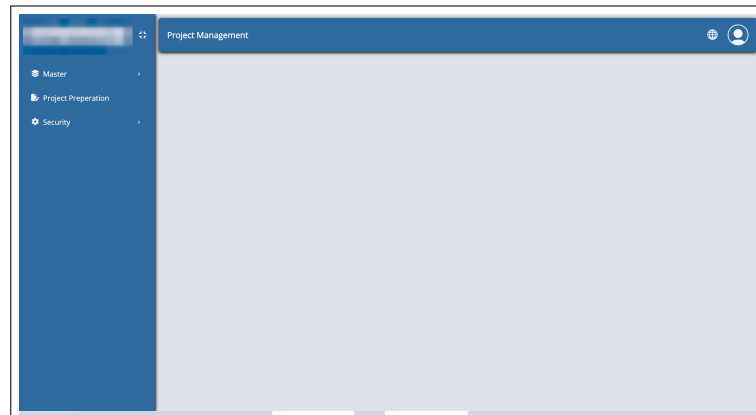
terkait dan dihubungkan langsung dengan data yang tersimpan dalam basis data sistem.

Halaman *login* merupakan titik awal akses pengguna ke dalam sistem, sebagaimana ditampilkan pada Gambar 3.10. Pada halaman ini, pengguna diminta untuk memasukkan kredensial berupa *email* dan *password* yang telah terdaftar. Sistem akan melakukan proses autentikasi untuk memverifikasi identitas pengguna serta menentukan hak akses sesuai dengan peran (*role*) yang dimiliki. Mekanisme ini bertujuan untuk menjaga keamanan sistem dan membatasi akses fitur berdasarkan kewenangan masing-masing pengguna.



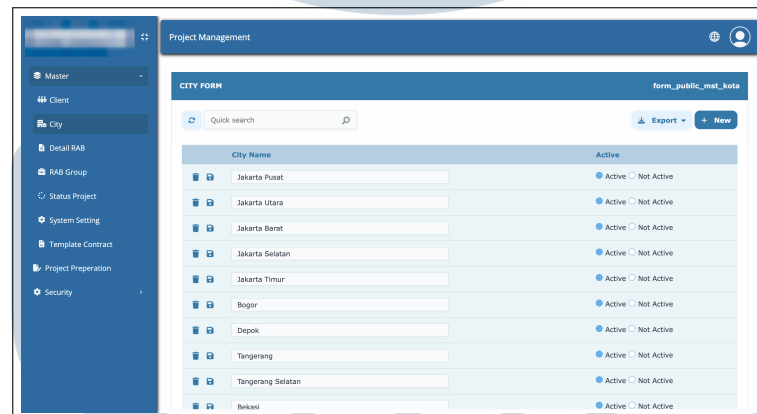
Gambar 3.10. Halaman Login Aplikasi Manajemen Proyek

Halaman *dashboard* berfungsi sebagai halaman utama setelah pengguna berhasil melakukan autentikasi, seperti ditunjukkan pada Gambar 3.11. Pada tahap pengembangan saat ini, tampilan dashboard masih bersifat dasar dan belum menampilkan ringkasan data secara visual. Meskipun demikian, halaman ini telah disiapkan sebagai pusat navigasi menuju modul-modul utama aplikasi, seperti pengelolaan klien, proyek, kontrak, dan laporan. Dashboard dirancang agar dapat dikembangkan lebih lanjut untuk menampilkan informasi ringkas mengenai status proyek dan aktivitas pengguna di masa mendatang.



Gambar 3.11. Halaman Dashboard Aplikasi Manajemen Proyek

Gambar 3.12 merupakan halaman *master data* kota. Melalui halaman ini, admin dapat mengelola data kota dalam sistem. Data kota ditampilkan melalui bentuk tabel yang dapat di edit atau disebut *form editable grid* di Scriptcase. Halaman ini berfungsi untuk mendukung pengelolaan proyek yang dilakukan di berbagai kota, dimana setiap kota memiliki harga yang berbeda-beda untuk bahan, alat, dan upah.



Gambar 3.12. Halaman *Master Data* Kota

Halaman Pengelolaan klien dapat dilihat pada Gambar 3.13. Halaman ini khusus admin untuk mengelola data klien yang terdaftar dalam sistem. Data ditampilkan melalui bentuk tabel atau disebut *grid* di Scriptcase. Tabel memuat data utama klien, seperti nama, email, kontak, dan alamat. Admin dapat menambah (*create*), mengubah (*edit*), dan menghapus (*delete*) data klien sesuai kebutuhan. Modul ini digunakan sebagai *master data* untuk modul proyek dan kontrak.

