Tugas Akhir

Perancangan Sistem Marketplace Digital Berbasis Web untuk Freelancer dan Klien sebagai Solusi Ekonomi Kreatif Alumni Universitas Muhammadiyah Magelang

Jenis Tugas Akhir

Terapan

Disusun sebagai salah satu syarat memperoleh gelar Ahli Madya Komputer (A.Md.Kom)

Disusun Oleh:

Ahmad Reza Pahlevi	**(NIM. 22.0502.0005)**
Ahmad Fajrulhaq	**(NIM. 22.0502.0006)**
Alssa Askfritho Amanata	**(NIM. 22.0502.0009)**

Pembimbing :
Mukhtar Hanafi, ST., MCs. (NIDN. 0602047502)
Bambang Pujiarto, M.Kom (NIDN. 0623107802)

Program Studi D3 Teknologi Informasi Fakultas Teknik

Universitas Muhammadiyah Magelang

Tahun 2025

Bab 1
Pendahuluan

1.1 Latar Belakang Masalah

Kebutuhan akan media digital untuk mendukung alumni dalam menjembatani kesempatan kerja freelance menjadi semakin penting di era transformasi digital saat ini. Alumni Universitas Muhammadiyah Magelang, khususnya dari bidang Teknologi Informasi, memiliki potensi besar untuk terlibat dalam ekosistem ekonomi kreatif yang menuntut kolaborasi fleksibel antara penyedia jasa dan pemberi kerja. Oleh karena itu, diperlukan suatu platform yang dapat mengakomodasi kebutuhan alumni dalam mengakses, menawarkan, dan mengelola proyek secara profesional.

Revolusi industri 4.0 mempengaruhi berbagai macam perubahan di banyak industri, contohnya bidang teknologi. Perubahan teknologi bergerak sangat cepat, dan pesat. Bidang teknologi sangat membantu mempermudah pekerjaan manusia contohnya penggunaan e-commerce . E-commerce berasal dari singkatan electronic commerce yaitu pemasaran melalui media elektronik mulai dari penjualan dan pembelian dari sebuah produk atau jasa dilakukan di internet. E-commerce memudahkan pembeli yang dulunya harus pergi ke pasar langsung sekarang hanya melalui internet . Ar-raniry et al. (2024)

Kebutuhan akan sistem digital yang mampu menyatukan freelancer dan klien telah menjadi fokus berbagai penelitian sebelumnya. Sari & Utami (2022) menegaskan bahwa lonjakan jumlah pekerja lepas selama masa pandemi semakin memperkuat urgensi hadirnya sistem kerja fleksibel berbasis teknologi. Penelitian tersebut menyimpulkan bahwa platform yang menyediakan fitur pengelolaan proyek serta komunikasi yang terpadu sangat dibutuhkan untuk mendukung produktivitas dan meningkatkan profesionalisme para freelancer.

Penelitian lain oleh Hidayat et al. (2022) menunjukkan bahwa para klien juga menghadapi tantangan besar dalam menemukan freelancer yang kompeten dan dapat dipercaya. Kurangnya transparansi serta tidak adanya sistem pengamanan dan verifikasi yang memadai sering kali menyebabkan proyek tidak berjalan sesuai harapan. Oleh karena itu, keberadaan platform yang mampu menyediakan sistem rating, ulasan pengguna, serta metode pembayaran yang aman menjadi sangat penting.

Berdasarkan berbagai temuan tersebut, maka diperlukan sebuah marketplace digital berbasis web yang tidak hanya menjadi jembatan antara freelancer dan klien, tetapi juga mampu menyediakan fitur-fitur yang mendukung profesionalisme, efisiensi, dan keberlanjutan dalam sektor ekonomi kreatif. Selain itu, perancangan platform ini juga

ditujukan untuk memberikan ruang produktif bagi alumni Universitas Muhammadiyah Magelang agar dapat terlibat aktif dalam ekosistem kerja digital melalui peran sebagai freelancer maupun klien.

Selain itu, alasan pemilihan topik ini juga dilatarbelakangi oleh kebutuhan alumni Universitas Muhammadiyah Magelang, khususnya dari Fakultas Teknik, akan platform digital yang mampu memfasilitasi mereka dalam menemukan peluang kerja freelance yang profesional dan terpercaya. Dengan hadirnya KerjaBos, diharapkan alumni memiliki akses yang lebih luas untuk menjalin kolaborasi dengan klien dan berkontribusi langsung dalam pengembangan sektor ekonomi kreatif melalui sistem kerja berbasis proyek.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka perumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana merancang sebuah platform web yang dapat mempertemukan freelancer dan klien dalam satu sistem?
2. Fitur-fitur inti apa saja yang harus dikembangkan untuk mendukung proses registrasi, posting proyek, pengajuan proposal, dan pemilihan freelancer?
3. Bagaimana merancang sistem transaksi yang aman dan terkelola, seperti top-up, dompet digital, dan verifikasi pembayaran dalam platform?

1.3 Tujuan Penelitian

Penelitian ini dilakukan dengan tujuan sebagai berikut:

1. Membangun sebuah platform web yang dapat mempertemukan freelancer dan klien dalam satu sistem yang terintegrasi.
2. Mengembangkan fitur utama seperti registrasi pengguna, posting proyek oleh klien, pengajuan proposal oleh freelancer, serta pemilihan freelancer oleh klien.
3. Merancang sistem transaksi yang aman, termasuk fitur dompet digital (wallet), proses top-up, dan verifikasi pembayaran.

1.4 Manfaat Penelitian

Adapun manfaat yang diharapkan dari penelitian ini antara lain:

1. **Manfaat Teoritis**

   Penelitian ini dapat menjadi referensi bagi pengembangan sistem informasi berbasis web, khususnya dalam bidang ekonomi kreatif dan digital entrepreneurship. Selain itu,

hasil penelitian ini juga dapat memperkaya literatur terkait pemanfaatan teknologi dalam menciptakan ekosistem kerja yang fleksibel dan profesional.

2. **Manfaat Praktis**

   Hasil dari penelitian ini diharapkan dapat memberikan solusi nyata bagi freelancer dan klien dalam menjalin kerja sama secara efisien dan aman. Platform yang dikembangkan dapat digunakan oleh pelaku industri kreatif untuk mempermudah proses pencarian mitra kerja, manajemen proyek, serta transaksi pembayaran dalam satu ekosistem digital.

Bab 2
TINJAUAN LITERATUR

2.1 Penelitian Relevan

Berikut adalah lima penelitian terdahulu dalam lima tahun terakhir yang relevan dengan Perancangan Marketplace Digital untuk Freelancer dan Klien sebagai Solusi Ekonomi Kreatif :

1. Dhian, A. N., Dewayani, E., & Rusdi, Z. 2022. Perancangan Dan Pembuatan Program Aplikasi *Freelance* Berbasis *Web*. Metode yang digunakan dalam pengembangan aplikasi freelance menggunakan metode *SDLC* (*System Development Life Cycle*) dengan tipe pengembangan *Waterfall*. Konsep *SDLC* ini mendasari berbagai jenis model pengembangan perangkat lunak untuk membentuk suatu kerangka kerja untuk perencanaan dan pengendalian pembuatan sistem informasi. Salah satunya merupakan model *Waterfall*. Model *Waterfall* merupakan model yang linier, sehingga mudah diimplementasikan.(Dhian, A. N., Dewayani, E., & Rusdi, Z, 2022)
2. Masakazu, K. M., Ersi Sisdianto, Gede Suwardika, & Dhimas Setyo Nugroho. (2023). Peran Digital Freelancer Marketplace dan Media Sosial Terhadap Perkembangan Gig Economy Worker. Jurnal Pendidikan Ekonomi Undiksha, 15(1), 214–225. Jenis penelitian ini Observasional analitik dengan menggunakan rancangan penelitian case control atau kasus kontrol. Penelitian ini termasuk ke dalam jenis penelitian yang menggunakan pendekatan kualitatif. Pendekatan kualitatif digunakan apabila data yang hendak dikumpulkan dan disajikan dalam bentuk kata atau kalimat. Penelitian kualitatif sangat mengutamakan kualitas data, sehingga dalam penelitian kualitatif tidak digunakan analisis statistika. Masakazu, K. M., Ersi Sisdianto, Gede Suwardika, & Dhimas Setyo Nugroho. (2023). Adapun hasil pada penelitian menunjukan masih minimnya perhatian pemerintah karena tidak memberikan jaminan social dan perlindungan hukum serta untuk hasil perkembangan Gig Economy Worker sangat dipengaruhi oleh Digital Freelancer Marketplace dan media sosial.
3. Ar-raniry, U. I. N., Tarbiyah, F., Keguruan, D. A. N., Studi, P., & Teknologi, P. (2024). Perancangan *User Interface* Dan *User Experience Website Freelance Marketplace* Jasa Digital Menggunakan Metode *User Centered Design (Ucd). User Centered Design* menjadikan pengguna sebagai inti ketika melakukan perancangan program mulai dari pengembangan sistem, tujuan, tools dan prosedur pengerjaan berdasarkan pengalaman dan keinginan pengguna sehingga kunci dari kesuksesan metode ini adalah komunikasi antara pihak pengembang dengan calon pengguna. Ar-raniry, U. I. N., Tarbiyah, F., Keguruan, D. A. N., Studi, P., & Teknologi, P. (2024). Hasil pengujian usability

menunjukkan skor akhir sebesar 98,96, yang berarti prototipe tersebut dapat dikatakan excellent dan sesuai dengan kebutuhan pengguna.

4. Ade Sutedi, Dzikri Nursyaban, Cahya Mutiara, Dede Sopiah, Diaz Radhian Salam, Diva Nuratnika Rahayu, Hasfi Syahrul Ramadhan, Ilham Muhamad Ramdan, Intan Sri Fatmalasari, & Irsyad Ahmad. (2021). Program Pelatihan Web Development untuk Komunitas Maya. *Jurnal PkM MIFTEK*, *2*(1), 19–27. Kemajuan teknologi yang semakin pesat, lowongan pekerjaan dapat mudah ditemukan di mana saja. Sekarang ini, sudah banyak kaum muda yang bekerja sebagai freelancer terutama dalam bidang IT, seperti halnya menjadi Web Developer yang kini sedang menjadi tren dan banyak diminati. Novika (2021). Hasil yang dicapai yaitu dengan meningkatnya pemahaman dan kemampuan para peserta dalam membuat web sederhana.
5. *Triyas Niko Saputra & Imam Husni Al Amin. (2023).* Aplikasi manajemen tugas taskify untuk menentukan prioritas pekerjaan freelancer dengan ahp dan topsis. *Jurnal INSTEK (Informatika Sains dan Teknologi), 8(2), 292–301.* Hasil penelitian ini adalah suatu situs web sistem manajemen tugas bagi freelancer yang menggunakan metode *AHP* dan *TOPSIS.* Dalam sistem ini terdapat daftar kegiatan yang harus dilaksanakan oleh freelancer dengan beberapa kriteria yang ada pada setiap kegiatan, lalu dilakukan peringkat untuk merekomendasikan urutan tugas yang harus diprioritaskan, sehingga memudahkan freelancer dalam memilihnya. Hasil penelitian menunjukkan bahwa aplikasi Taskify dapat mengurangi masalah penilaian urutan pengerjaan tugas yang harus diselesaikan lebih awal yang sebelumnya hanya dilakukan dengan sistem perkiraan

2.1.1 Teori Perancangan Marketplace Digital

Marketplace merupakan suatu platform perdagangan elektronik yang menyediakan ruang bagi penjual dan pembeli untuk berinteraksi dan melakukan transaksi secara online. Dalam ekosistem marketplace, berbagai jenis produk dan layanan dapat ditemukan, mencakup segala hal mulai dari barang konsumen hingga jasa profesional. Soegiarto, R., & Susilowati, M. (n.d.). (2024)

Marketplace memainkan peran kunci dalam merampingkan proses perdagangan dengan menyatukan berbagai penawaran dalam satu tempat, memberikan kemudahan bagi konsumen untuk menemukan dan membandingkan produk atau layanan yang mereka butuhkan.

Namun, dibalik manfaatnya, marketplace juga membawa sejumlah tantangan. Persaingan yang ketat di antara penjual, biaya iklan untuk mendapatkan visibilitas yang optimal, serta kebijakan platform yang dapat mempengaruhi operasional bisnis, merupakan beberapa aspek yang perlu diperhatikan oleh penjual di dalam marketplace.

2.1.2 Teori Freelancer

Freelance atau pekerjaan lepas adalah bentuk pekerjaan yang memungkinkan seseorang bekerja secara independen tanpa terikat dengan satu perusahaan atau pemberi kerja tetap. Freelancer bekerja berdasarkan kontrak proyek dengan klien, yang bisa datang dari berbagai sektor atau industri. Pekerjaan ini telah berkembang pesat berkat kemajuan teknologi , terutama internet, yang memungkinkan pekerja untuk terhubung dengan klien di seluruh dunia dan bekerja dari lokasi manapun. Ilhamsyah, S., Fajri, M. H. F., & Defana, M. A. B. (2025).

Freelance adalah pengaturan kerja di mana seseorang melakukan pekerjaan secara mandiri, bukan secara penuh waktu untuk perusahaan tertentu. Pekerja lepas (freelance) dicirikan dengan tidak adanya ketaatan pada jadwal kerja yang kaku dan lokasi yang ditentukan, yang memberikan otonomi dan fleksibilitas yang cukup besar. Freelance biasanya bekerja di bawah kontrak yang disediakan klien atau proyek jangka pendek. Banyak dari freelancer menawarkan jasa atau keahlian mereka untuk mendapatkan sebuah proyek. Jasa yang mereka tawarkan dapat berupa jasa design, programmer, virtual assistant, social media manager, dan banyak lagi (Barlage et al., 2022).

2.1.3 Teori Ekonomi Kreatif

Ekonomi kreatif dapat didefinisikan sebagai sebuah konsep yang berkembang dari aset kreatif yang memiliki potensi untuk mendorong pertumbuhan dan perkembangan ekonomi. Ekonomi kreatif merupakan gabungan dari seluruh bagian industri kreatif, termasuk perdagangan, tenaga kerja, dan produksi, yang kemudian membentuk ekonomi kreatif. Inti dari industri kreatif meliputi periklanan, arsitektur, seni dan kerajinan, desain, fesyen, film, video, fotografi, musik, seni pertunjukan, penerbitan, penelitian & pengembangan, perangkat lunak, permainan komputer, penerbitan elektronik, dan TV/radio. Khoirunisa, Q., Mayana, R. F., & Muchtar, H. N. (2023).

Pembangunan ekonomi berkelanjutan berbasis kreativitas, seperti yang diungkapkan Howkins dalam bukunya "The Creative Economy, How People Make Money from Ideas" bahwa orang-orang yang memiliki ide akan menjadi lebih kuat dibanding mereka yang bekerja menggunakan mesin bahkan jika mereka memiliki mesin tersebut. Namun sayangnya hubungan antara kreativitas dan ekonomis tetap belum terlihat. Padahal sebuah hasil pemikiran telah melalui proses yang sulit bagi penciptanya sendiri baik dari segi pemikiran, tenaga maupun waktu hingga dapat memiliki nilai ekonomis.

2.1.4 Teori *Laravel*

Framework Laravel merupakan salah satu aplikasi *open source* yang didesain oleh Tylor Otwell. *Framework Laravel* merupakan *framework bundle*, salah satu *framework PHP* yang populer untuk membangunaplikasi web modern. Memiliki keunggulan *MVC Architecture*

(Model-View-Controller), routing yang Sederhana dan Fleksibel, *Eloquent ORM (Object-Relational Mapping),* keamanan yang tinggi dan memiliki *blade template engine* yang dapat memudahkan penggunanya. Raynaldi & Somya (2023).

Laravel merupakan *framework* yang dikembangkan oleh Taylor Otwell pada bulan Juni 2011 yang memiliki banyak pengguna hingga saat ini . Pada framework Laravel terdapat fungsi-fungsi kode yang disediakan di library kemudian di install ke dalam Laravel. Keuntungan umum menggunakan Laravel adalah penyebaran komunitas yang besar berdampak pada penemuan banyak library yang berbeda, beberapa peneliti menemukan bahwa library di Laravel dapat cukup besar sehingga mereka dapat menyelesaikan proyek pengembangan web dari skala rendah hingga menengah. Endra, R. Y., Aprilinda, Y., Dharmawan, Y. Y., & Ramadhan, W. (2021).

2.1.5 Teori *PostgreSQL*

PostgreSQL adalah sistem basis data objek-relasional open source yang kuat dengan lebih dari 30 tahun pengembangan aktif dan reputasi tinggi dalam performa dan kinerja. Aspek objek-relasional PostgreSQL meningkatkan model data relasional dengan dukungan untuk array, pewarisan, dan fungsi. PostgreSQL juga mendukung ekstensibilitas tipe data dan bahasa prosedural, dengan tabel dianggap sebagai kelas, dan baris serta kolom sebagai objek dan atributnya. Faruqi, M. H., Anshary, F. M. A., & Lathifah, S. N. (n.d.).(2025)

Penulis memilih PostgreSQL karena merupakan database objek relasional yang terpercaya dan stabil, memiliki tipe data geometri, menyediakan semua fitur database, kemampuan konektivitas dengan database lain, memiliki rute, mampu menampung data spasial, dan dapat diperluas dengan mudah. Kekurangan PostgreSQL adalah kurangnya popularitas dan belum tersedianya replikasi. (Wijayanti, ., and Saraswati 2023).

2.1.6 Teori Monolit

Arsitektur monolitik adalah pendekatan pengembangan perangkat lunakyang menggabungkan semua komponen sistem menjadi satu unit lengkap. Dalam arsitektur ini, database, lapisan logika bisnis, lapisan presentasi, dan lapisan antarmuka pengguna digabungkan menjadi satu unit terintegrasi. Ini berbeda dengan pendekatan arsitektur berbasis layanan dan arsitekturberorientasi mikro, yang mengambil pendekatan membagi sistem menjadi beberapa bagian terpisah. Sidiq, M. A. Z., Anshori, M. I., & Yaqin, R. A. (2024).

Berdasarkan hasil penerapan arsitektur monolitik tersebut pengembangan dilakukan dengan mudah karena hanya ada satu modul yang harus dikelola. lebih mudah untuk melakukan testing karena semua komponen aplikasi terintegrasi dalam satu modul, mudah untuk mengelola dependensi antar komponen karena semuanya terintegrasi dalam satu modul. Namun, untuk

aplikasi yang lebih besar dan kompleks, arsitektur monolitik kurang sesuai dan lebih baik menggunakan arsitektur terpisah atau arsitektur yang terpisah secara fisik (seperti arsitektur microservices). Iqbal, M., Syahputra, A. K., & Handoko, W. (2022).

2.1.7 Teori *User Acceptance Testing (UAT)*

User Acceptance Testing (UAT) atau Uji Penerimaan Pengguna adalah salah satu proses dari pengujian perangkat lunak dan merupakan tahapan dari pengembangan perangkat lunak. Pengujian ini akan melibatkan pengguna akhir yang akan menggunakan perangkat lunak yang dibuat. User atau pengguna akhir dari website akan memberikan feedback atau umpan balik setelah menggunakan websiteKanal Pengetahuan Dikti untuk melihat kepuasan user. Feedback tersebut bisa didapatkan dengan mengisi kuesioner. Kuesioner yang digunakan adalah kuesioner jenis PSSUQ (Post-Study System Quality Questionnaire). Responden yang ditargetkan pada penelitian ini adalah 80 responden. Puspita, M. A., Pradipta, R. A., & Muhammad, M. A. (2024).

UAT dapat juga dikatakan pengujian beta (beta testing), pengujian aplikasi (application testing), pengujian akhir pengguna akhir (end user testing) merupakan tingkatan pengembangan perangkat lunak ketika perangkat lunak diuji pada dunia nyata yang dimaksudkan oleh pengguna. UAT dapat dilakukan dengan in-house testing dengan membayar relawan atau subjek pengujian menggunakan perangkat lunak atau, biasanya mendistribusikan perangkat lunak secara luas dengan menggunakan pengujian versi yang tersedia gratis untuk diunduh melalui website. Pengalaman awal pengguna akan diteruskan kembali kepada para pengembang yang membuat perubahan sebelum akhirnya melepaskan perangkat lunak komersial. Dillah, A., Nama, G. F., Budiyanto, D., & Muhammad, M. A. (2024).

2.1.8 *State of The Art*

Berdasarkan studi literatur yang telah dikaji, dapat disimpulkan bahwa pengembangan platform digital untuk mempertemukan freelancer dan klien merupakan bidang yang terus berkembang seiring meningkatnya tren kerja fleksibel di era digital. Berbagai penelitian sebelumnya memberikan kontribusi penting terhadap pemahaman tentang bagaimana platform freelance sebaiknya dirancang dan dikembangkan.

