

BAB 3

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Pelaksanaan kerja magang dilaksanakan di PT Dana Kini Indonesia pada Departemen *IT Solution* dengan posisi sebagai *IT Front End Developer Intern*. Selama periode magang, bimbingan teknis diberikan secara langsung oleh Bapak Fathariq selaku Mentor, yang bertanggung jawab dalam memastikan kualitas teknis dan kesesuaian proyek dengan standar perusahaan. Penugasan harian disusun melalui koordinasi antara Bapak Fathariq dan Bapak Suryadi selaku *Senior IT Front End Developer*, sedangkan evaluasi akhir atas kinerja dilakukan oleh Bapak Ricky selaku *Supervisor*.

Koordinasi pekerjaan dilakukan secara luring dan daring melalui platform Google Meet. Proyek dikembangkan dalam beberapa *sprint* dan dipantau menggunakan sistem manajemen proyek Jira. Pertemuan *daily stand-up* dilaksanakan setiap dua hari sekali guna memastikan kelancaran proses dan keselarasan antar tim. Proyek tersebut dipimpin oleh seorang *Business Analyst* dan melibatkan kolaborasi antar anggota tim dengan peran berbeda, yaitu *UI/UX Designer*, *Back End Developer*, *Front End Developer*, serta *Quality Assurance*.

3.2 Tugas yang Dilakukan

Selama menjalani program magang sebagai *IT Front End Developer Intern* di PT Dana Kini Indonesia, terdapat beberapa tugas yang dilakukan, antara lain:

1. Pengembangan Modul Autentikasi dan Modul Fungsional

Tugas difokuskan pada peningkatan sistem keamanan dan kenyamanan pengguna melalui pengembangan *two-factor authentication* (2FA) yang dilengkapi dengan CAPTCHA, serta registrasi berbasis OTP sebagai pengganti skema pendaftaran sebelumnya. Selain itu, dilakukan pengembangan berbagai modul utama seperti Dashboard, Sejarah Pembiayaan, Monitoring Verifikasi Data, dan Status Dokumen untuk mendukung kelengkapan serta integritas sistem secara menyeluruh. Proses ini juga mencakup integrasi antarmuka dengan layanan API serta penyesuaian desain antarmuka sesuai dengan kebutuhan pengguna.

2. Optimasi Proses *Front-End*

Untuk mendukung efisiensi kerja, diterapkan otomatisasi *build process* menggunakan Gulp pada aset *frontend*. Di samping itu, proses *obfuscation* diterapkan untuk meningkatkan keamanan *source code*, serta dilakukan standarisasi penamaan dan penyesuaian struktur direktori demi mempermudah proses debugging dan pemeliharaan proyek.

3. Perbaikan dan Penyempurnaan Sistem yang Telah Ada

Selama proses pengembangan, dilakukan identifikasi dan perbaikan *bug*, optimalisasi tampilan dan alur sistem berdasarkan standar UI/UX, serta penyelarasan kode dengan dokumentasi teknis yang berlaku untuk menjamin konsistensi dan keterbacaan jangka panjang.

3.3 Uraian Pelaksanaan Magang

Berikut pelaksanaan kerja magang yang dilakukan di PT Dana Kini Indonesia diuraikan pada Tabel 3.1.

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama magang

Minggu Ke -	Pekerjaan yang dilakukan
1	Pengenalan lingkungan kerja, mengerjakan tes <i>intern</i> , serta mempelajari dasar Java Spring Boot.
2	Melanjutkan tes <i>intern</i> , serta mempelajari konsep <i>MVC</i> pada Java Spring Boot.
3	Mempresentasikan hasil tes <i>intern</i> dan melakukan revisi.
4	Melakukan <i>upgrade dependencies</i> dan penyesuaian tampilan web internal.
5	Melanjutkan <i>upgrade dependencies</i> , pembuatan halaman Master Data Limit, serta integrasi dengan API internal perusahaan.
6	Pembuatan tampilan halaman Pembaruan Master Data dan Pembaruan Master Data Limit.
7	Melakukan integrasi API untuk halaman Pembaruan Master Data dan Pembaruan Master Data Limit.
Lanjut pada halaman berikutnya	

Tabel 3.1 Pekerjaan yang dilakukan tiap minggu selama magang (lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
8	Melakukan revisi <i>flow</i> halaman Master Data Limit.
9	Pengembangan tampilan halaman Autentikasi pada <i>website</i> Kini Usaha.
10	Melanjutkan pengembangan halaman Autentikasi pada <i>website</i> Kini Usaha.
11	Pengembangan tampilan halaman Dashboard, Data Pokok, dan Pembiayaan pada <i>website</i> Kini Usaha serta integrasi API untuk halaman Autentikasi.
12	Melanjutkan proses integrasi API untuk halaman Autentikasi.
13	Melanjutkan proses integrasi API untuk halaman Autentikasi dan perbaikan <i>bug</i> .
14	Pengembangan tampilan halaman Inquiry pada <i>website</i> internal perusahaan.
15	Melanjutkan pengembangan tampilan halaman Inquiry dan Komite Kredit pada <i>website</i> internal perusahaan.
16	Integrasi API untuk halaman Inquiry pada <i>website</i> internal perusahaan.
17	Integrasi API untuk halaman Inquiry pada <i>website</i> internal perusahaan dan otomatisasi sistem.
18	Integrasi API untuk halaman Komite Kredit pada <i>website</i> internal perusahaan.