Action	Client Name	Email	Phone Number 1	Phone Number 2	Address	Active
...	Budi Santoso	budi.santoso@example.com	081234567890	081298765432	Jl. Merdeka No. 123, Jakarta Pusat	✓
...	Ani Yulianti	ani.yulianti@example.com	085678901234		Jl. Sudirman Kav. 56, Jakarta Selatan	✓
...	Joko Susilo	joko.s@example.com	087812345678		Perumahan Griya Indah Blok A5/10, Bogor	✓
...	Siti Aminah	siti.aminah@example.com	081567890123	081511223344	Jl. Margonda Raya No. 25, Depok	✓
...	Eko Prasetyo	eko.prasetyo@example.com	089987654321		Komp. Villa Melati Mas, Tangerang Selatan	✓
...	Dewi Lestari	dewi.lestari@example.com	082123456789		Jl. Patriot No. 45, Bekasi Barat	✓
...	Agus Wijaya	agus.wijaya@example.com	081809876543	081812348765	Summarecon Gading Serpong, Tangerang	✓
...	Rina Hartati	rina.h@example.com	081312345678		Jl. Gajah Mada No. 99, Jakarta Barat	✓

Gambar 3.13. Halaman Pengelolaan Data Klien

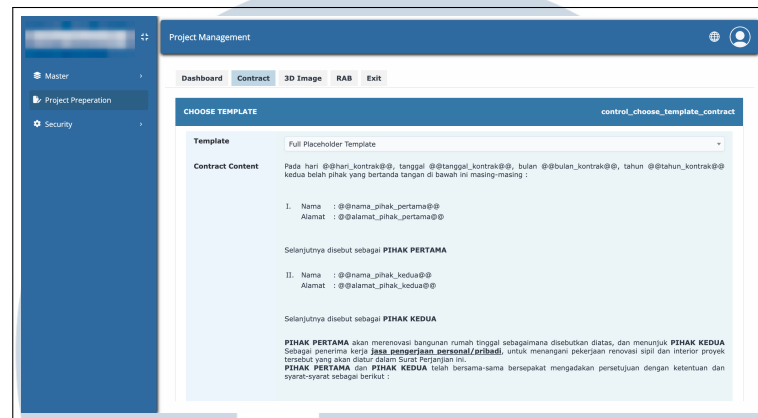
Halaman proyek dapat dilihat pada Gambar 3.14. Halaman ini berfungsi sebagai pengelolaan proyek yang ditampilkan dalam bentuk *grid*. Setiap baris pada tabel merupakan satu proyek yang mempunyai informasi seperti nama, klien, dan status proyek. Di setiap baris proyek, terdapat aksi lanjutan untuk melihat detail proyek atau melakukan pengubahan data, melalui *icon* di kolom *Action* di paling kiri baris.

Action	Project Number	Project Name	Project Address	Client	Status Project	Project Description	Active
...	EXMP/001/12/25	Example Project GS	Summarecon Gading Serpong, Tangerang	Agus Wijaya	Initialized	Project Rumah Pak Agus 2 Lantai	✓

Gambar 3.14. Halaman Pengelolaan Data Proyek

Setiap data proyek yang dipilih akan diarahkan ke halaman detail proyek. Halaman detail proyek disusun menggunakan format tab untuk memisahkan dan mengelompokkan fitur-fitur utama dalam satu proyek, seperti *Dashboard* Proyek, Kontrak, Gambar, serta RAB. Pada tahap pengembangan saat ini, tab Kontrak telah diimplementasikan dan dapat digunakan, sementara tab-tab lainnya masih berada pada fase pengembangan. Gambar 3.15 menunjukkan halaman pembuatan kontrak baru pada tahap pemilihan *template* kontrak yang dapat disesuaikan dengan jenis

dan kebutuhan proyek, sehingga pengguna dapat menyusun kontrak tanpa perlu membuatnya dari awal.



Gambar 3.15. Halaman Pemilihan *Template* Kontrak

3.3.4 Testing and Evaluation

Tahap pengujian dan evaluasi (*testing and evaluation*) dilakukan untuk memastikan bahwa seluruh fungsi aplikasi berjalan sesuai dengan kebutuhan serta meminimalisir terjadinya kesalahan (*bug*). Metode pengujian yang digunakan adalah *Black Box Testing*, di mana pengujian berfokus pada fungsionalitas sistem berdasarkan hubungan antara masukan (*input*) dan keluaran (*output*) tanpa memperhatikan struktur kode internal.