Beberapa penelitian terdahulu yang telah dilakukan pada tabel 2.1 dan tabel 2.2 antara lain :

Tabel 2. 1 Perbandingan Tabel

No	Peneliti/Tahun	Fokus Penelitian	Kelebihan	Kekurangan
1.	Dhian et al. (2022)	Perancangan dan pembuatan sistem aplikasi freelance berbasis web	Proses pengembangan sistem berbasis SDLC jelas dan sistematis	Belum menyentuh aspek interaksi dan transaksi digital antara user
2.	Masakazu et al. (2023)	Pengaruh marketplace digital dan media sosial terhadap perkembangan gig economy worker	Analisis mendalam tentang pengaruh platform digital terhadap pekerja lepas	Tidak membahas perancangan atau implementasi sistem digital secara teknis
3.	Ar-raniry et al. (2024)	Perancangan UI/UX website freelance berbasis User Centered Design (UCD)	Fokus pada desain UI/UX berbasis kebutuhan pengguna dengan hasil excellent	Tidak mencakup sistem backend dan proses bisnis digital secara menyeluruh
4.	Ade Sutedi et al. (2021)	Pelatihan web development untuk komunitas freelance IT	Memberikan wawasan tren meningkatnya minat freelance di bidang IT	Hanya sebatas pelatihan, tidak menghadirkan solusi sistem yang konkrit
5.	Triyas & Imam (2023)	Aplikasi manajemen tugas freelancer dengan metode AHP dan TOPSIS	Menyediakan metode pengambilan keputusan berbasis data untuk pekerjaan	Tidak terintegrasi dengan sistem kolaborasi dan transaksi digital

Tabel 2. 2 Perbedaan Sistem yang Dikembangkan

Aspek	Penelitian Sebelumnya	Sistem KerjaBos(Pengembangan)
Arsitektur Sistem	Umumnya berbasis web sederhana, beberapa tanpa arsitektur yang terstruktur	Menggunakan arsitektur monolith berbasis Laravel 12 yang terintegrasi
Fokus Fungsional	Beberapa hanya fokus pada UI/UX, pelatihan, atau manajemen tugas	Mencakup seluruh proses kerja: posting proyek, apply, transaksi, notifikasi, manajemen user
Fitur Transaksi	Belum mengintegrasikan sistem pembayaran digital secara aman	Memiliki fitur connect, Wallet dan sistem transaksi (top up, bukti transfer, validasi admin)
Sistem Lamaran	Tidak terdapat pengaturan apply proyek secara efisien	Menggunakan sistem connect untuk mengatur jumlah pelamaran proyek
Dukungan Keamanan	Belum fokus pada otentikasi atau validasi transaksi	Menggunakan validasi connect, middleware role, serta verifikasi pembayaran
Tujuan Akhir Sistem	Fokus pada eksplorasi atau edukasi tanpa integrasi sistem kerja langsung	Mendorong kolaborasi online dan mendukung pertumbuhan ekonomi kreatif secara langsung melalui fitur terpadu
Validasi Sistem	Umumnya tidak dilakukan UAT atau pengujian pengguna akhir	Dilakukan User Acceptance Testing (UAT) untuk mengukur efektivitas dan penerimaan pengguna

Tabel diatas mempertegas bahwa sistem *KerjaBos* merupakan solusi yang lebih lengkap dan kontekstual terhadap kebutuhan nyata di bidang freelance dan ekonomi kreatif.

2.2 Kerangka Konsep

Kerangka konsep sistem *KerjaBos* merupakan gambaran menyeluruh mengenai alur hubungan antara pengguna, fitur utama dalam sistem, serta tujuan yang ingin dicapai melalui pengembangan platform ini. *KerjaBos* dirancang sebagai sebuah sistem berbasis web yang mempertemukan freelancer dan klien dalam satu wadah digital yang terintegrasi, dengan dukungan peran administratif dari pihak admin. Setiap pihak memiliki fungsi dan peran spesifik dalam proses kolaboratif yang berlangsung di dalam sistem.

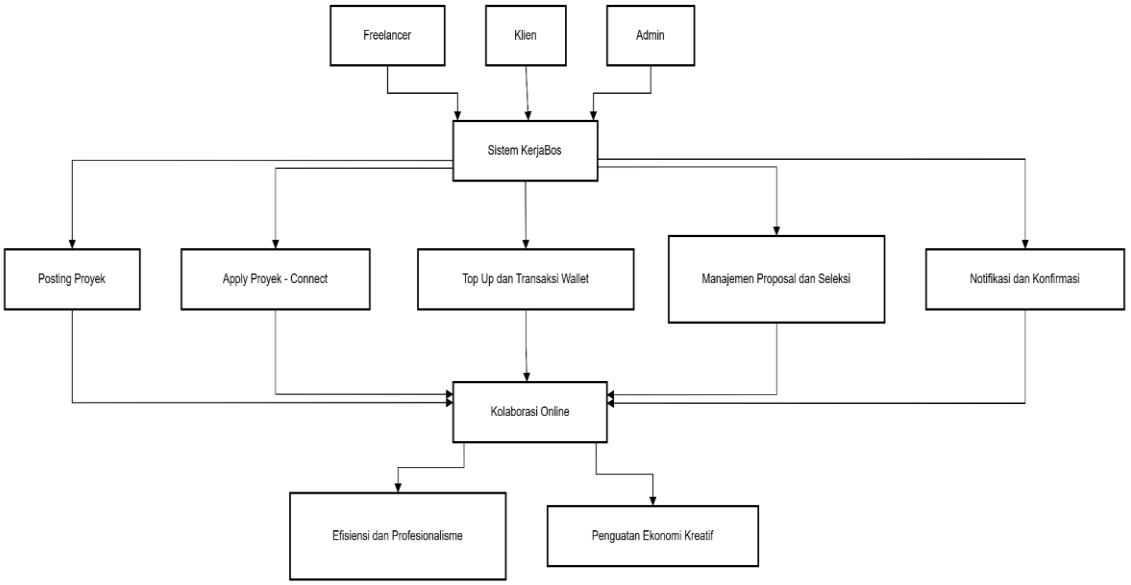
Freelancer berperan sebagai penyedia jasa yang dapat mencari dan melamar proyek sesuai keahlian mereka, sedangkan klien bertindak sebagai pihak yang mempublikasikan proyek dan mencari freelancer yang tepat. Admin, di sisi lain, berfungsi mengelola dan memverifikasi aktivitas dalam sistem, khususnya dalam aspek transaksi dan kontrol data.

Seluruh interaksi tersebut difasilitasi oleh sistem *KerjaBos* melalui serangkaian fitur inti. Fitur posting proyek memungkinkan klien mengunggah informasi pekerjaan yang dibutuhkan. Untuk dapat melamar proyek yang tersedia, freelancer menggunakan sistem connect yang dirancang sebagai bentuk seleksi terbatas agar proses lamaran lebih efisien dan terkontrol. Aktivitas keuangan antara klien dan freelancer dikelola melalui sistem wallet yang memungkinkan klien melakukan top-up saldo, serta mencatat riwayat transaksi secara aman dan transparan.

Dalam proses perekrutan, terdapat fitur manajemen proposal yang memungkinkan klien membaca, mengevaluasi, dan memilih proposal dari freelancer yang paling sesuai. Sementara itu, sistem notifikasi berperan penting dalam memberikan informasi real-time kepada pengguna atas setiap interaksi dan perubahan status dalam proyek yang sedang berlangsung, seperti penerimaan lamaran, persetujuan pembayaran, maupun konfirmasi penyelesaian.

Seluruh rangkaian aktivitas tersebut berpuncak pada terbentuknya kolaborasi online antara freelancer dan klien. Kolaborasi ini merupakan inti dari *KerjaBos*, yang menjadi sarana untuk meningkatkan efektivitas kerja berbasis proyek secara digital. Dari proses ini, dua hal utama yang diharapkan dapat dicapai adalah efisiensi serta peningkatan profesionalisme para pengguna, dan pada akhirnya mendorong kontribusi nyata terhadap pertumbuhan sektor ekonomi kreatif di era kerja fleksibel saat ini.

Dengan demikian, kerangka konsep ini menegaskan bahwa *KerjaBos* bukan hanya sebuah platform teknis, melainkan solusi sistematis yang menghubungkan kebutuhan, proses, dan tujuan dalam satu ekosistem kerja digital yang berkelanjutan. Adapun gambaran kerangka konsep dalam gambar 2.1 sebagai berikut :

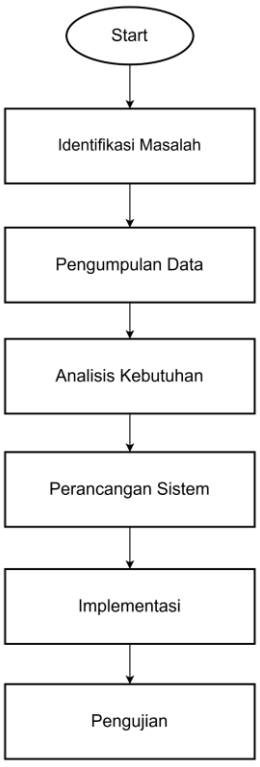


Gambar 2. 1 Kerangka Konsep Perancangan *Marketplace Digital* untuk *Freelancer* dan Klien sebagai Solusi Ekonomi Kreatif

Bab 3
METODE PENELITIAN

3.1 Prosedur Penelitian

Berdasarkan tinjauan pustaka yang telah dilakukan, maka platform KerjaBos dibuat sebagai sebuah aplikasi web yang dibangun secara monolith untuk mempertemukan freelancer dan klien dalam satu sistem terpadu. Platform ini menyediakan berbagai fitur utama, seperti posting proyek, pengajuan proposal, sistem koneksi antar pengguna, serta dompet digital yang mendukung transaksi aman dan transparan.. Dibangun dengan arsitektur monolith menggunakan Laravel dan PostgreSQL, KerjaBos bertujuan menjadi solusi digital yang efisien dalam mendukung kolaborasi kerja di sektor ekonomi kreatif. Tujuan penelitian ini adalah merancang dan membangun platform *KerjaBos* secara sistematis. Terdapat beberapa tahap prosedur pengembangan dari sistem tersebut, mulai dari analisis kebutuhan hingga tahap evaluasi akhir terhadap sistem yang telah dibangun. Adapun alur prosedur penelitian dapat digambarkan dalam bentuk flowchart pada gambar 3.1 sebagai berikut :



Gambar 3. 1 Alur Prosedur Penelitian

3.2 Identifikasi Permasalahan

Dalam pengembangan platform *KerjaBos*, terdapat beberapa permasalahan utama yang telah diidentifikasi. Masalah kepercayaan menjadi salah satu tantangan signifikan, mengingat kesulitan membangun kepercayaan antara freelancer dan klien dalam lingkungan digital. Risiko penipuan atau pengerjaan yang tidak sesuai dengan kesepakatan menjadi kekhawatiran bagi kedua belah pihak, terutama bagi klien yang memiliki kekhawatiran terhadap kualitas hasil pekerjaan.

Sistem pembayaran juga menjadi aspek yang menantang dalam pengembangan platform. Kesulitan dalam mengelola pembayaran yang aman dan transparan memerlukan perhatian khusus. Perlunya sistem *escrow* yang menjamin pembayaran hanya dilakukan setelah pekerjaan selesai dan disetujui menjadi solusi yang dipertimbangkan, namun implementasinya perlu dirancang dengan hati-hati. Manajemen *top-up wallet* dan *withdraw* juga memerlukan mekanisme verifikasi yang ketat untuk mencegah penyalahgunaan.

Sistem seleksi freelancer menjadi permasalahan lain yang perlu diatasi. Klien sering menghadapi kesulitan dalam memilih freelancer yang tepat dari banyaknya aplikasi yang masuk. Di sisi lain, freelancer juga mengalami kesulitan dalam menonjolkan keahlian mereka di tengah persaingan. Implementasi sistem *"Connect"* yang efektif diharapkan dapat mengatur aplikasi proyek secara lebih efisien dan berkualitas.

Komunikasi efektif antara klien dan freelancer juga menjadi tantangan yang perlu diatasi. Kebutuhan akan dokumentasi yang jelas tentang requirement proyek serta manajemen feedback dan revisi yang efisien menjadi aspek penting dalam memastikan kelancaran proses pengerjaan proyek. Sistem komunikasi yang terintegrasi dalam platform perlu dirancang dengan mempertimbangkan kebutuhan kedua belah pihak.

Dari segi teknis, pengembangan platform juga menghadapi beberapa tantangan, seperti kebutuhan akan sistem yang *scalable* untuk menangani pertumbuhan pengguna di masa depan. Keamanan data dan transaksi harus terjamin untuk membangun kepercayaan pengguna. Integrasi sistem yang kompleks antara manajemen proyek, pembayaran, dan komunikasi juga memerlukan perencanaan yang matang untuk memastikan semua komponen berfungsi secara harmonis.

3.3 Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk mengidentifikasi dan merumuskan kebutuhan sistem yang akan dikembangkan, baik dari sisi fungsionalitas maupun teknis. Tahapan ini penting untuk memastikan bahwa fitur-fitur yang dirancang pada platform KerjaBos sesuai dengan harapan dan kebutuhan pengguna, yakni freelancer, klien, dan admin.

Berdasarkan hasil studi literatur, observasi, dan wawancara dengan calon pengguna, diperoleh sejumlah kebutuhan utama yang harus dipenuhi oleh sistem. Kebutuhan tersebut diklasifikasikan menjadi dua bagian, yaitu kebutuhan fungsional dan kebutuhan non-fungsional.

1. **Kebutuhan Fungsional**

   Kebutuhan fungsional merupakan fitur-fitur utama yang harus dimiliki sistem agar dapat berjalan sesuai tujuan. Adapun kebutuhan fungsional pada sistem KerjaBos adalah sebagai berikut:

   a. Sistem dapat melakukan proses registrasi dan login sesuai peran (freelancer atau klien).
   b. Klien dapat membuat dan memposting proyek lengkap dengan kategori, tools, deskripsi, dan anggaran.
   c. Freelancer dapat melamar proyek dengan sistem pembatasan connect.
   d. Sistem dapat mencatat dan mengelola proposal dari freelancer ke proyek yang dipilih.
   e. Tersedia sistem dompet digital (wallet) untuk klien dan freelancer.
   f. Klien dapat melakukan top-up saldo, mengunggah bukti pembayaran, dan menunggu verifikasi dari admin.
   g. Admin dapat memverifikasi transaksi top-up, dan sistem akan menambahkan saldo ke akun klien.
   h. Sistem dapat memberikan notifikasi otomatis pada pengguna ketika ada aksi penting (misalnya saat ada proposal masuk atau status transaksi berubah).
   i. Admin dapat mengelola pengguna dan memantau aktivitas sistem.

2. **Kebutuhan Non Fungsional**

   Kebutuhan non-fungsional adalah syarat tambahan untuk mendukung kenyamanan dan keamanan penggunaan sistem. Pada platform KerjaBos, kebutuhan non-fungsional meliputi:

   a. Sistem dibangun dengan arsitektur monolith menggunakan framework Laravel 12 dan PostgreSQL sebagai database.
   b. Sistem harus memiliki tampilan antarmuka (UI) yang intuitif dan mudah digunakan.

c. Akses fitur dibatasi berdasarkan role pengguna, dan dilindungi dengan middleware.
d. Sistem harus menjaga keamanan data pengguna, termasuk enkripsi password dan validasi form.
e. Sistem harus mampu menangani aktivitas transaksi digital secara aman dan terstruktur.

Analisis kebutuhan ini menjadi dasar utama dalam proses perancangan sistem serta pengembangan modul-modul aplikasi agar sesuai dengan harapan pengguna akhir. Dengan memahami kebutuhan secara menyeluruh, sistem KerjaBos diharapkan dapat menjadi solusi yang efisien, aman, dan relevan dalam menjembatani kolaborasi antara freelancer dan klien secara digital.

3.4 Pengumpulan Data

Data dalam penelitian ini dikumpulkan dengan menggunakan tiga metode utama, yaitu:

1. Studi Literatur

   Studi literatur dilakukan untuk memperoleh landasan teoritis dan referensi empiris dalam merancang serta mengembangkan platform *KerjaBos*. Tahapan ini mencakup telaah terhadap penelitian terdahulu, teori-teori utama, dan penggunaan teknologi yang relevan dengan sistem marketplace digital untuk freelancer.

   Dari studi literatur ini, diperoleh pemahaman tentang:

   a. Pola kerja freelancer dan klien dalam dunia digital.
   b. Tantangan dan kebutuhan sistem dalam ekonomi kreatif.
   c. *Best practice* pengembangan platform berbasis *Laravel dan PostgreSQL*.
   d. Fitur-fitur penting seperti dompet digital, sistem connect, dan verifikasi pembayaran.

   Seluruh hasil studi ini menjadi dasar dalam menyusun kebutuhan sistem, fitur yang akan dikembangkan, serta pendekatan desain dan implementasi sistem *KerjaBos.*

2. Wawancara

   Metode wawancara digunakan untuk menggali kebutuhan, kendala, dan ekspektasi dari para calon pengguna sistem secara langsung. Wawancara dilakukan dengan beberapa freelancer aktif untuk memahami tantangan yang sering mereka alami ketika menggunakan platform digital, seperti kurangnya transparansi proyek, kesulitan mendapatkan klien, dan keterbatasan sistem komunikasi. Selain itu, wawancara juga dilakukan terhadap klien potensial, baik dari kalangan profesional maupun pemilik bisnis, dengan tujuan mengetahui harapan mereka terhadap fitur-fitur yang harus dimiliki oleh platform, seperti keamanan transaksi, kemudahan dalam mencari freelancer, serta kualitas hasil kerja. Penulis juga mewawancarai seorang praktisi industri TI guna memperoleh

wawasan mengenai tren teknologi, aspek keamanan sistem, serta cara membangun platform yang scalable dan user-friendly. Data yang diperoleh dari rangkaian wawancara ini memberikan insight langsung dari pengguna akhir dan menjadi acuan penting dalam menentukan prioritas fitur yang akan dikembangkan.

3. Observasi

   Observasi dilakukan untuk memahami secara nyata bagaimana pengguna berinteraksi dengan platform freelancer yang sudah ada, serta mengidentifikasi area yang masih bisa ditingkatkan.

   a. Peneliti mengamati perilaku pengguna saat menggunakan platform seperti *Upwork* dan *Fiverr*, termasuk bagaimana mereka menavigasi halaman, membuat akun, mengajukan proposal, dan mengelola transaksi.
   b. Selama observasi, dicatat pain points atau kendala yang muncul, misalnya proses verifikasi yang rumit, tidak adanya sistem pengelolaan proyek yang jelas, atau komunikasi yang terbatas antara klien dan freelancer.
   c. Selain itu, dilakukan pemetaan terhadap alur kerja dan proses bisnis umum dalam industri freelance, seperti proses perekrutan, negosiasi, pengerjaan proyek, hingga pembayaran, yang dijadikan sebagai referensi dalam membangun sistem yang relevan dan efisien.