Pada minggu pertama magang, dilakukan pengenalan lingkungan kerja oleh senior serta pemahaman awal terhadap *tech stack* yang digunakan perusahaan, yaitu PHP Laravel dan Java Spring Boot. Selama minggu pertama hingga minggu ketiga, diberikan masa pembelajaran untuk mempelajari *tech stack* dan mengerjakan tes *intern* berupa pembuatan tampilan halaman menggunakan Java Spring Boot untuk *website* internal perusahaan, termasuk melakukan integrasi API, serta mempresentasikan hasil tes kepada senior sebagai bagian dari evaluasi awal. Pada minggu keempat hingga minggu kelima, diberikan penugasan utama berupa pengembangan halaman Master Data Limit beserta proses integrasi API, serta tugas sampingan berupa *upgrade dependencies* dan *plugin* yang telah *outdated* pada *website* internal perusahaan. Selain itu, terdapat juga kegiatan *workshop* IT yang berlangsung selama dua hari.

Pada minggu keenam hingga kedelapan, dilakukan pembuatan halaman Pembaruan Master Data dan Pembaruan Master Data Limit disertai dengan integrasi API, bersamaan dengan pengerjaan tugas sampingan. Selanjutnya, pada minggu kesembilan hingga minggu ketigabelas, dilakukan pada pengembangan tampilan sejumlah halaman pada *website* Kini Usaha, seperti halaman Login dan Register, Dashboard, dan Sejarah Pembiayaan, serta integrasi API untuk halaman Ubah Kata Sandi dan Lupa Kata Sandi. Pengembangan *website* Kini Usaha ini dilaksanakan melalui empat siklus *sprint* selama delapan minggu, dengan pelaksanaan *sprint review* secara berkala setiap dua hari guna mengevaluasi progres, mengidentifikasi kendala, serta memastikan kesesuaian implementasi terhadap kebutuhan fungsional yang telah ditentukan.

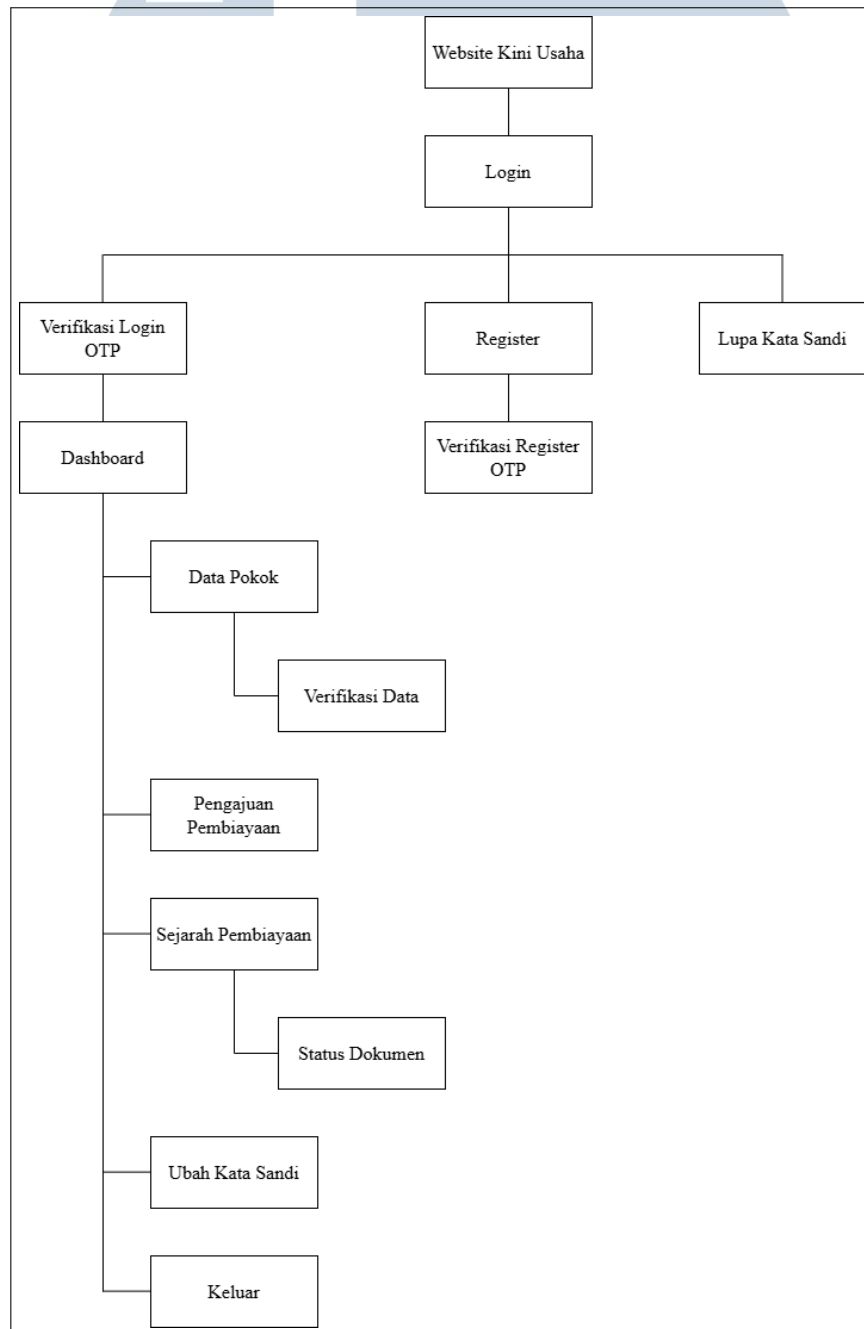
Pada tahap ini juga dilakukan otomatisasi proses penamaan ulang file *.css* menggunakan Gulp sebagai mekanisme *cache busting*, serta refaktorisasi kode dan penyesuaian konvensi penamaan pada file yang sudah ada, mencakup elemen seperti fungsi, *id*, *name*, *class*, dan *route*, agar sesuai dengan standar konvensi penamaan terbaru [8]. Pada minggu keempatbelas hingga minggu keenambelas, dilakukan pembuatan tampilan halaman Komite Kredit, serta pengembangan tampilan halaman dan integrasi API untuk halaman Inquiry pada *website* internal perusahaan yang memiliki keterkaitan langsung dengan alur proses dan data pada *website* Kini Usaha. Selain itu, dilakukan integrasi API untuk halaman Sejarah Pembiayaan dan halaman Verifikasi Data Direksi via layanan *e-sign* pada *website* Kini Usaha.