Pengujian difokuskan pada modul-modul dasar yang berfungsi sebagai fondasi sistem, seperti pengelolaan data master, pembuatan data proyek, serta pengamanan sistem melalui autentikasi pengguna dan pengaturan hak akses berbasis peran (*role-based access*). Setiap fungsi diuji menggunakan data uji yang telah ditentukan untuk memastikan bahwa perilaku sistem sesuai dengan spesifikasi fungsional yang dirancang.

Status pengujian pada Tabel 3.2 menunjukkan kesesuaian antara keluaran sistem dengan hasil yang diharapkan. Setiap skenario pengujian yang menghasilkan keluaran sesuai dengan spesifikasi fungsional dinyatakan **Valid (V)**, sedangkan skenario yang tidak menghasilkan keluaran sesuai dinyatakan **Tidak Valid (TV)**.

Tabel 3.2. Hasil Pengujian Black Box pada Modul Dasar Sistem

No	Modul	Skenario	Hasil yang Diharapkan	Status
1	Autentikasi Pengguna	Login dengan email dan kata sandi yang valid	Sistem mengarahkan pengguna ke halaman utama (<i>dashboard</i>)	V
2	Autentikasi Pengguna	Login dengan email atau kata sandi yang tidak valid	Sistem menampilkan pesan kesalahan autentikasi	V
3	Autentikasi Pengguna	Logout dari sistem	Sistem mengakhiri sesi dan kembali ke halaman login	V
4	Manajemen Pengguna	Menambahkan pengguna baru melalui formulir pengguna	Data pengguna tersimpan dan muncul pada daftar pengguna	V
5	Manajemen Pengguna	Mengubah data pengguna yang sudah ada	Perubahan data tersimpan dan muncul dengan benar	V
6	Data Master	Menambahkan data pada seluruh jenis data master	Data master berhasil disimpan pada sistem	V
7	Data Master	Menghapus data pada seluruh jenis data master	Data master berhasil dihapus dari sistem	V
8	Manajemen Proyek	Membuat data proyek baru melalui formulir proyek	Data proyek tersimpan dan dapat diakses oleh pengguna	V
9	Hak Akses Berbasis Peran	Mengakses menu yang tidak sesuai dengan peran pengguna	Sistem menolak akses dan tidak menampilkan menu terkait	V
10	Hak Akses Berbasis Peran	Mengakses menu sesuai dengan peran pengguna	Sistem menampilkan menu dan fitur yang diizinkan	V

3.3.5 Deployment and Maintenance

Tahap penerapan (*deployment*) dilakukan setelah aplikasi dinyatakan lolos uji internal. Aplikasi diunggah ke lingkungan produksi (*production environment*) yang menggunakan infrastruktur berbasis *Virtual Private Server* (VPS). Penggunaan VPS dipilih untuk memberikan aksesibilitas penuh secara daring dan fleksibilitas konfigurasi tanpa membebankan klien dengan pengadaan infrastruktur fisik *on-premise*.

Proses ini mencakup instalasi dan konfigurasi *web server*, migrasi basis data PostgreSQL ke lingkungan *cloud*, serta penyesuaian parameter koneksi pada Scriptcase. Pengelolaan server produksi dilakukan sepenuhnya oleh tim teknis guna memastikan konfigurasi keamanan dan ketersediaan layanan tetap terjaga.

Setelah aplikasi berjalan, masuk ke tahap pemeliharaan (*maintenance*). Pada tahap ini, dilakukan pemantauan terhadap performa aplikasi (*monitoring*) dan penanganan insiden jika terjadi kendala teknis. Pemeliharaan juga mencakup pembaruan sistem (*patching*) secara berkala untuk menjaga stabilitas sistem selama digunakan oleh operasional klien.

3.4 Kendala dan Solusi yang Ditemukan

3.4.1 Kendala

Selama pelaksanaan kegiatan magang, terdapat sejumlah hambatan yang dialami, baik dari sisi teknis maupun non-teknis. Adapun beberapa kendala utama yang dihadapi selama kegiatan magang adalah sebagai berikut:

1. Kurangnya pengalaman sebelumnya dalam menggunakan platform *low-code* Scriptcase, sehingga memerlukan waktu adaptasi tambahan untuk memahami struktur dan cara kerjanya
2. Perubahan struktur basis data yang terjadi selama proses pengembangan sehingga memerlukan penyesuaian ulang pada beberapa modul aplikasi.

3.4.2 Solusi

Untuk mengatasi kendala tersebut, langkah penyelesaian yang dilakukan adalah:

1. Mempelajari dokumentasi Scriptcase serta berkonsultasi dengan mentor untuk mempercepat pemahaman terhadap penggunaan platform.
2. Melakukan koordinasi dengan pembimbing sebelum melakukan perubahan struktur basis data untuk menjaga konsistensi, serta memperbarui modul aplikasi secara bertahap.