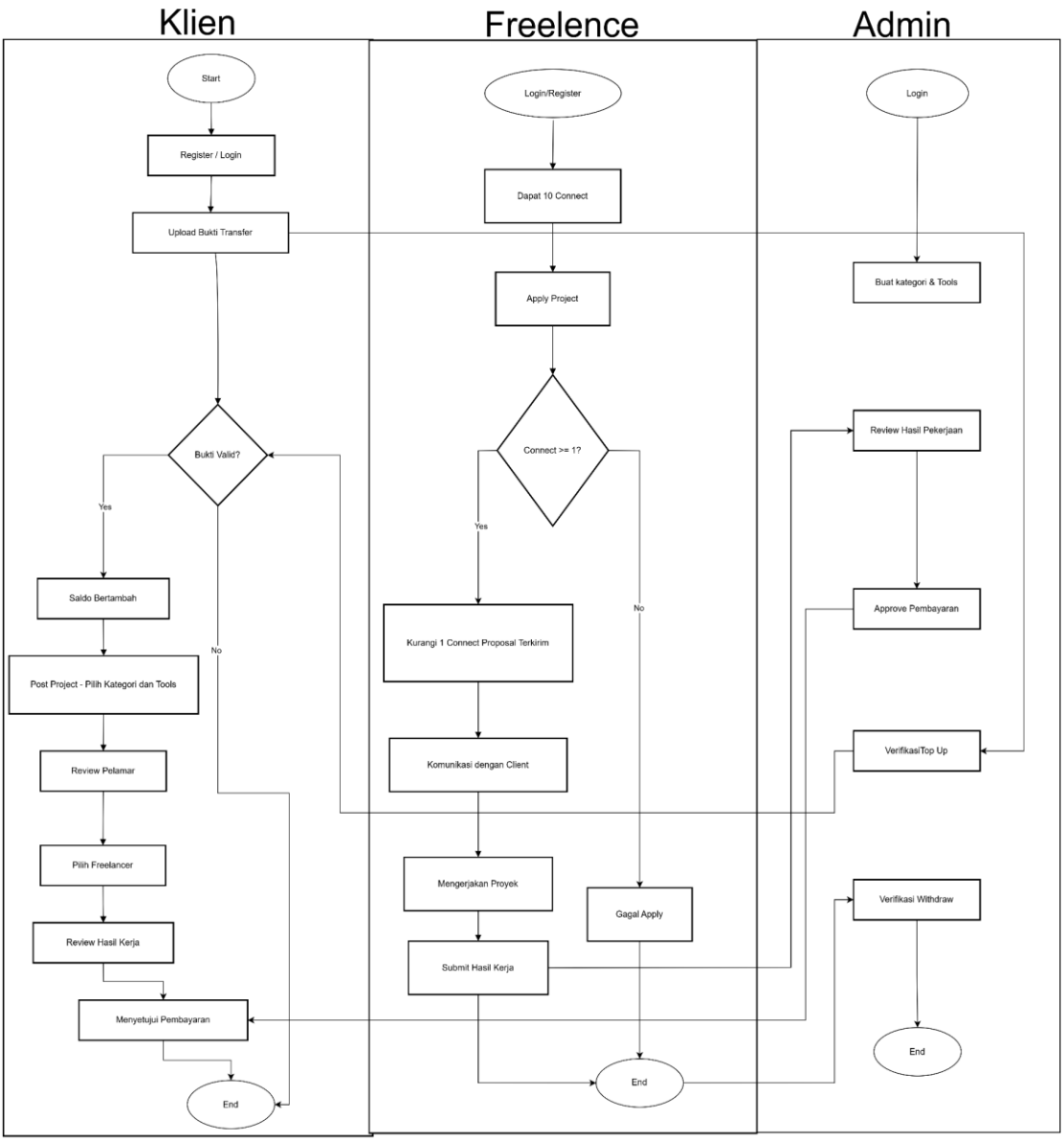
3.5 Perancangan Sistem

Perancangan bertujuan untuk menjelaskan struktur sistem dan bagaimana komponen-komponen utamanya saling terhubung. Dalam metafora *monolith*, semua modul, yaitu coba-coba, transaksi, signing, manajemen user, dan tidak dipisahkan menjadi aplikasi mandiri. Mereka disatukan dalam satu kesatuan aplikasi *Laravel*, yang menjadi gerbang antara pengguna dan database.

Perancangan sistem terdiri dari 4 poin utama diantaranya:

A. Arsitektur sistem

Arsitektur sistem berbasis *monolith*, di mana seluruh fungsi *frontend dan backend* berada dalam satu struktur aplikasi *Laravel*. Semua logika bisnis dan tampilan dikemas dalam satu repositori dan dijalankan melalui satu server aplikasi. Adapun flowchart yang menggambarkan arsitektur sistem pada gambar 3.2 sebagai berikut :



Gambar 3. 2 Arsitektur Sistem

B. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) yang digunakan untuk memodelkan struktur data dan hubungan antar entitas seperti *users*, *projects*, *applicants*, *tools*, *wallet*, dan *transaction*. ERD membantu memastikan desain database yang efisien dan mendukung kebutuhan fungsional sistem. Adapun entitas yang dirancang dalam sistem KerjaBos ini terdapat pada tabel 3.1 sebagai berikut :

Tabel 3. 1 Entity Relationship Diagram

Nama Entitas	Deskripsi	Atribut Utama	Relasi
Users	Menyimpan data pengguna, baik klien maupun freelancer.	id, name, email, password, avatar, occupation, connect	Wallets (1:1), Projects (1:M), Project_Applicants (freelancer_id)
Wallets	Menyimpan saldo dompet digital untuk masing-masing user.	id, user_id, balance	Users (1:1), Wallet_Transaction (1:M)
Wallet_ Transaction	Mencatat riwayat transaksi top up, pembayaran proyek, atau withdraw saldo.	id, user_id, amount, type, is_paid, proof, bank_name, bank_account_name, bank_account_number	Wallets (M:1 lewat user_id)
Projects	Menyimpan informasi proyek yang diposting oleh klien.	id, name, slug, thumbnail, about, category_id, client_id, budget, skill_level, has_finished, has_started	Users (client_id), Categories (category_id), Project_Applicants (1:M), Project_Tool (1:M)
Categories	Menyimpan daftar kategori proyek.	id, name, slug, icon	Projects (1:M)
Tools	Menyimpan daftar alat atau teknologi yang digunakan dalam proyek.	id, name, slug, icon	Project_Tool (1:M)

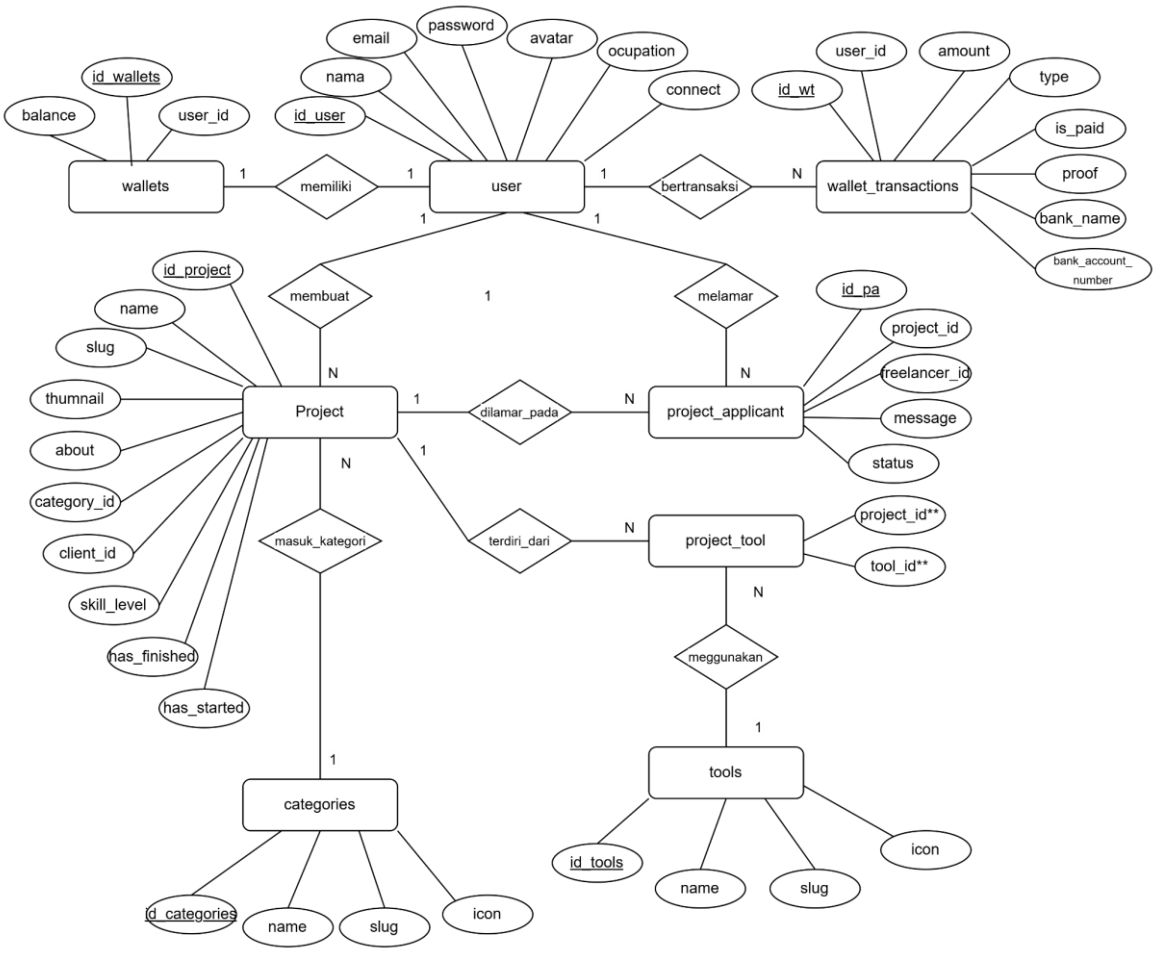
Project_Tool	Menyimpan relasi many-to-many antara proyek dan tools.	id, project_id, tool_id	Projects (M:1), Tools (M:1)
Project_Applicants	Menyimpan data pelamar proyek (freelancer yang apply ke proyek).	id, project_id, freelancer_id, message, status	Projects (M:1), Users (freelancer_id)

Penjelasan Tambahan Berdasarkan Note di ERD

a. Freelancer wajib memiliki minimal 1 connect agar bisa apply ke proyek (terhubung dengan field connect di tabel Users).
b. Klien wajib memiliki saldo di Wallet sebelum bisa memposting proyek baru (terhubung dengan Wallets dan Wallet_Transaction).
c. Role dan permission diatur menggunakan package Spatie dan tidak dihandle langsung di ERD utama ini (ditangani lewat Laravel permission management).

Relasi antar entitas dalam sistem *KerjaBos* dibangun dengan memperhatikan dependensi logis dan kebutuhan operasional dalam pengelolaan proyek freelance. Sebagai contoh, Projects memiliki relasi dengan users sebagai klien yang memposting proyek, *Project_Applicants* terhubung dengan users sebagai freelancer yang mengajukan lamaran proyek, dan *Wallet_ Transactions* mencatat seluruh aktivitas transaksi saldo pengguna.

Struktur basis data ini dirancang untuk memastikan integritas data, kemudahan dalam pengelolaan informasi pengguna dan proyek, serta mendukung kelancaran proses transaksi secara aman dan transparan. Hubungan antar entitas juga dirancang untuk mendukung alur kerja, mulai dari pendaftaran pengguna, posting proyek, pengajuan proposal, hingga pembayaran dan distribusi dana. Diagram Entity Relationship Diagram (ERD) sistem secara keseluruhan disajikan pada gambar 3.3 sebagai berikut :



Gambar 3. 3 *ERD KerjaBos*

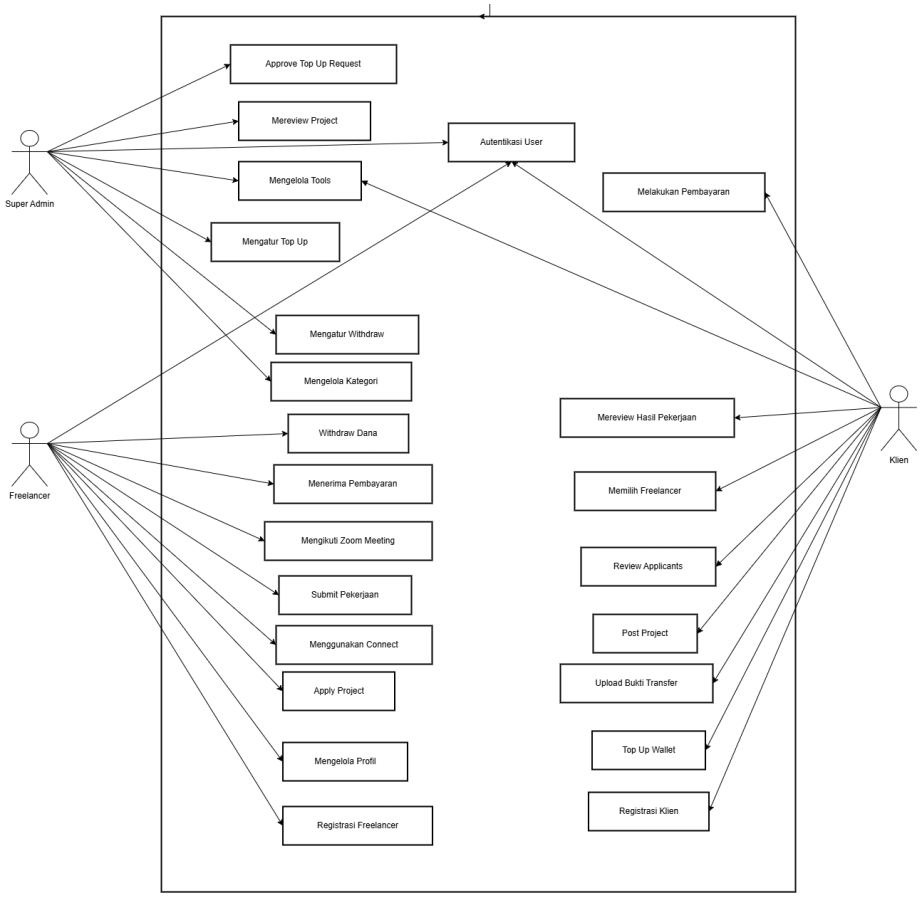
C. Diagram UML

Berbagai jenis diagram dari Unified Modeling Language (UML) digunakan untuk menggambarkan interaksi antara proses bisnis dan pengguna dan sistem kerja. Diagram ini membantu Anda memvisualisasikan dengan jelas persyaratan fungsional sistem, alur kerja utama, dan hubungan antara komponen yang mendukung proses operasional pada platform.

Jenis diagram UML yang digunakan dalam pengembangan sistem ini adalah:

1. Use Case Diagram

Use Case Diagram yang menggambarkan interaksi antara sistem dengan aktor (Admin, Freelancer, Klien). Diagram ini memetakan fitur-fitur utama dari sudut pandang pengguna dan memberikan gambaran menyeluruh tentang fungsionalitas sistem. Visualisasi dari skenario interaksi antara aktor dan sistem dapat dilihat pada gambar 3.4 sebagai berikut :



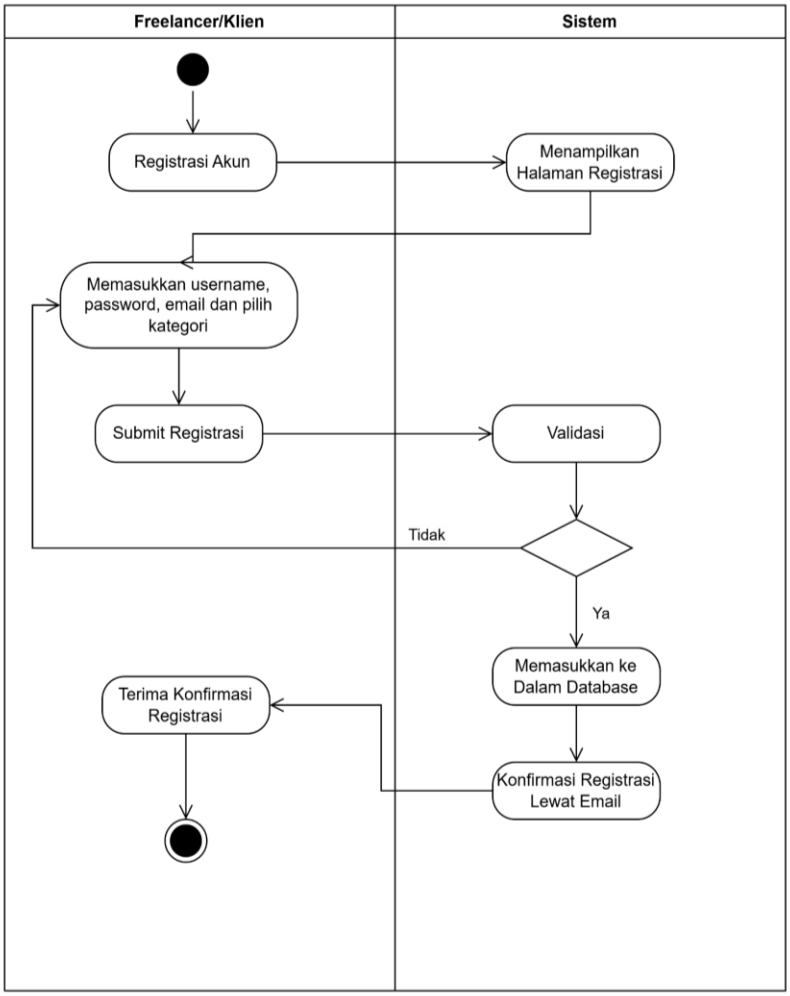
Gambar 3. 4 *UseCase* Diagram

2. Activity Diagram

Activity Diagram digunakan untuk menjelaskan alur proses bisnis utama secara terstruktur, seperti proses registrasi, top up saldo, posting proyek, hingga freelancer menerima pembayaran.

1. Proses Registrasi

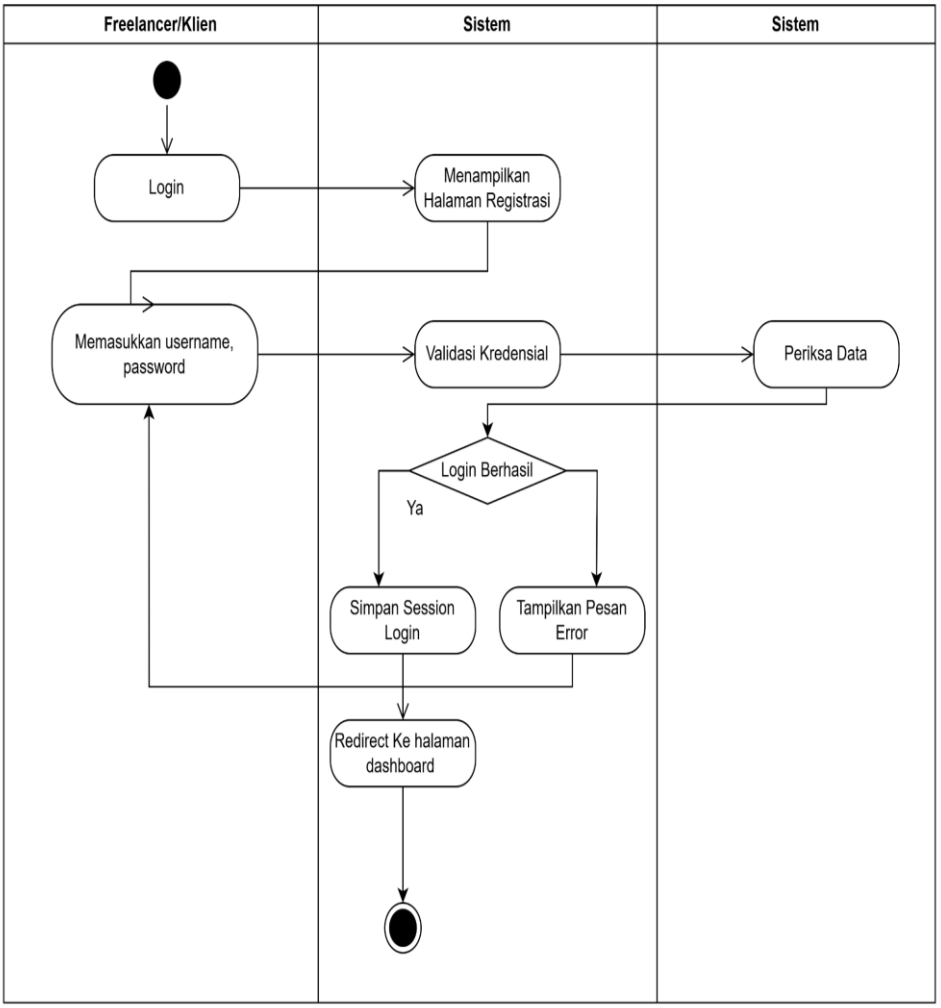
Activity diagram ini menggambarkan alur registrasi akun di platform *KerjaBos*. Pengguna mengisi data registrasi, kemudian sistem melakukan validasi. Jika validasi berhasil, data disimpan ke database dan sistem mengirimkan email konfirmasi. Proses registrasi selesai setelah pengguna menerima konfirmasi. Adapun proses registrasi dapat digambarkan pada gambar 3.5 sebagai berikut :



Gambar 3. 5 *Diagram Activity Registrasi*

2. Proses Login

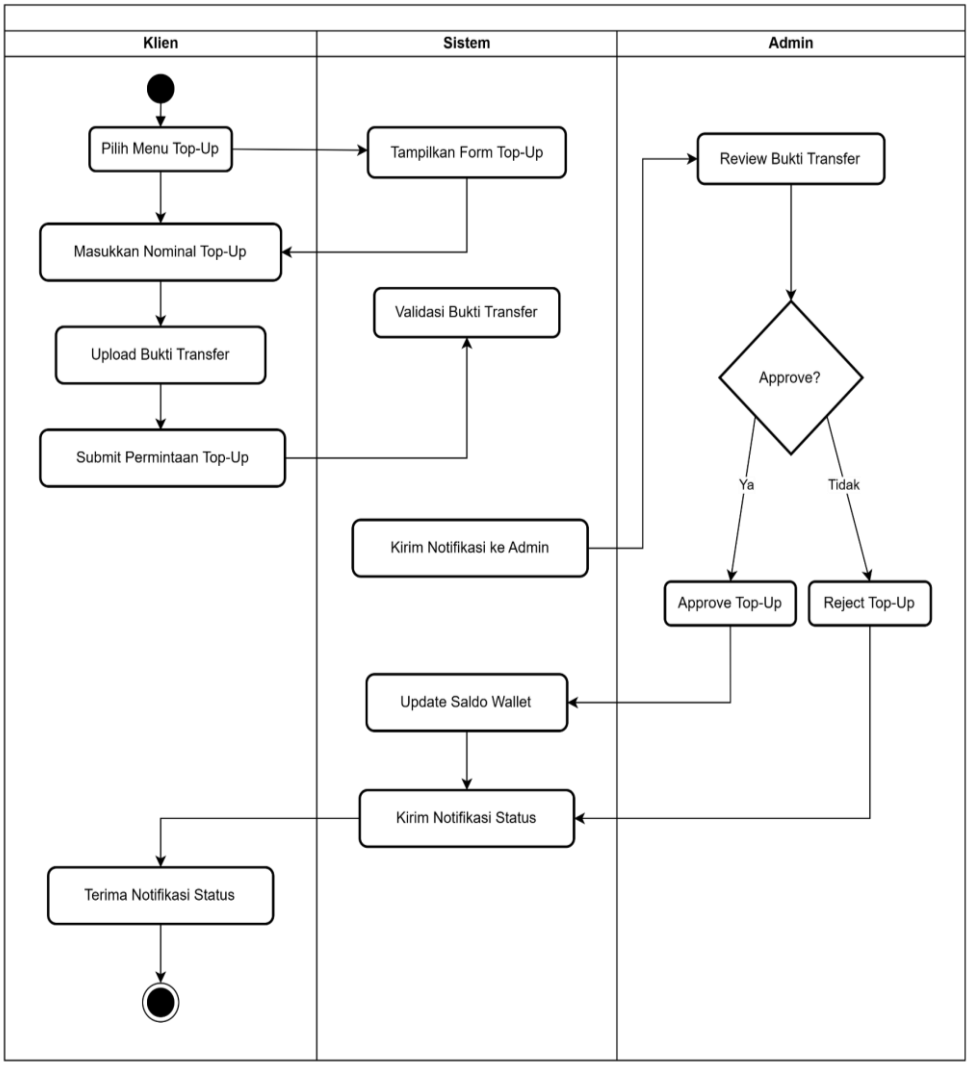
Activity diagram ini menunjukkan alur login pengguna di platform KerjaBos. Pengguna memasukkan username dan password, kemudian sistem memvalidasi data tersebut. Jika valid, sistem menyimpan sesi login dan mengarahkan pengguna ke dashboard. Jika tidak valid, sistem menampilkan pesan error. Proses login dapat digambarkan pada gambar 3.6 sebagai berikut :



Gambar 3. 6 Diagram *Activity Login*

3. Proses Top Up Wallet

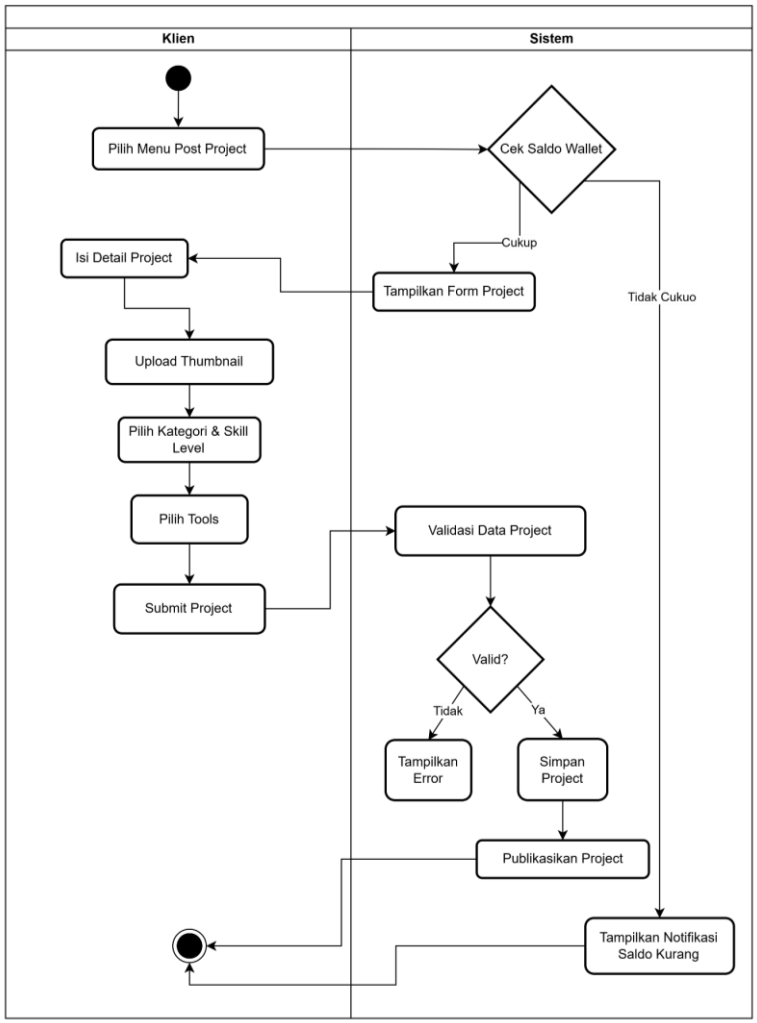
Activity diagram ini menggambarkan alur proses top up saldo wallet pada platform *KerjaBos*. Klien memilih menu top up, mengisi nominal, dan mengunggah bukti transfer. Setelah submit, sistem memvalidasi bukti dan mengirim notifikasi ke admin. Admin kemudian meninjau bukti transfer, dan memutuskan untuk menyetujui atau menolak top up. Jika disetujui, sistem memperbarui saldo wallet klien dan mengirimkan notifikasi status top up. Proses *top up wallet* dapat digambarkan pada gambar 3.7 sebagai berikut :



Gambar 3. 7 Diagram *Activity Top Up Wallet*

4. Proses Posting Project

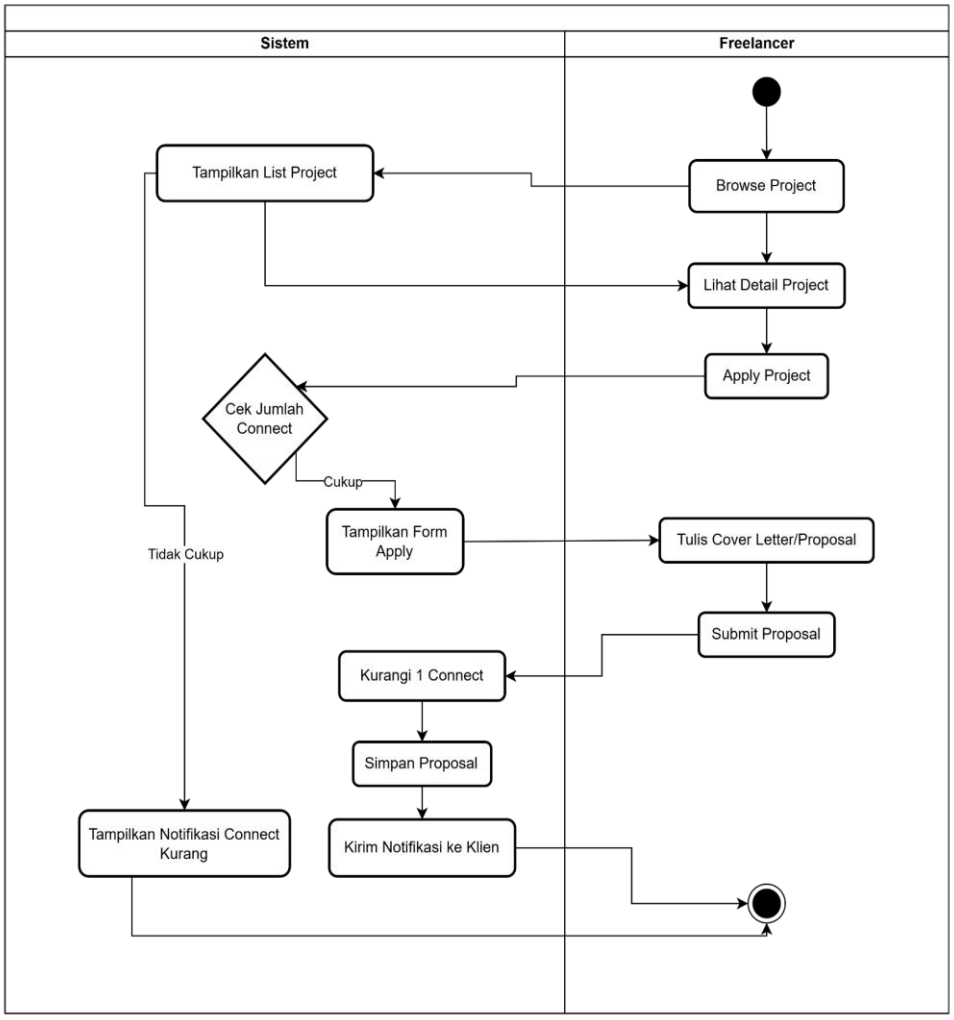
Activity diagram ini menggambarkan alur proses posting proyek pada platform *KerjaBos*. Klien memilih menu post project, lalu sistem memeriksa saldo wallet. Jika saldo mencukupi, klien mengisi detail proyek, mengunggah thumbnail, memilih kategori, skill level, dan tools, kemudian mengirimkan proyek. Sistem memvalidasi data, dan jika valid, proyek disimpan serta dipublikasikan. Jika saldo tidak mencukupi atau data tidak valid, sistem menampilkan notifikasi atau pesan error. Proses posting project dapat digambarkan pada gambar 3.8 sebagai berikut :



Gambar 3. 8 Diagram *Activity Posting* Proyek

5. Proses Apply Project

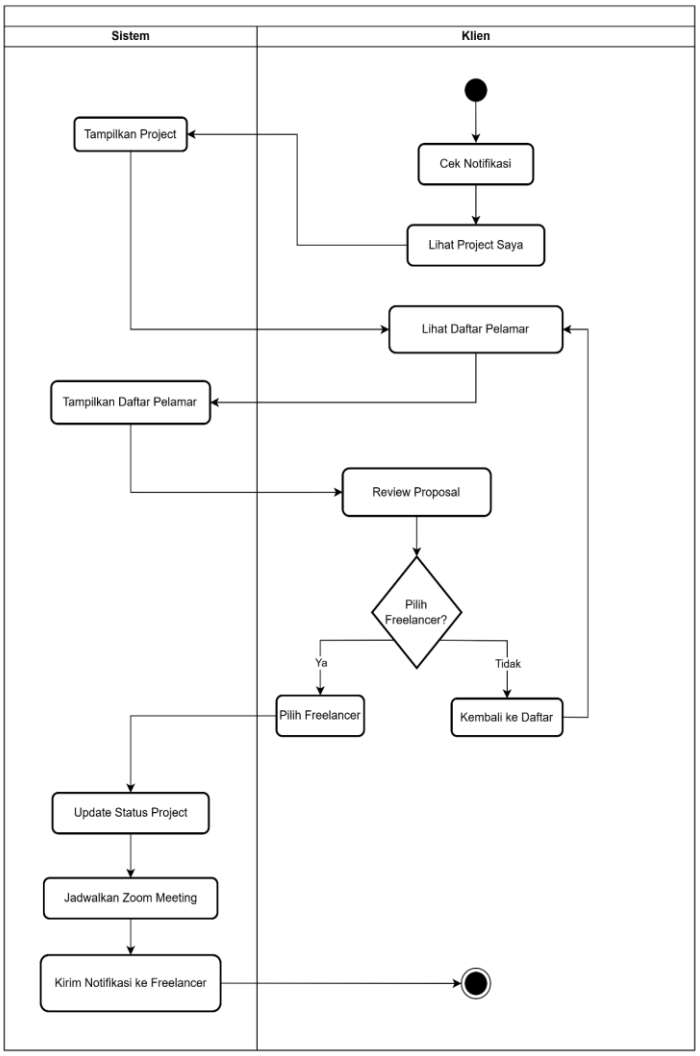
Activity diagram diatas menggambarkan alur proses *apply* proyek oleh freelancer pada platform *KerjaBos*. Freelancer melakukan browsing dan melihat detail proyek, lalu memilih untuk *apply*. Sistem memeriksa jumlah connect freelancer. Jika cukup, freelancer mengisi form apply, submit proposal, dan sistem mengurangi satu connect serta menyimpan proposal. Notifikasi kemudian dikirimkan ke klien. Jika connect tidak mencukupi, sistem menampilkan notifikasi kekurangan connect. Proses *apply* project dapat digambarkan pada gambar 3.9 sebagai berikut :



Gambar 3. 9 Diagram *Activity Apply* Proyek

6. Proses Review dan pemilihan Freelancer

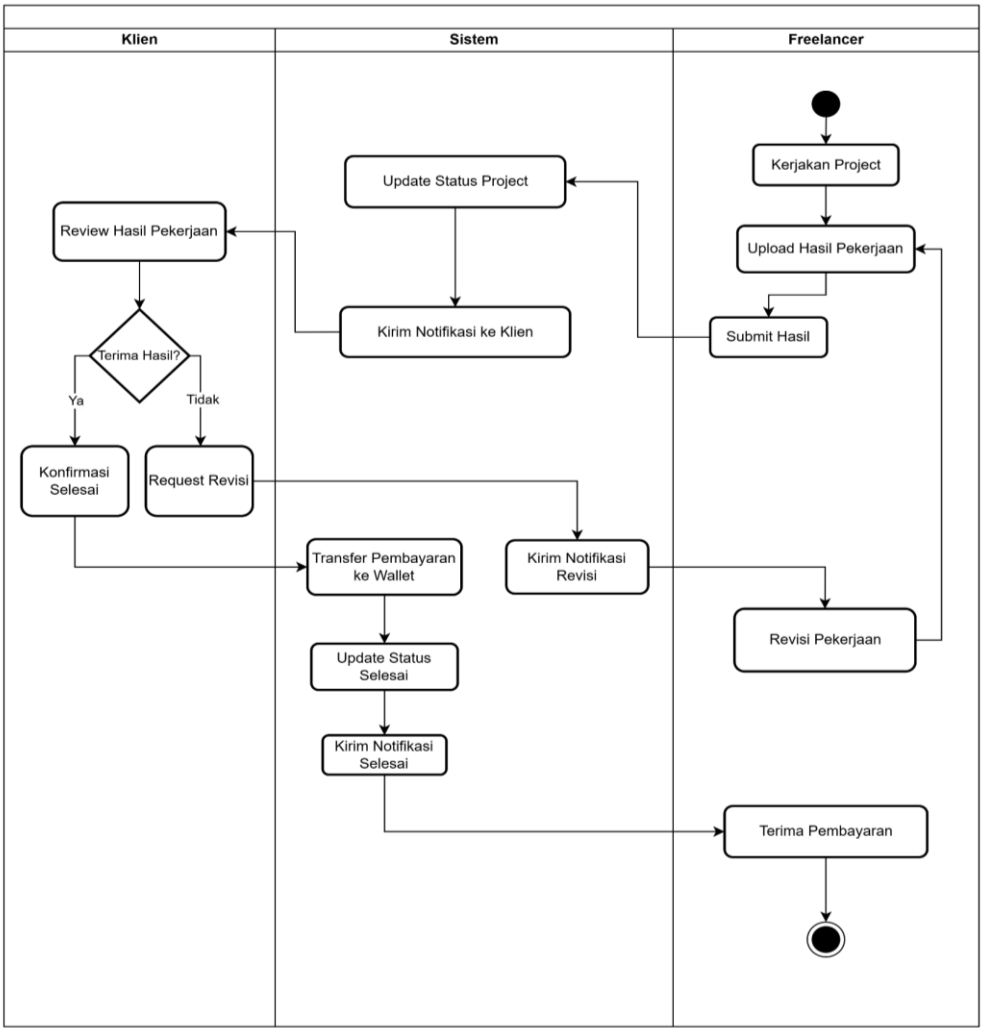
Activity diagram ini menjelaskan alur proses saat klien melakukan review dan pemilihan freelancer pada platform *KerjaBos*. Klien memulai dengan mengecek notifikasi, melihat proyek yang dimiliki, lalu membuka daftar pelamar. Sistem akan menampilkan semua pelamar yang tersedia, kemudian klien mereview proposal yang diajukan. Jika klien memilih freelancer, sistem akan memperbarui status proyek, menjadwalkan meeting, dan mengirimkan notifikasi ke freelancer. Jika tidak memilih, klien akan kembali ke daftar pelamar. Proses review dan pemilihan freelancer dapat digambarkan pada gambar 3.10 sebagai berikut :



Gambar 3. 10 Diagram *Activity Review* dan Pemilihan

7. Proses Submit Pekerjaan dan Pembayaran

Activity diagram ini menjelaskan alur saat freelancer mengerjakan proyek dan menyerahkan hasilnya kepada klien di platform *KerjaBos*. Freelancer mulai dengan mengerjakan proyek lalu mengupload dan submit hasil kerja. Sistem memperbarui status proyek dan mengirim notifikasi ke klien. Klien kemudian mereview hasil pekerjaan; jika hasil diterima, klien mengkonfirmasi penyelesaian. Jika tidak, klien minta revisi dan sistem mengirimkan notifikasi revisi kepada freelancer. Freelancer melakukan revisi dan mengirimkan ulang hasilnya. Setelah pekerjaan diterima, sistem mentransfer pembayaran ke wallet freelancer dan mengirimkan notifikasi penyelesaian. Proses submit pekerjaan dan pembayaran dapat digambarkan pada gambar 3.11 sebagai berikut :



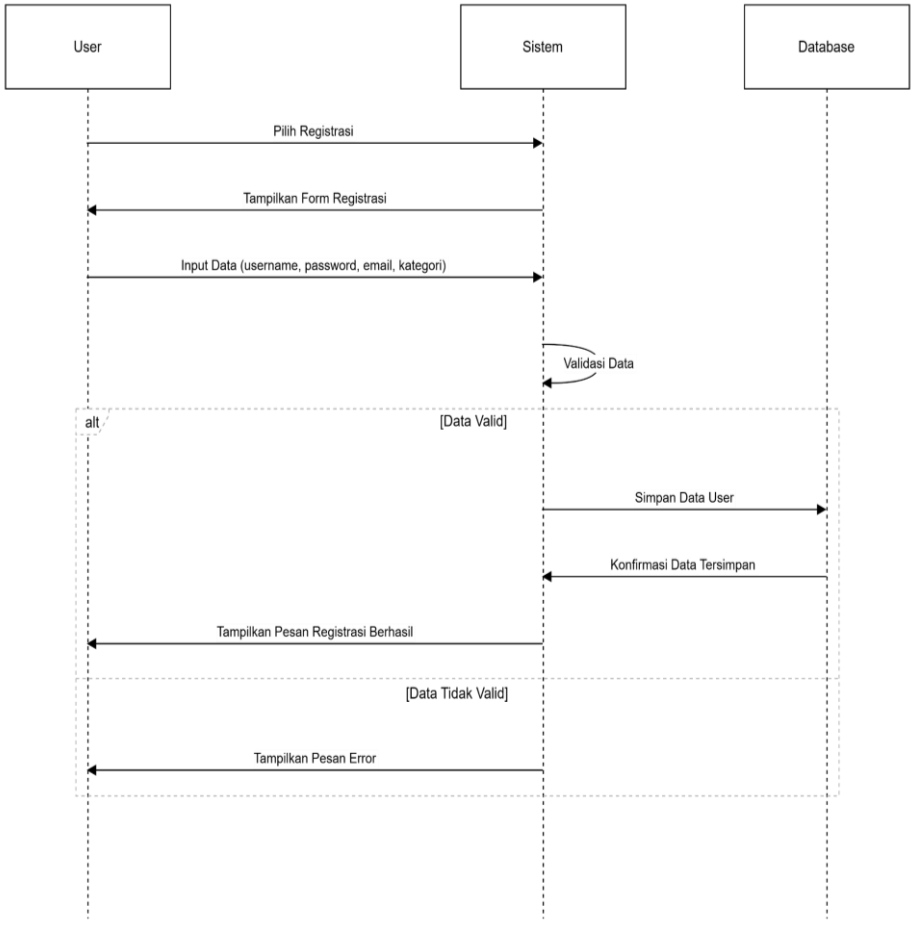
Gambar 3. 11 Diagram *Activity Submit* Pekerjaan dan Pembayaran

3. Sequence Diagram

Sequence Diagram yang menunjukkan urutan interaksi antar objek (Klien, Freelancer, Admin, dan Platform) dalam satu skenario tertentu. Diagram ini membantu menggambarkan bagaimana data dan perintah mengalir dalam sistem, misalnya saat proses klien memposting proyek, freelancer mengirim proposal, hingga transaksi selesai dilakukan.

1. Proses Registrasi

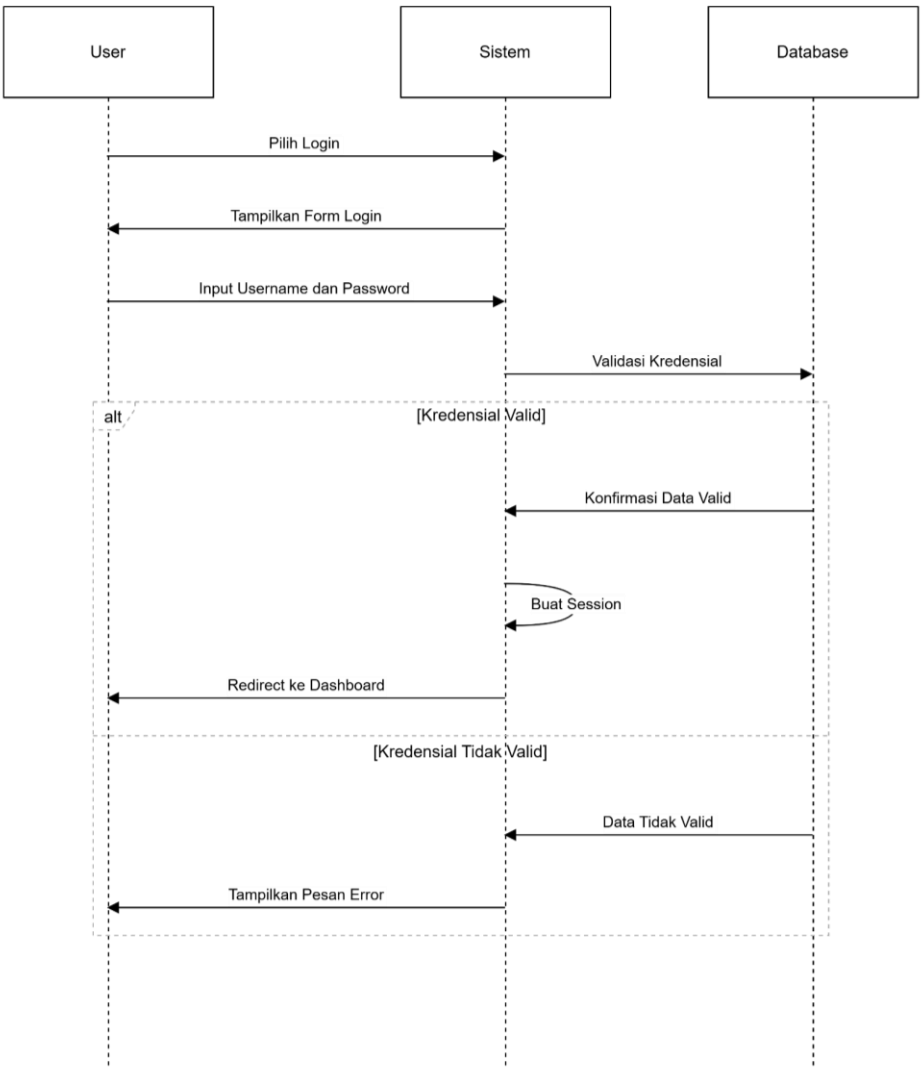
Sequence diagram ini menjelaskan alur proses registrasi pengguna pada platform KerjaBos. Pengguna memulai dengan memilih opsi registrasi, lalu sistem menampilkan form registrasi. Pengguna mengisi data seperti username, password, email, dan kategori, kemudian sistem melakukan validasi data. Jika data valid, sistem menyimpan data ke database dan mengirimkan konfirmasi bahwa data telah tersimpan, lalu menampilkan pesan registrasi berhasil kepada pengguna. Jika data tidak valid, sistem menampilkan pesan error. Proses registrasi dapat digambarkan pada gambar 3.12 sebagai berikut :



Gambar 3. 12 *Sequence* Diagram *Register*

2. Proses Login

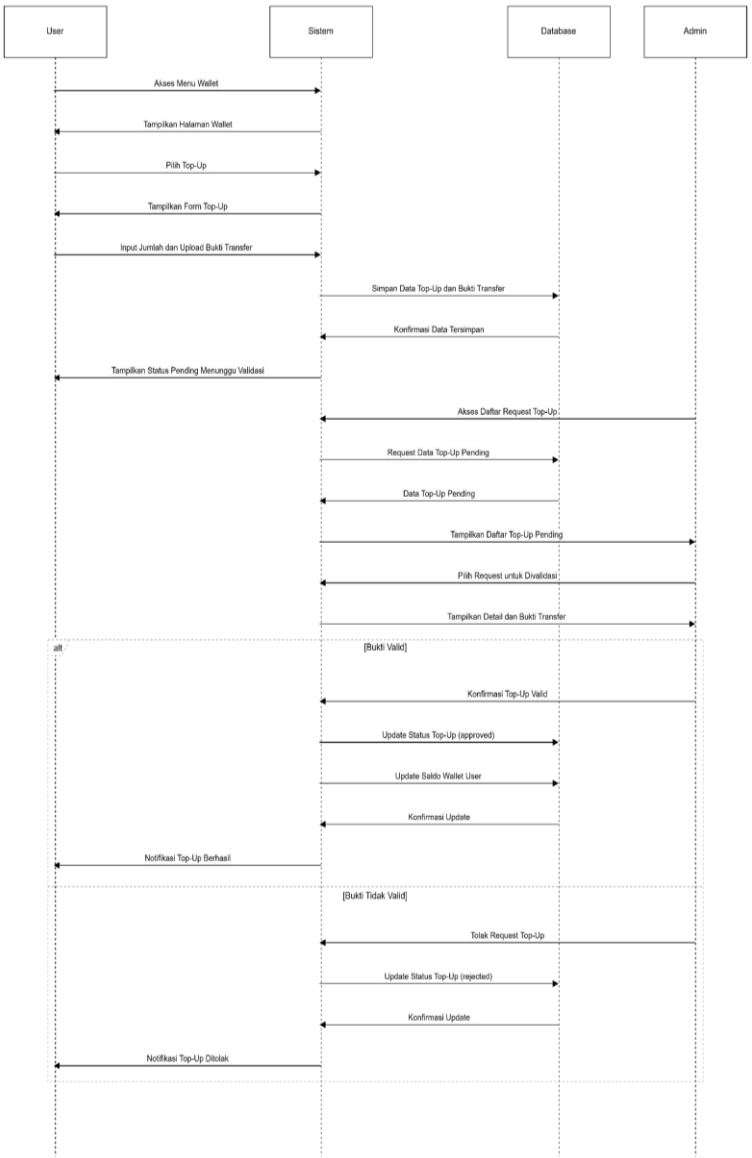
Sequence diagram ini menggambarkan proses login pengguna pada platform *KerjaBos*. Pengguna memulai dengan memilih menu login, kemudian sistem menampilkan form untuk memasukkan username dan password. Setelah pengguna menginput data, sistem memvalidasi kredensial ke database. Jika kredensial valid, sistem mengonfirmasi data, membuat sesi login, dan mengarahkan pengguna ke dashboard. Jika kredensial tidak valid, sistem memberikan notifikasi bahwa data tidak valid dan menampilkan pesan error kepada pengguna. Proses login dapat digambarkan pada gambar 3.13 sebagai berikut :



Gambar 3. 13 *Sequence* Diagram *Login*

3. Proses Top Up

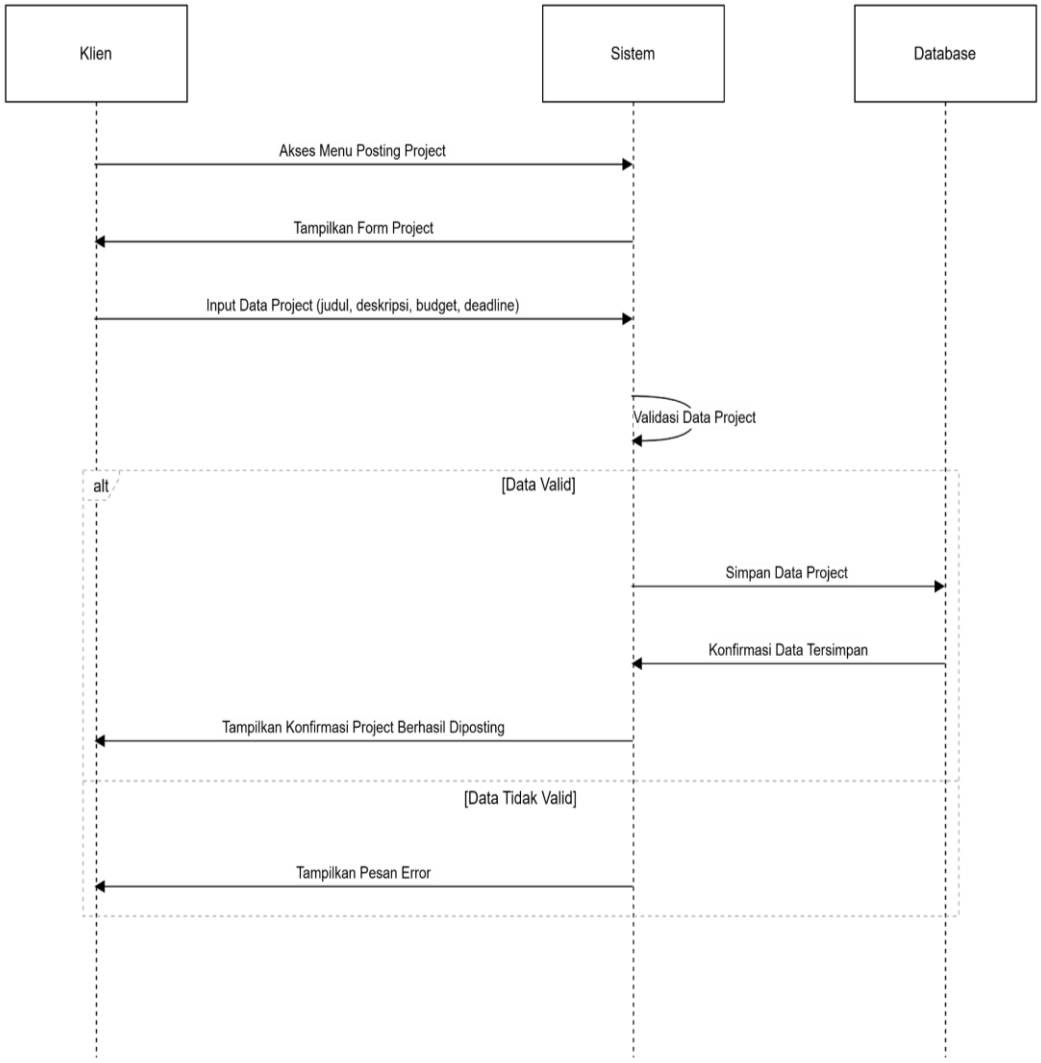
Sequence diagram ini menjelaskan proses top up saldo wallet pengguna pada platform *KerjaBos.* Pengguna mengakses menu wallet, memilih opsi top up, lalu sistem menampilkan form top up. Setelah pengguna mengisi jumlah saldo dan mengunggah bukti transfer, sistem menyimpan data dan menampilkan status pending. Admin kemudian memvalidasi bukti transfer; jika valid, sistem menambahkan saldo ke wallet pengguna dan mengirim notifikasi top up berhasil. Jika bukti transfer tidak valid, admin menolak permintaan dan sistem mengirim notifikasi bahwa top up gagal. Proses top up dapat digambarkan pada gambar 3.14 sebagai berikut :



Gambar 3. 14 *Sequence* Diagram *Top Up*

4. Proses Posting Project

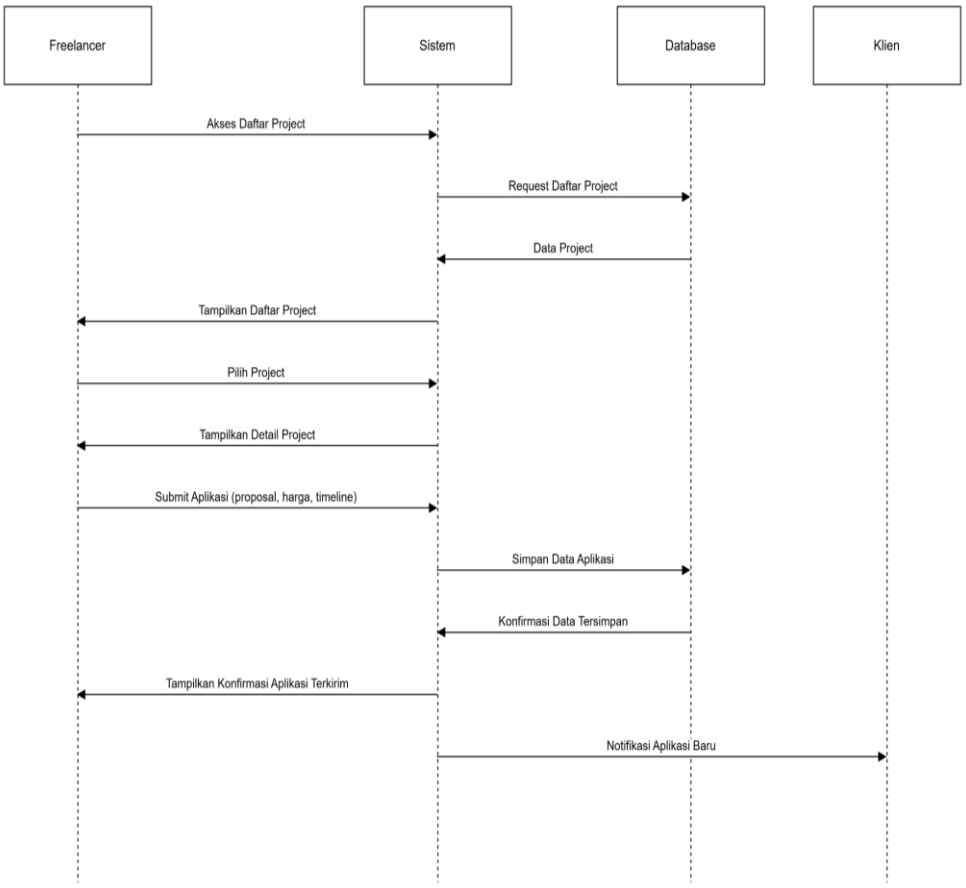
Sequence diagram ini menjelaskan proses saat klien memposting proyek di platform *KerjaBos.* Klien mengakses menu posting proyek, lalu sistem menampilkan form pengisian data proyek seperti judul, deskripsi, budget, dan deadline. Setelah data diinput, sistem melakukan validasi; jika data valid, sistem menyimpan data proyek ke database dan memberikan konfirmasi bahwa proyek berhasil diposting. Jika data tidak valid, sistem akan menampilkan pesan error kepada klien.



Gambar 3. 15 *Sequence* Diagram *Posting* Proyek

5. Proses Apply Project

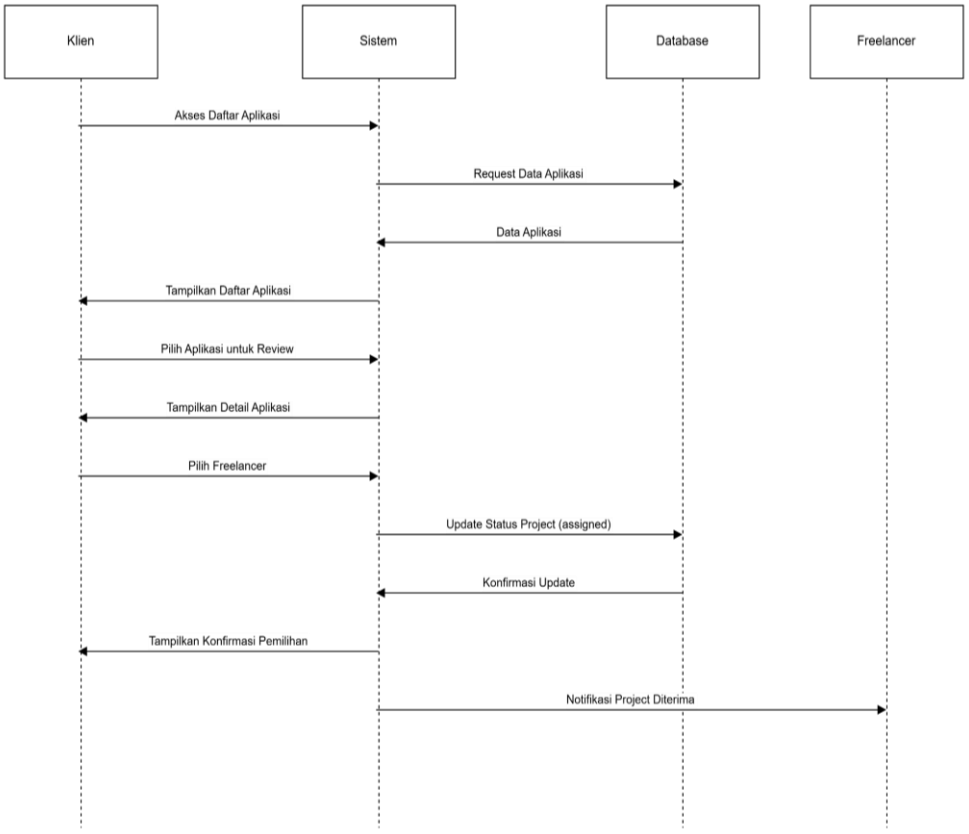
Sequence diagram ini menggambarkan alur saat freelancer melakukan apply pada proyek di platform *KerjaBos*. Freelancer mengakses daftar proyek, lalu sistem mengirimkan permintaan data ke database dan menampilkan daftar proyek kepada freelancer. Setelah memilih proyek, freelancer melihat detail proyek dan mengirimkan aplikasi berupa proposal, harga, dan timeline. Sistem menyimpan data aplikasi ke database dan memberikan konfirmasi bahwa aplikasi berhasil, kemudian sistem juga mengirimkan notifikasi kepada klien tentang adanya aplikasi baru. Proses apply project dapat digambarkan pada gambar 3.16 sebagai berikut :



Gambar 3. 16 *Sequence* Diagram *Apply* Proyek

6. Proses Review dan Pilih Freelancer

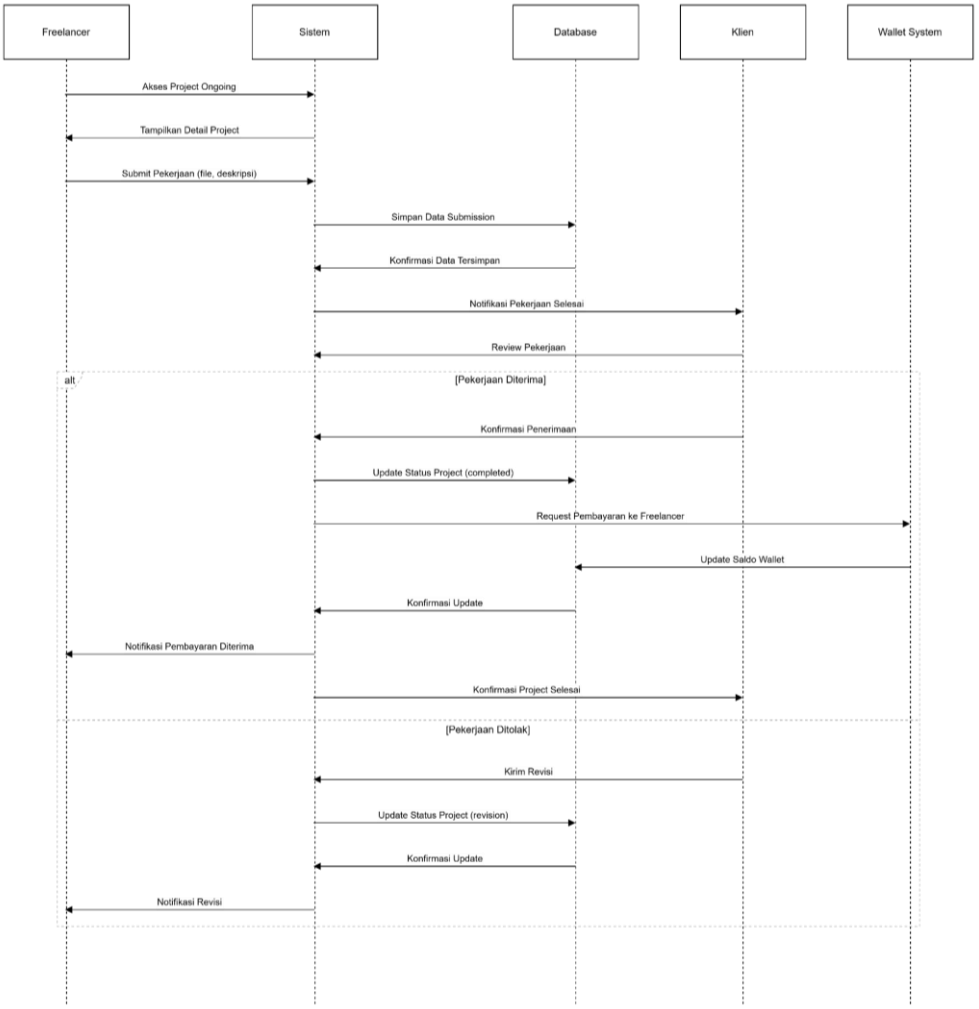
Dalam proses review dan pemilihan freelancer, klien mengakses daftar aplikasi melalui sistem. Sistem mengambil data dari database dan menampilkan daftar kepada klien. Setelah memilih dan meninjau aplikasi freelancer, klien menentukan pilihan. Sistem kemudian memperbarui status proyek di database, mengkonfirmasi ke klien, dan mengirimkan notifikasi kepada freelancer terpilih. Proses review dan pilih freelancer dapat digambarkan pada gambar 3.17 sebagai berikut :



Gambar 3. 17 *Sequence* Diagram *Review* & Pilih Freelancer

7. Proses Submit dan Pembayaran

Pada proses submit dan pembayaran, freelancer terlebih dahulu menyelesaikan proyek dan mengirimkan hasilnya melalui sistem. Sistem kemudian menyimpan data penyelesaian ke dalam database dan mengirimkan notifikasi kepada klien bahwa hasil proyek telah tersedia. Klien melakukan review terhadap hasil tersebut, dan apabila disetujui, klien mengonfirmasi penyelesaian pekerjaan. Sistem memperbarui status proyek menjadi selesai dan meneruskan informasi pembayaran ke sistem wallet. Setelah pembayaran diproses, sistem mengirimkan notifikasi kepada freelancer bahwa pembayaran telah diterima, sekaligus mengubah status proyek menjadi selesai secara keseluruhan. Proses submit dan pembayaran dapat digambarkan pada gambar 3.18 sebagai berikut :



Gambar 3. 18 *Sequence* Diagram *Submit* dan Pembayaran

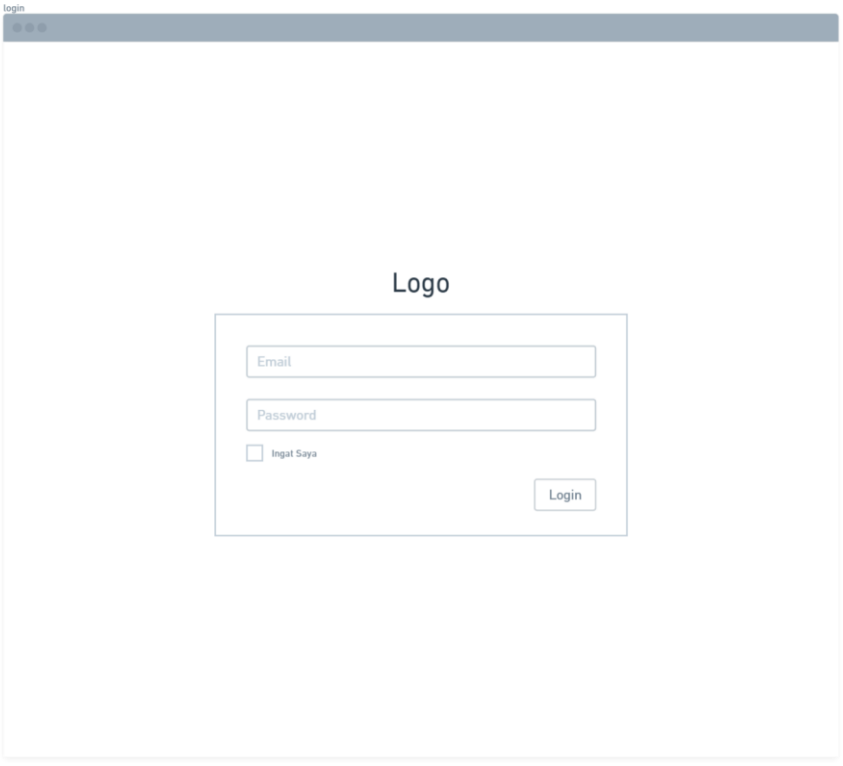
D. Desain Antarmuka Pengguna (UI)

Desain antarmuka (UI) pada platform KerjaBos berfokus pada tampilan visual yang bersih, konsisten, dan mudah dipahami oleh pengguna. Elemen-elemen seperti warna, ikon, tombol, tipografi, serta layout disusun secara sistematis untuk menciptakan kesan profesional dan modern. Setiap halaman memiliki struktur yang jelas, mulai dari navbar, konten utama, hingga footer, sehingga memudahkan pengguna dalam menavigasi fitur. Desain ini dibuat responsif agar dapat diakses dengan baik melalui berbagai perangkat, termasuk desktop dan mobile. Pemilihan UI yang tepat diharapkan dapat meningkatkan kenyamanan dan efisiensi saat pengguna berinteraksi dengan sistem.

Berikut ini adalah beberapa komponen utama antarmuka pengguna yang dirancang dalam sistem, disertai dengan deskripsinya:

1. Halaman Login

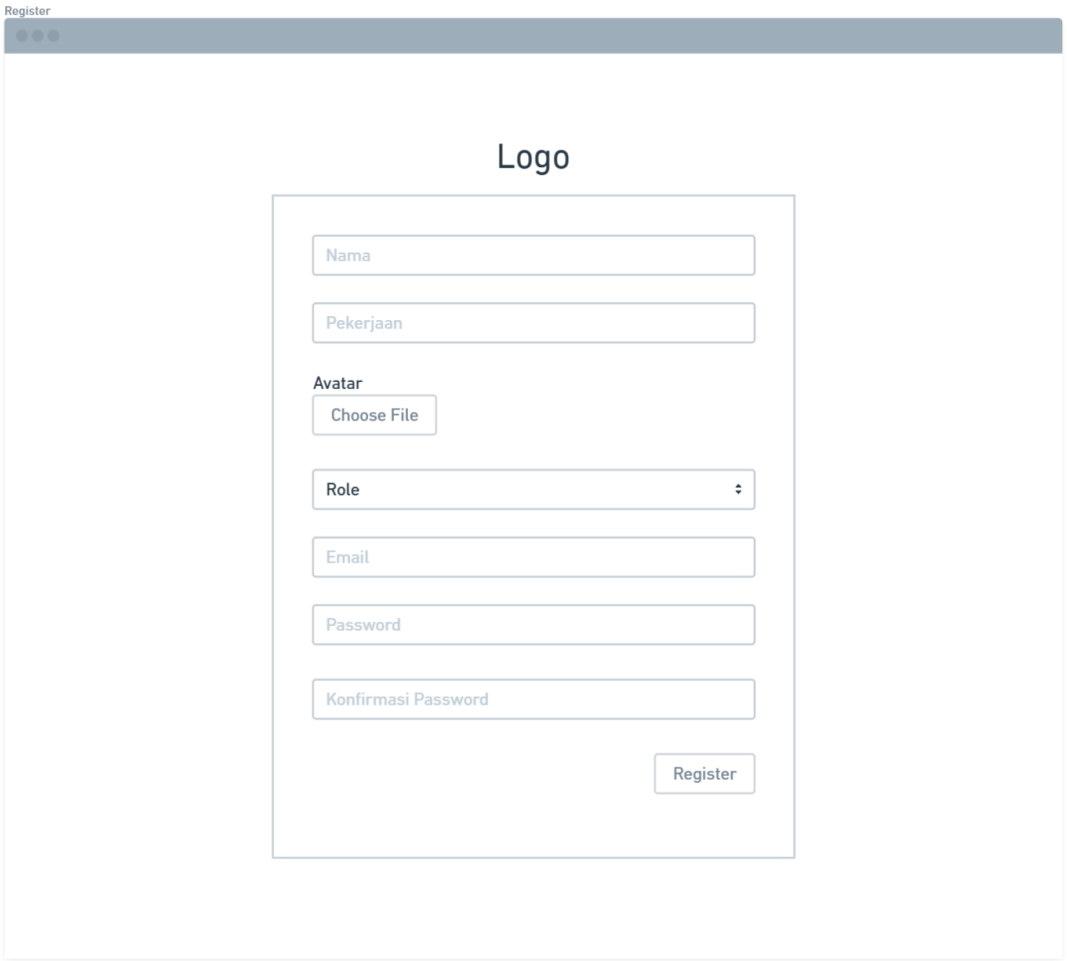
Halaman login KerjaBos dirancang sebagai akses awal pengguna ke dalam sistem. Terdapat form input email dan password, serta opsi “Ingat Saya” untuk menyimpan sesi. Desain dibuat sederhana, rapi, dan responsif agar mudah digunakan di berbagai perangkat. Penempatan elemen difokuskan pada kemudahan autentikasi secara cepat dan aman. Tampilan Halaman Login digambarkan pada gambar 3.19 sebagai berikut :



Gambar 3. 19 Tampilan Halaman *Login*

2. Halaman Register

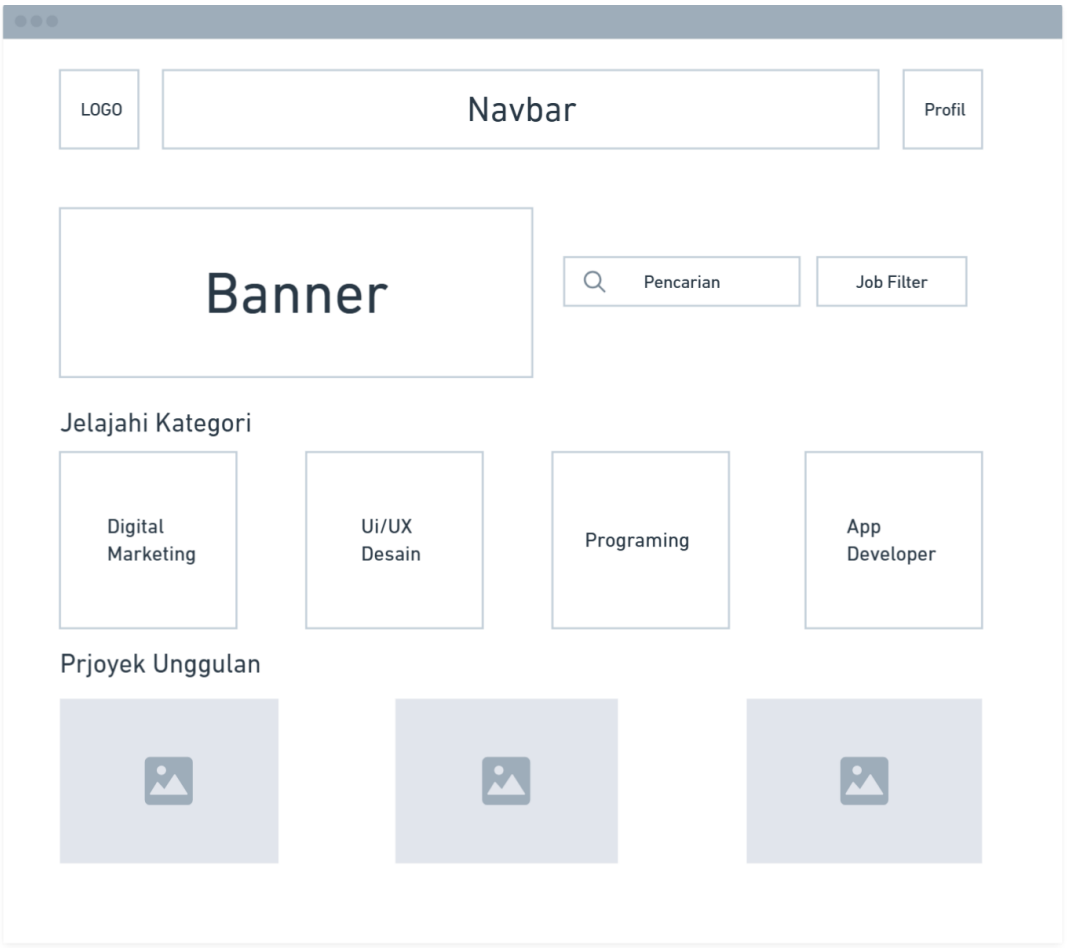
Halaman register pada platform KerjaBos dirancang untuk memudahkan pengguna baru dalam membuat akun. Form registrasi mencakup input nama, pekerjaan, avatar, peran pengguna (freelancer atau klien), email, password, dan konfirmasi password. Desain disusun secara sederhana dan terpusat agar mudah diisi, dengan struktur yang jelas untuk menghindari kesalahan input. Tujuannya adalah memastikan proses pendaftaran berjalan cepat, rapi, dan aman sebelum pengguna mengakses sistem. Tampilan Halaman Register digambarkan pada gambar 3.20 sebagai berikut :



Gambar 3. 20 Tampilan Halaman *Register*

3. Halaman Dashboard

Halaman beranda merupakan tampilan utama yang diakses oleh pengguna setelah login ke platform KerjaBos. Halaman ini menampilkan navbar yang berisi logo, menu navigasi, dan tombol profil. Terdapat banner utama untuk informasi promosi, kolom pencarian, serta fitur job filter untuk memudahkan pencarian proyek. Pengguna juga dapat menjelajahi berbagai kategori pekerjaan seperti Digital Marketing, UI/UX Desain, Programming, dan App Developer. Di bagian bawah, ditampilkan proyek unggulan yang menjadi referensi awal bagi freelancer. Desain halaman ini dibuat sederhana, responsif, dan terstruktur untuk menunjang kemudahan akses dan pengalaman pengguna. Tampilan Halaman Dashboard digambarkan pada gambar 3.21 sebagai berikut :



Gambar 3. 21 Tampilan Halaman *Dashboard*

4. Halaman Mengelola Kategori

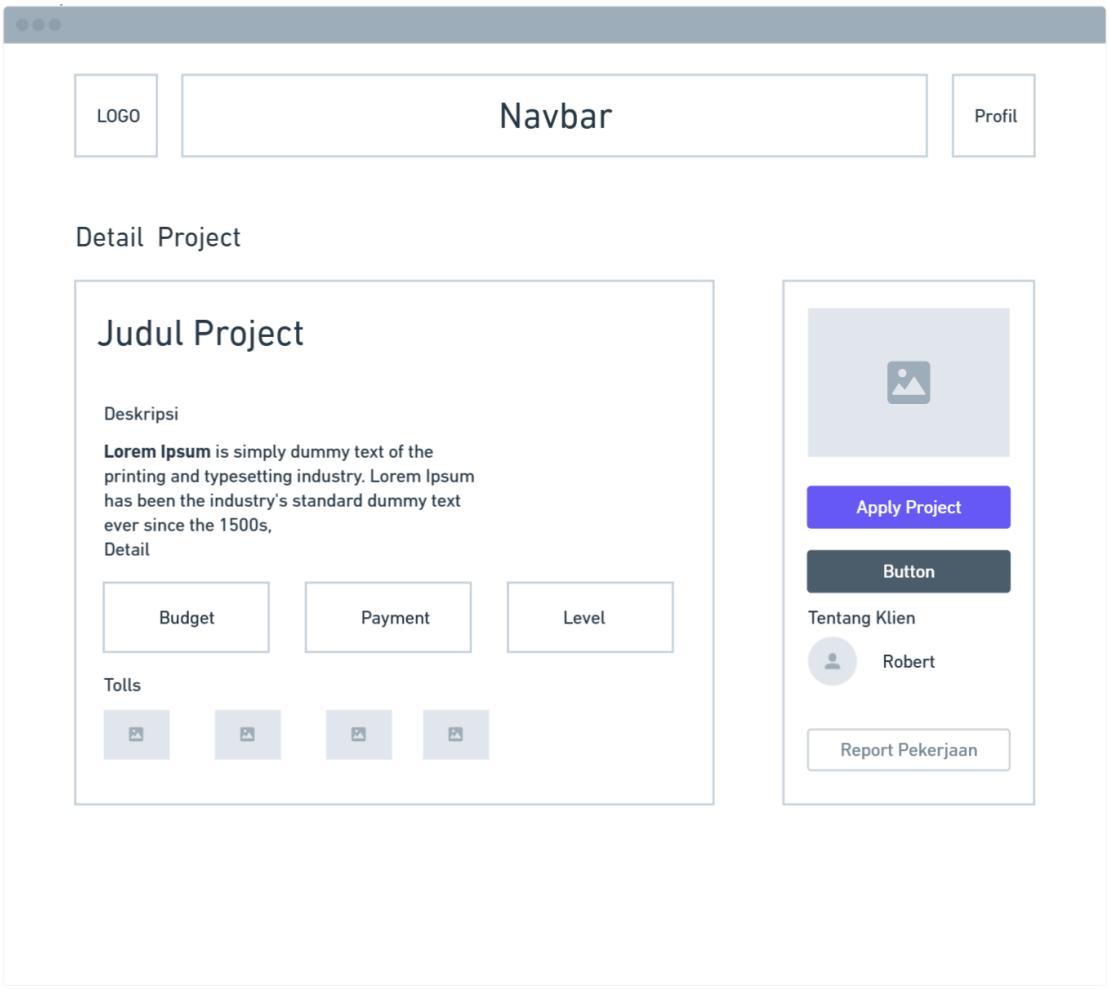
Halaman Mengelola Kategori dirancang untuk memudahkan admin dalam mengatur daftar kategori pekerjaan pada platform KerjaBos. Tampilan terdiri dari daftar kategori yang ditampilkan secara vertikal, dilengkapi tombol Edit dan Hapus di setiap baris untuk pengelolaan data. Tersedia juga tombol Tambahkan untuk menambah kategori baru. Desain dibuat minimalis dan terstruktur agar memudahkan navigasi dan mempercepat proses manajemen kategori secara efisien. Tampilan Halaman Mengelola Kategori digambarkan pada gambar 3.22 sebagai berikut :



Gambar 3. 22 Tampilan Halaman Manajemen Kategori

5. Halaman Detail *Project*

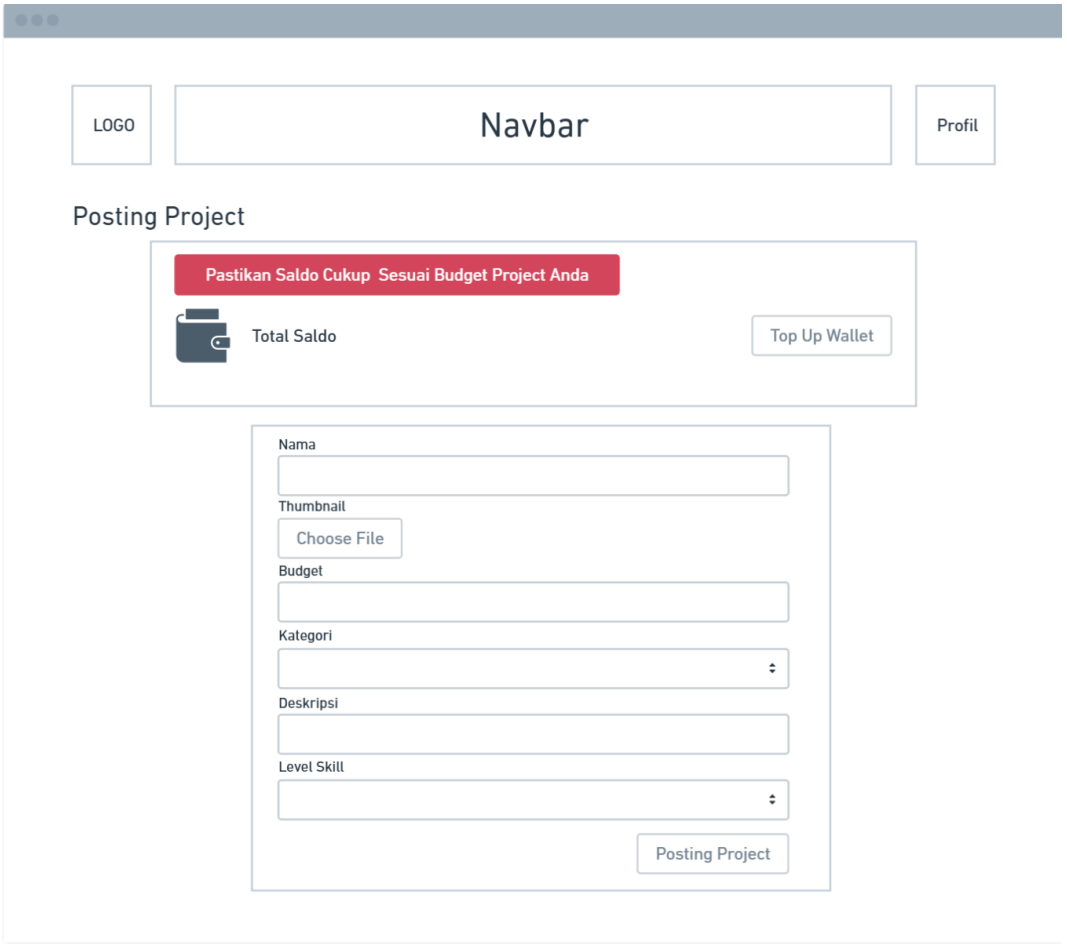
Halaman Detail Project menampilkan informasi lengkap mengenai sebuah proyek yang tersedia di platform KerjaBos. Di bagian utama, terdapat judul proyek, deskripsi singkat, serta informasi pendukung seperti budget, metode pembayaran, dan tingkat kesulitan (level). Fitur tambahan seperti daftar tools yang dibutuhkan juga ditampilkan secara visual. Di sisi kanan, terdapat profil klien termasuk nama dan foto, serta tombol Apply Project bagi freelancer yang tertarik melamar proyek. Halaman ini dirancang sederhana namun informatif, memudahkan pengguna memahami proyek dan melakukan aksi secara langsung. Tampilan Halaman Detail Project digambarkan pada gambar 3.23 sebagai berikut :



Gambar 3. 23 Tampilan Halaman Detail *Project*

6. Halaman Posting *Project*

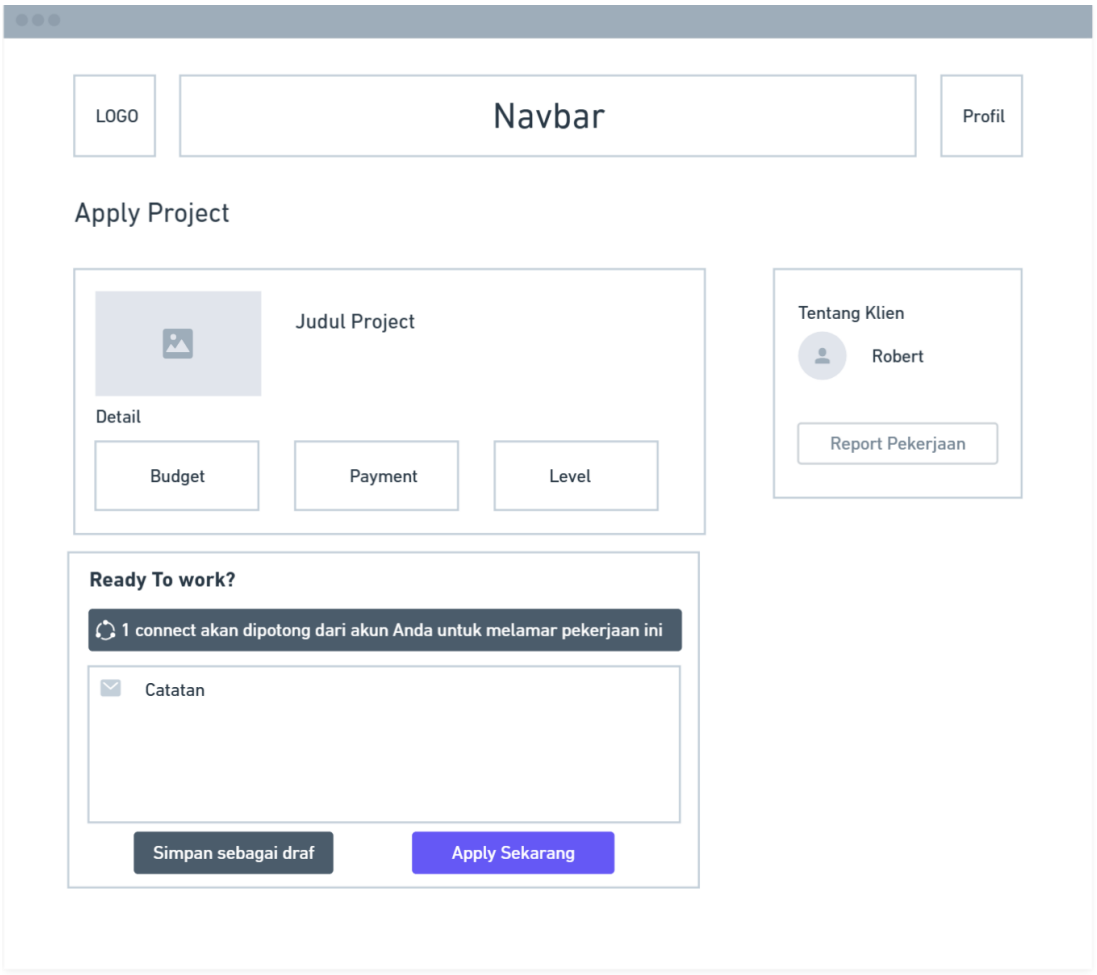
Halaman Posting Project berfungsi sebagai tempat bagi klien untuk membuat dan mempublikasikan proyek baru di platform. Di bagian atas, terdapat peringatan untuk memastikan saldo mencukupi sesuai budget proyek, dengan opsi tombol Top Up Wallet. Formulir yang tersedia mencakup input untuk nama proyek, thumbnail gambar, budget, kategori proyek, deskripsi, serta tingkat keahlian (level skill) yang dibutuhkan. Setelah semua data diisi, klien dapat mengklik tombol Posting Project untuk mempublikasikan proyek. Tampilan ini sederhana, terstruktur, dan memudahkan pengguna dalam mengisi informasi penting secara efisien. Tampilan Halaman Posting *Project* digambarkan pada gambar 3.24 sebagai berikut :



Gambar 3. 24 Tampilan Halaman *Posting Project*

7. Halaman *Apply Project*

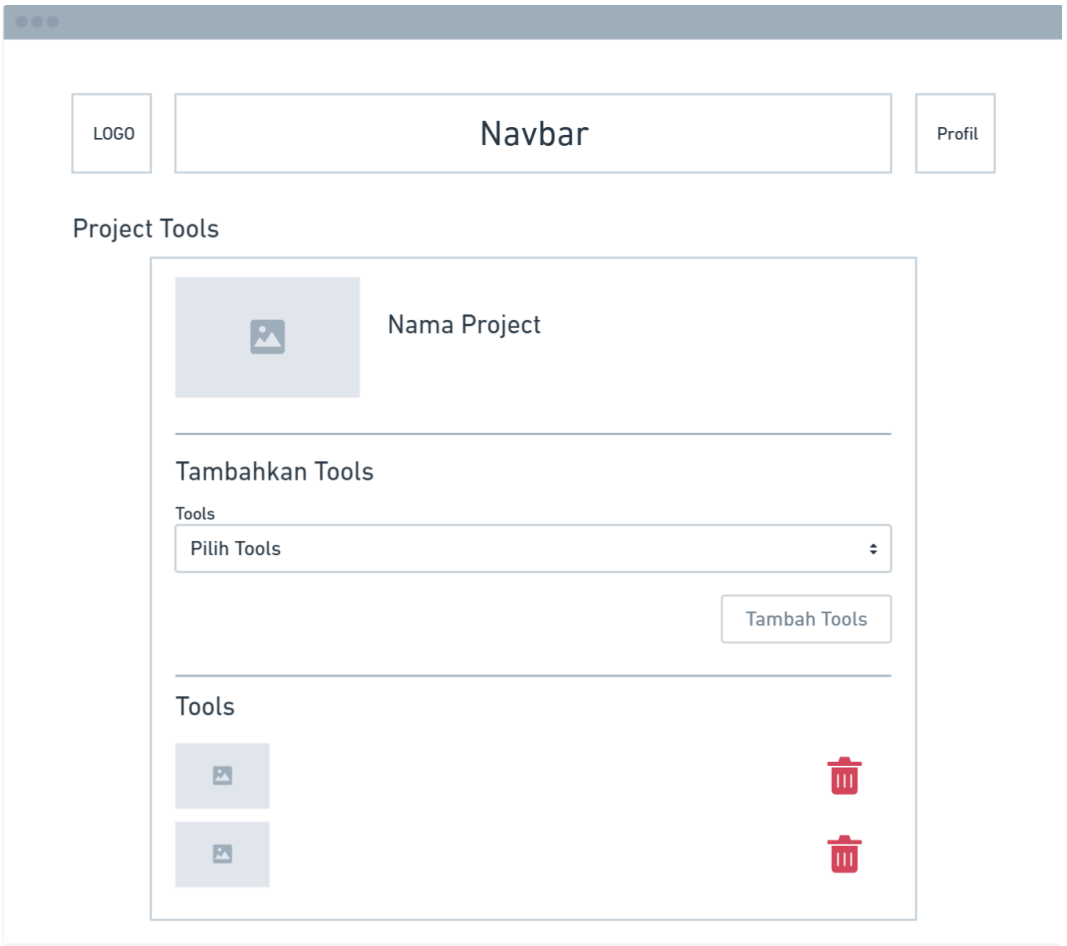
Halaman ini digunakan oleh freelancer untuk melamar sebuah proyek. Di bagian kiri, ditampilkan informasi proyek seperti judul, budget, metode pembayaran, dan level skill yang dibutuhkan. Freelancer dapat menulis catatan tambahan kepada klien dan memilih untuk menyimpan sebagai draf atau langsung melamar dengan menekan tombol "Apply Sekarang". Sistem akan memotong 1 connect saat melamar. Di sisi kanan, terdapat informasi tentang klien dan tombol untuk melaporkan proyek jika diperlukan. Tampilan Halaman *Apply Project* digambarkan pada gambar 3.25 sebagai berikut :



Gambar 3. 25 Tampilan Halaman *Apply Project*

8. Halaman Mengelola *Tools*

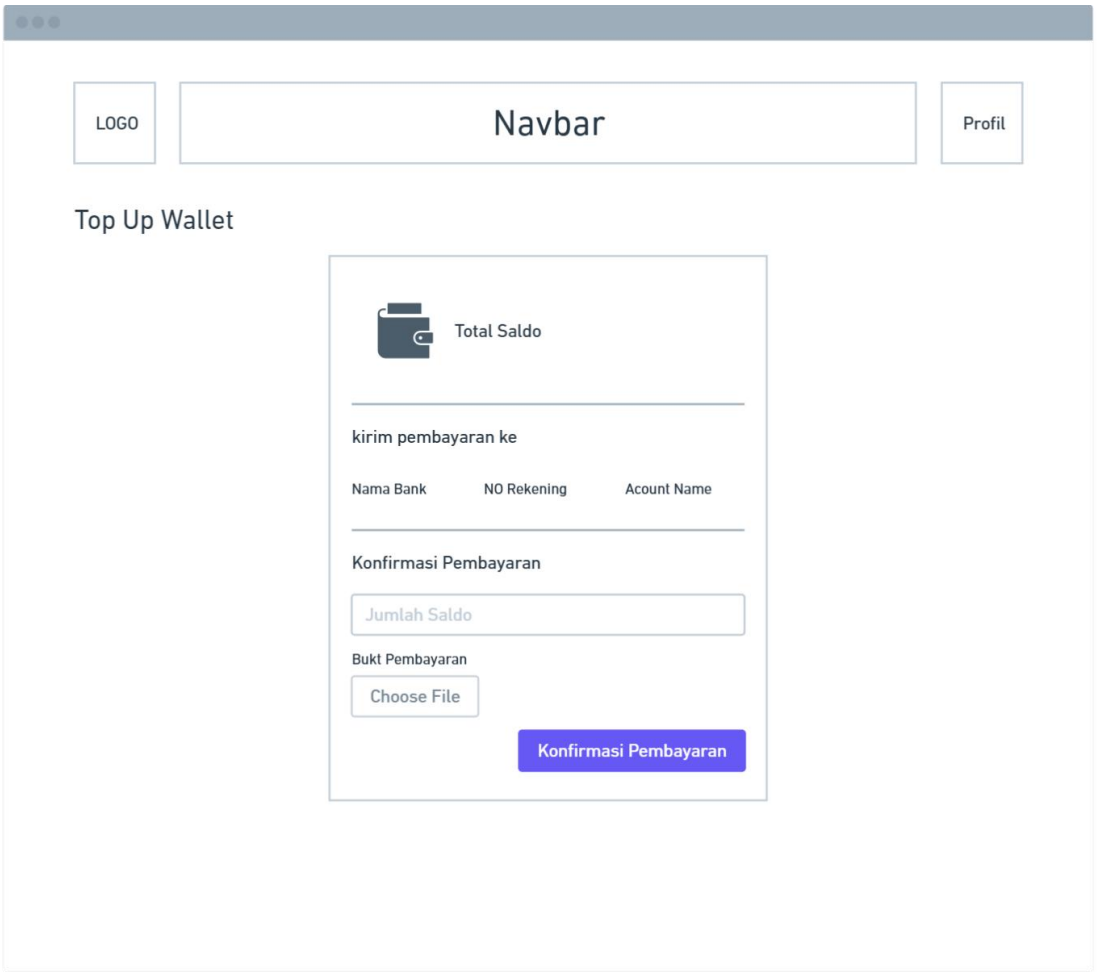
Halaman ini berfungsi untuk mengelola *tools* yang digunakan dalam suatu proyek. Pengguna dapat memilih tools dari daftar dropdown dan menambahkannya ke proyek menggunakan tombol "Tambah *Tools*". *Tools* yang sudah ditambahkan akan ditampilkan di bawahnya lengkap dengan ikon hapus untuk mengelola atau menghapus *tools* yang tidak diperlukan. Tampilan halaman disusun rapi untuk memudahkan pengguna dalam pengelolaan alat kerja proyek. Tampilan Halaman Mengelola *Tools* digambarkan pada gambar 3.26 sebagai berikut :



Gambar 3. 26 Tampilan Halaman Mengelola *Tools*

9. Halaman *Top Up*

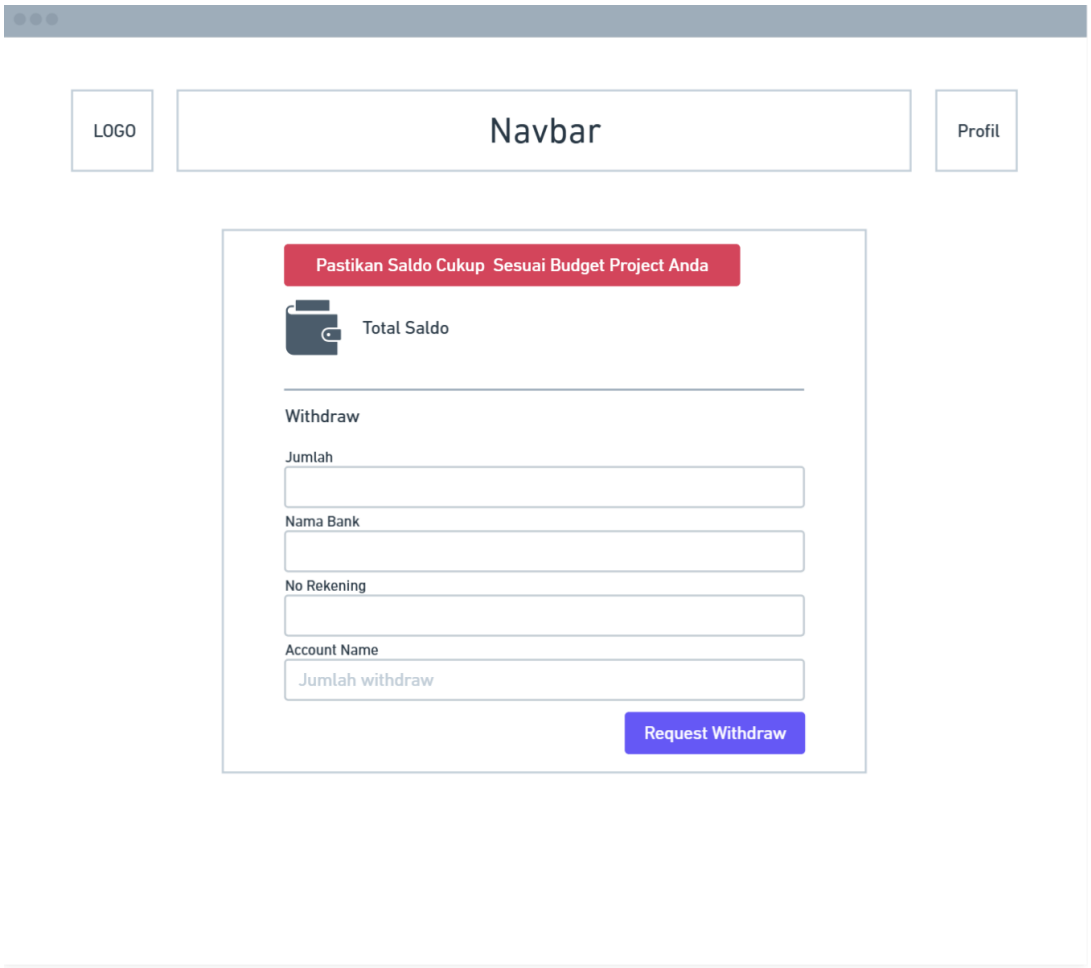
Halaman ini digunakan untuk melakukan pengisian saldo ke dalam dompet digital pengguna. Pengguna akan mendapatkan informasi rekening tujuan berupa nama bank, nomor rekening, dan nama pemilik akun. Setelah melakukan transfer, pengguna wajib mengisi jumlah saldo yang ditransfer dan mengunggah bukti pembayaran melalui form konfirmasi. Proses ini diselesaikan dengan menekan tombol "Konfirmasi Pembayaran" untuk memvalidasi transaksi. Tampilan Halaman *Top Up* digambarkan pada gambar 3.27 sebagai berikut :



Gambar 3. 27 Tampilan Halaman *Top Up*

10. Halaman *Withdraw*

Halaman ini digunakan oleh pengguna untuk menarik saldo dari dompet digital ke rekening bank pribadi. Pengguna diminta mengisi jumlah penarikan, nama bank, nomor rekening, dan nama pemilik akun. Terdapat peringatan di bagian atas untuk memastikan saldo cukup sebelum melakukan penarikan. Setelah data diisi, pengguna menekan tombol "Request Withdraw" untuk memproses permintaan penarikan dana. Tampilan Halaman Withdraw digambarkan pada gambar 3.28 sebagai berikut :



Gambar 3. 28 Tampilan Halalaman *Withdraw*

3.6 Implementasi

Implementasi pada tahap ini adalah tahap nyata dalam mewujudkan rancangan sistem ke dalam bentuk aplikasi yang siap digunakan. Implementasi dilakukan dengan metode terstruktur berdasarkan hasil dari analisis kebutuhan dan perancangan sistem. Metode yang digunakan untuk melakukan seluruh proses pengembangan pada proyek ini adalah dengan menggunakan pendekatan arsitektur *monolith*, di mana seluruh logika aplikasi, antarmuka pengguna dan pengelolaan data terintegrasi dalam satu struktur proyek *laravel*.

Proses implementasi mencakup beberapa langkah sebagai berikut:

1. Penyiapan Lingkungan Pengembangan

   Langkah awal dilakukan dengan melakukan konfigurasi lingkungan kerja menggunakan *Laravel* versi 12 sebagai dasar utama framework. *Laravel* dipilih karena Lisensi memiliki struktur yang jelas, mendukung pengembangan berbasis *MVC* , serta memiliki ekosistem yang kuat untuk membangun aplikasi berbasis web dengan efisien .

   Database sistem dibangun menggunakan *PostgreSQL*, yang memiliki beberapa kelebihan diantaranya yakni kecepatan, keamanan, dan integritas data. Sistem ini dikembangkan di sekitar server lokal yang menyimpan *PHP* dan *composer* sebagai pengelola *package dependency PHP*, dimana *Laravel King* untuk *version controll system git* dan *mix* untuk manajemen *aset frontend*.

2. Pembuatan Struktur Database

   Berdasarkan *Entity Relationship Diagram* (ERD) yang telah dirancang sebelumnya, database dibangun untuk mengakomodasi seluruh entitas utama seperti *users*, *projects*, *wallet*, *transactions*, *project applicants*, dan *tools*. Relasi antar tabel diimplementasikan menggunakan migrasi *Laravel* yang memungkinkan pengelolaan skema database secara terstruktur dan konsisten. Penyesuaian tipe data, relasi *one-to-many* dan *many-to-many*, serta penambahan *indexing* dilakukan guna mendukung efisiensi pengambilan data selama aplikasi berjalan

3. Pengembangan Fitur Inti

   Setelah struktur database selesai dibangun, pengembangan dilanjutkan pada sisi logika aplikasi dan antarmuka pengguna. Fitur-fitur yang dikembangkan meliputi:

   a. Modul Registrasi dan Login

      Pengguna dapat mendaftar sebagai freelancer atau klien. Sistem otentikasi menggunakan *Laravel Auth* dengan tambahan *middleware role-based* menggunakan *Spatie Permission*.

   b. Manajemen Proyek

      Klien dapat membuat proyek baru dengan mengisi informasi seperti nama

proyek, deskripsi, kategori, skill level, dan tools yang digunakan. Proyek akan ditampilkan di dashboard dapat dilamar oleh freelancer.

c. Sistem Apply

Freelancer yang telah mendaftar akan mendapatkan 10 *connect* awal secara otomatis. Setiap kali *apply* proyek, sistem akan mengurangi jumlah connect secara otomatis. Mekanisme ini bertujuan untuk mengatur keseimbangan antara peluang dan kualitas pelamar.

d. Pengelolaan Proposal dan Seleksi Freelancer

Freelancer dapat mengirimkan proposal ke proyek yang diminati. Klien dapat meninjau semua proposal, mengatur jadwal interview atau meeting (misalnya melalui Zoom), dan memilih freelancer yang sesuai.

e. Fitur Connect

Fitur Connect dalam sistem KerjaBos berperan sebagai mekanisme kontrol dalam proses lamaran proyek oleh freelancer. Setiap freelancer yang telah melakukan registrasi akan otomatis memperoleh sejumlah connect (misalnya 10 connect) sebagai saldo awal. Connect ini akan berkurang setiap kali freelancer mengajukan lamaran pada suatu proyek.

Konsep ini diadaptasi dari model token-based application yang umum digunakan pada platform freelance profesional, di mana penggunaan connect mendorong selektivitas dan keseriusan freelancer dalam melamar proyek. Dengan demikian, klien akan menerima proposal dari kandidat yang memang benar-benar tertarik dan berkualitas.

Fitur ini juga memberi insentif kepada freelancer untuk menggunakan connect secara bijak, serta dapat dikembangkan untuk ditop-up melalui sistem dompet digital (wallet) apabila connect habis. Dari sisi sistem, logika pemotongan connect diatur secara otomatis melalui backend Laravel dan dikaitkan langsung dengan data pengguna di basis data.

Dengan adanya fitur ini, sistem KerjaBos tidak hanya mengatur alur kerja secara efisien, tetapi juga menciptakan ekosistem yang lebih profesional dan kompetitif dalam mempertemukan freelancer dan klien.

f. Dompet Digital (*Wallet*) dan Transaksi

Fitur Wallet pada platform KerjaBos dirancang sebagai dompet digital internal yang memfasilitasi seluruh proses transaksi antara klien, freelancer, dan admin dalam ekosistem sistem. Fitur ini berfungsi untuk menyimpan saldo pengguna, baik untuk kebutuhan pembayaran proyek maupun untuk pengelolaan penghasilan dari pekerjaan yang telah diselesaikan.

Bagi klien, wallet digunakan untuk melakukan top up saldo sebelum memposting proyek. Proses top up dilakukan melalui metode transfer manual ke rekening yang disediakan, disertai dengan bukti pembayaran yang harus diunggah melalui sistem. Admin kemudian akan memverifikasi bukti tersebut, dan jika valid, saldo akan otomatis masuk ke akun wallet klien.

Saldo dalam wallet inilah yang akan digunakan sebagai metode pembayaran kepada freelancer setelah proyek diselesaikan. Hal ini mendukung sistem transaksi yang aman, transparan, dan terverifikasi.

Sementara itu, bagi freelancer, wallet berfungsi sebagai tempat menampung pendapatan dari proyek yang telah diselesaikan. Freelancer juga memiliki akses untuk melakukan withdraw dana dari wallet ke rekening pribadi mereka, setelah mendapatkan konfirmasi dari pihak admin.

Secara teknis, pengelolaan wallet dikembangkan menggunakan Laravel dan PostgreSQL. Tabel wallets dan wallet_transactions digunakan untuk mencatat informasi saldo, riwayat transaksi, jenis transaksi (top-up atau withdraw), serta status verifikasi.

Implementasi fitur ini bertujuan untuk meminimalkan risiko penipuan, meningkatkan kepercayaan antar pengguna, serta menciptakan sistem pembayaran internal yang terintegrasi dan efisien di dalam platform KerjaBos.

4. Integrasi Antarmuka Pengguna

Semua fitur dikemas dalam antarmuka berbasis web yang *user-friendly* dan responsif. *Frontend* dibangun menggunakan *Blade Template Engine* dari *Laravel*, dengan tambahan library *CSS* dan *JavaScript* seperti *TailwindCSS* atau *Bootstrap* (jika digunakan), untuk mendukung tampilan yang modern dan mudah digunakan oleh pengguna dari berbagai perangkat.

3.7 Pengujian

Pengujian merupakan tahap penting dalam proses pengembangan perangkat lunak untuk memastikan bahwa sistem yang telah dibangun berjalan sesuai dengan fungsinya, aman digunakan, dan memenuhi kebutuhan pengguna. Dalam penelitian ini, pengujian dilakukan terhadap platform *KerjaBos* guna mengevaluasi kinerja, kestabilan, serta pengalaman pengguna dari berbagai fitur yang telah diimplementasikan.

Pengujian Penerimaan Pengguna *(User Acceptance Testing - UAT)*

User Acceptance Testing dilakukan untuk menilai seberapa baik sistem diterima oleh pengguna akhir. Pengujian ini bersifat eksploratif dan subjektif, dengan melibatkan beberapa calon pengguna dari masing-masing peran (freelancer dan klien) untuk mencoba sistem secara langsung. Kegiatan UAT dilakukan dengan memberikan akun demo kepada pengguna dan mengarahkan mereka untuk menyelesaikan satu siklus penggunaan, mulai dari proses registrasi, melakukan posting proyek, mengajukan aplikasi pada proyek, hingga menyelesaikan proses pembayaran. Setelah pengguna menyelesaikan seluruh alur tersebut, tim kemudian mengumpulkan umpan balik melalui kuesioner sederhana atau wawancara informal guna mengevaluasi pengalaman dan kendala yang ditemui selama pengujian. Aspek yang dievaluasi dalam kegiatan UAT meliputi kemudahan penggunaan atau usability, kejelasan tampilan antarmuka (UI), kecepatan respon sistem, serta tingkat kepuasan pengguna terhadap alur kerja yang disediakan. Seluruh hasil evaluasi dari UAT ini kemudian digunakan sebagai bahan pertimbangan untuk tahap selanjutnya, yaitu proses evaluasi lanjutan dan perbaikan sistem agar lebih optimal

REFERENSI

Dhian, A. N., Dewayani, E., & Rusdi, Z. 2022. Perancangan Dan Pembuatan Program Aplikasi Freelance Berbasis Web.

Masakazu, K. M., Ersi Sisdianto, Gede Suwardika, & Dhimas Setyo Nugroho. (2023). Peran Digital Freelancer Marketplace dan Media Sosial Terhadap Perkembangan Gig Economy Worker. Jurnal Pendidikan Ekonomi Undiksha, 15(1), 214–225.

Ar-raniry, U. I. N., Tarbiyah, F., Keguruan, D. A. N., Studi, P., & Teknologi, P. (2024). PERANCANGAN USER INTERFACE DAN USER EXPERIEN CE WEBSITE FREELANCE MARKETPLACE JASA DIGITAL MENGGUNAKAN METODE USER CENTERED DESIGN (UCD).

Triyas Niko Saputra & Imam Husni Al Amin. (2023). APLIKASI MANAJEMEN TUGAS TASKIFY UNTUK MENENTUKAN PRIORITAS PEKERJAAN FREELANCER DENGAN AHP DAN TOPSIS. Jurnal INSTEK (Informatika Sains dan Teknologi), 8(2), 292–301. https://doi.org/10.24252/instek.v8i2.42202

Ade Sutedi, Dzikri Nursyaban, Cahya Mutiara, Dede Sopiah, Diaz Radhian Salam, Diva Nuratnika Rahayu, Hasfi Syahrul Ramadhan, Ilham Muhamad Ramdan, Intan Sri Fatmalasari, & Irsyad Ahmad. (2021). Program Pelatihan Web Development untuk Komunitas Maya. *Jurnal PkM MIFTEK*, *2*(1), 19–27. https://doi.org/10.33364/miftek/v.2-1.964

Soegiarto, R., & Susilowati, M. (n.d.). (2024) *PERANCANGAN SISTEM INFORMASI MARKETPLACE USAHA MIKRO KECIL MENENGAH DENGAN FOKUS PADA SISTEM KONSINYASI. 6*.

Ilhamsyah, S., Fajri, M. H. F., & Defana, M. A. B. (2025). *Pengaruh Pekerjaan Freelance Dan Kegiatan Berorganinasi Dalam Meningkatkan Kesiapan Kerja Mahasiswa Universitas Bangka Belitung*.

Khoirunisa, Q., Mayana, R. F., & Muchtar, H. N. (2023). IMPLEMENTASI SKEMA PEMBIAYAAN PADA HAK CIPTA DALAM EKONOMI KREATIF DIHUBUNGKAN DENGAN ASAS ALTER EGO, TEORI PENGHARGAAN DAN TEORI HASIL KERJA. EKSEKUSI, 5(2), 141. https://doi.org/10.24014/je.v5i2.24010

Raynaldi, J., & Somya, R. (2023). Perancangan aplikasi e-kantin berbasis website menggunakan framework laravel. Jurnal Pendidikan Teknologi Informasi (JUKANTI), 6(2). https://doi.org/10.37792/jukanti.v6i2.1052

Endra, R. Y., Aprilinda, Y., Dharmawan, Y. Y., & Ramadhan, W. (2021). Analisis Perbandingan Bahasa Pemrograman PHP Laravel dengan PHP Native pada Pengembangan Website. EXPERT: Jurnal Manajemen Sistem Informasi dan Teknologi, 11(1), 48. https://doi.org/10.36448/expert.v11i1.2012

Faruqi, M. H., Anshary, F. M. A., & Lathifah, S. N. (n.d.). (2025). Perancangan Back-end Aplikasi Education Management System Modul Akademik Menggunakan Metode Iterative Incremental Pada SMPN 1 Magetan.

Wijayanti, A., ., F., & Saraswati, N. M. S. (2023). RANCANG BANGUN SISTEM INFORMASI GEOGRAFIS PELAYANAN KESEHATAN DI KABUPATEN BREBES SELATAN BERBASIS WEB. Jurnal Informatika Dan Riset, 1(1), 24 - 30. https://doi.org/10.36308/iris.v1i1.473

Sidiq, M. A. Z., Anshori, M. I., & Yaqin, R. A. (2024). Penerapan Arsitektur Monolitik Pada Aplikasi Jasa Service Online Tekku Berbasis Web. JUKI : Jurnal Komputer dan Informatika, 6(1), 27–36. https://doi.org/10.53842/juki.v6i1.418

Iqbal, M., Syahputra, A. K., & Handoko, W. (2022). PENERAPAN MONOLITIC ARSITECTURE PADA APLIKASI UJIAN ONLINE BERBASIS WEB. 2(2).

Barlage, M., Born, A. V., & Witteloostuijn, A. V. (2022). The needs of freelancers and the characteristics of 'gigs': Creating beneficial relations between freelancers and their hiring organizations. Emerald Open Research, 1(4).

Puspita, M. A., Pradipta, R. A., & Muhammad, M. A. (2024). User Acceptance Testing dan Performance Testing Pada Pengembangan Website Kanal Pengetahuan Dikti. 08.

Dillah, A., Nama, G. F., Budiyanto, D., & Muhammad, M. A. (2024). RANCANG BANGUN APLIKASI MONITORING OPERASI P2TL PENGUKURAN TIDAK LANGSUNG 2 PHASA DI PT. PLN (PERSERO) UNIT PELAKSANA PELAYANAN PELANGGAN (UP3) METRO. Jurnal Informatika dan Teknik Elektro Terapan, 12(3). https://doi.org/10.23960/jitet.v12i3.4458

Bab 5
PENUTUP

5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pembahasan yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Platform KerjaBos berhasil dirancang sebagai sistem marketplace digital berbasis web yang mampu mempertemukan freelancer dan klien dalam satu wadah kerja yang terintegrasi. Sistem ini mendukung kolaborasi digital yang fleksibel, efisien, dan profesional, khususnya bagi alumni Universitas Muhammadiyah Magelang.
2. Fitur-fitur utama seperti registrasi akun, posting proyek, pengajuan proposal oleh freelancer, pemilihan freelancer oleh klien, serta notifikasi sistem telah terimplementasi secara fungsional dan mendukung proses kerja berbasis proyek dari awal hingga akhir.
3. Sistem transaksi yang dikembangkan berbasis dompet digital (wallet), dengan dukungan proses top-up, unggah bukti pembayaran, dan validasi oleh admin. Hal ini menjawab kebutuhan akan sistem pembayaran yang aman dan transparan, serta membangun kepercayaan antar pengguna di dalam platform.
4. Arsitektur monolitik berbasis Laravel dan PostgreSQL memberikan stabilitas dan struktur yang solid bagi sistem KerjaBos. Pendekatan ini memudahkan proses integrasi antar fitur dan memungkinkan pengembangan sistem lanjutan secara efisien.
5. Dibandingkan dengan platform sejenis seperti Fiverr atau Sribulancer, sistem KerjaBos memiliki keunggulan dalam konteks lokal dan keberpihakan terhadap alumni perguruan tinggi, terutama dengan fitur verifikasi manual, sistem connect, serta dashboard yang disesuaikan. Hal ini menjadikan KerjaBos sebagai alternatif kompetitif untuk kolaborasi kerja berbasis proyek di lingkup kampus maupun komunitas profesional nasional.
6. Potensi pengembangan sistem KerjaBos di masa mendatang sangat terbuka, di antaranya dengan mengintegrasikan fitur escrow, sistem pembayaran otomatis melalui gateway lokal (e-wallet), serta penguatan fitur komunikasi real-time antar pengguna. Pengembangan ini akan semakin memperkuat nilai tambah platform sebagai solusi digital end-to-end yang modern dan adaptif

7. Hasil pengujian User Acceptance Testing (UAT) menunjukkan bahwa sistem *KerjaBos* telah diterima dengan sangat baik oleh pengguna. Pengujian dilakukan dengan menyebarkan kuesioner digital kepada 9 responden yang terdiri dari klien, freelancer, dan admin. Masing-masing responden memberikan penilaian terhadap 10 indikator aspek sistem. Berdasarkan hasil rata-rata pernyataan, diperoleh skor rata-rata sebesar 4.4 dari skala 5, yang dikonversikan ke dalam bentuk persentase menjadi:
$$\left(\frac{4.4}{5}\right) \times 100\% = 88\%$$
Nilai ini menunjukkan bahwa sistem *KerjaBos* memiliki tingkat penerimaan pengguna yang sangat baik. Mayoritas responden merasa bahwa sistem mudah digunakan, navigasi intuitif, dan fitur berjalan sesuai kebutuhan.
8. Dengan hasil UAT tersebut, sistem dinilai layak untuk digunakan secara fungsional, serta telah memenuhi sebagian besar kebutuhan pengguna akhir dalam ekosistem kerja digital.

5.2 Saran

Berdasarkan hasil pengembangan sistem dan kesimpulan yang telah dicapai, terdapat beberapa saran yang dapat diajukan untuk pengembangan dan penelitian lanjutan sebagai bentuk penyempurnaan terhadap sistem KerjaBos:

1. Penerapan sistem escrow otomatis perlu dipertimbangkan dalam versi selanjutnya, agar alur transaksi antara klien dan freelancer lebih terjamin keamanannya. Sistem ini akan memastikan bahwa pembayaran hanya diproses setelah proyek benar-benar diselesaikan dan disetujui, sehingga risiko perselisihan dapat diminimalkan.
2. Pengembangan sistem pembayaran digital yang terintegrasi langsung dengan layanan e-wallet lokal seperti Dana, OVO, atau QRIS, sangat penting untuk mendukung efisiensi dan kenyamanan pengguna dalam proses top-up dan withdraw saldo. Fitur ini juga akan mengurangi ketergantungan pada proses verifikasi manual oleh admin.
3. Penambahan fitur komunikasi langsung dalam platform, seperti chat antar pengguna atau ruang diskusi proyek, dapat meningkatkan efisiensi kerja sama antara freelancer dan klien serta mengurangi potensi miskomunikasi selama pengerjaan proyek.
4. Untuk menjaga kualitas pengguna, perlu dikembangkan sistem validasi portofolio atau sertifikasi skill, khususnya bagi freelancer. Hal ini akan membantu klien dalam menilai kredibilitas pelamar dan meningkatkan kepercayaan antar pengguna.
5. Dari sisi keamanan, penelitian lebih lanjut perlu dilakukan untuk mengoptimalkan sistem enkripsi data dan autentikasi pengguna, guna mengantisipasi risiko kebocoran data dan penyalahgunaan akun yang mungkin terjadi di masa mendatang.

5.3 Keterbatasan

Dalam pengembangan sistem KerjaBos, terdapat beberapa keterbatasan yang perlu diakui secara objektif sebagai ruang lingkup yang belum sepenuhnya terakomodasi oleh sistem yang telah dirancang. Keterbatasan ini disampaikan agar pembaca memahami batas implementasi sistem saat ini dan menjadi landasan bagi pengembangan di masa mendatang antara lain:

1. Fungsionalitas transaksi masih bersifat semi-manual, khususnya pada proses top-up saldo dompet digital. Meskipun sistem telah menyediakan unggahan bukti transfer dan fitur verifikasi oleh admin, belum terdapat integrasi langsung dengan layanan pembayaran otomatis (payment gateway), sehingga menimbulkan keterbatasan dalam hal kecepatan dan kenyamanan transaksi.
2. Sistem komunikasi antara klien dan freelancer masih terbatas, karena belum tersedianya fitur chat atau forum diskusi internal yang dapat memfasilitasi koordinasi selama masa pengerjaan proyek. Hal ini dapat berpotensi menyebabkan kesalahpahaman atau keterlambatan dalam proses revisi dan validasi pekerjaan.
3. Validasi terhadap portofolio atau keahlian pengguna belum dilakukan secara sistematis. Sistem saat ini belum menyediakan mekanisme untuk mengecek keabsahan skill atau riwayat kerja freelancer secara otomatis, sehingga klien masih perlu melakukan penilaian secara manual terhadap proposal yang masuk.
4. Pengujian terhadap sistem dilakukan dalam skala terbatas, baik dari segi jumlah pengguna maupun variasi skenario. Uji penerimaan pengguna (User Acceptance Testing) belum mencakup kondisi nyata pada lingkungan produksi dengan lalu lintas tinggi, sehingga performa sistem dalam skala besar belum dapat dievaluasi secara menyeluruh.
5. Penggunaan arsitektur monolitik membawa konsekuensi keterbatasan skalabilitas, khususnya jika sistem dikembangkan lebih lanjut untuk mendukung jumlah pengguna yang jauh lebih besar. Migrasi ke pendekatan mikro servis mungkin perlu dipertimbangkan di masa depan untuk meningkatkan fleksibilitas dan efisiensi pengelolaan modul.
6. Antarmuka sistem belum bersifat responsif, sehingga tampilan platform KerjaBos hanya optimal di perangkat desktop. Pengguna yang mengakses melalui perangkat mobile atau tablet akan mengalami kendala dalam navigasi dan kenyamanan visual. Hal ini membatasi aksesibilitas pengguna, terutama di era penggunaan mobile yang semakin dominan.
7. Kategori pekerjaan yang tersedia masih terbatas pada beberapa bidang IT tertentu, seperti UI/UX, programming, dan digital marketing. Sistem belum mendukung seluruh sub bidang pekerjaan IT secara menyeluruh, seperti jaringan, keamanan siber, analisis data, atau cloud computing. Hal ini membatasi keragaman proyek yang dapat dipublikasikan dan mengurangi daya tarik bagi pengguna dengan spesialisasi yang lebih luas.