Pada minggu ketujuhbelas, dilakukan pengembangan lanjutan tampilan dan integrasi API untuk halaman Inquiry pada *website* internal perusahaan. Selain itu, dilakukan integrasi API untuk halaman Status Dokumen pada *website* Kini Usaha, serta penambahan fitur otomatisasi proses penamaan ulang file *.js* menggunakan Gulp sebagai mekanisme *cache busting*, dan penerapan proses *obfuscation* menggunakan *plugin* JavaScript Obfuscator untuk meningkatkan keamanan *source code* [9], [10]. Selanjutnya, pada minggu kedelapanbelas, dilakukan integrasi API untuk halaman Komite Kredit pada *website* internal perusahaan dan *fixing bugs* untuk halaman Inquiry.

3.3.1 Perancangan

A Sitemap

Berikut merupakan *Sitemap* dari website Kini Usaha yang telah dikembangkan, dapat dilihat pada Gambar 3.1.



Gambar 3.1. Sitemap website Kini Usaha

Sitemap ini menggambarkan struktur navigasi halaman bagi pengguna selama mengakses sistem. Halaman utama yang pertama kali diakses adalah halaman Login, yang merupakan pintu masuk ke dalam sistem autentikasi. Pada halaman ini, pengguna dapat memilih untuk melakukan login dengan kredensial yang dimiliki, melakukan proses registrasi jika belum memiliki akun, atau mengakses halaman Lupa Kata Sandi apabila mengalami kendala saat mengisi kata sandi. Setelah proses login, pengguna akan diarahkan ke halaman Verifikasi Login OTP untuk memastikan keamanan melalui autentikasi dua faktor. Sementara itu, pengguna baru yang melakukan registrasi juga harus melalui proses Verifikasi Register OTP guna mengaktifkan akun mereka.

Setelah berhasil melalui proses autentikasi, pengguna akan masuk ke dalam halaman Dashboard sebagai halaman utama yang berisi berbagai halaman inti dari sistem. Di dalam halaman Dashboard, terdapat beberapa halaman penting, seperti halaman Data Pokok, yang digunakan untuk mengisi informasi identitas pribadi dan usaha pengguna. Selain itu, terdapat halaman Pengajuan Pembiayaan untuk mengajukan pinjaman, serta halaman Sejarah Pembiayaan yang mencatat seluruh riwayat pengajuan yang telah dilakukan. Pada halaman Sejarah Pembiayaan, pengguna juga dapat mengakses halaman Status Dokumen untuk melihat kelengkapan dokumen dari masing-masing pengajuan.

Halaman tambahan lainnya yang tersedia dalam Dashboard adalah halaman Ubah Kata Sandi, yang memungkinkan pengguna untuk memperbarui kredensial keamanannya, serta halaman Keluar (*logout*) untuk mengakhiri sesi pengguna. Dengan struktur *sitemap* ini, *website* Kini Usaha dirancang agar memfasilitasi alur navigasi pengguna secara sistematis, intuitif, dan efisien, sekaligus mendukung kelancaran proses pembiayaan yang ditawarkan kepada pelaku usaha.

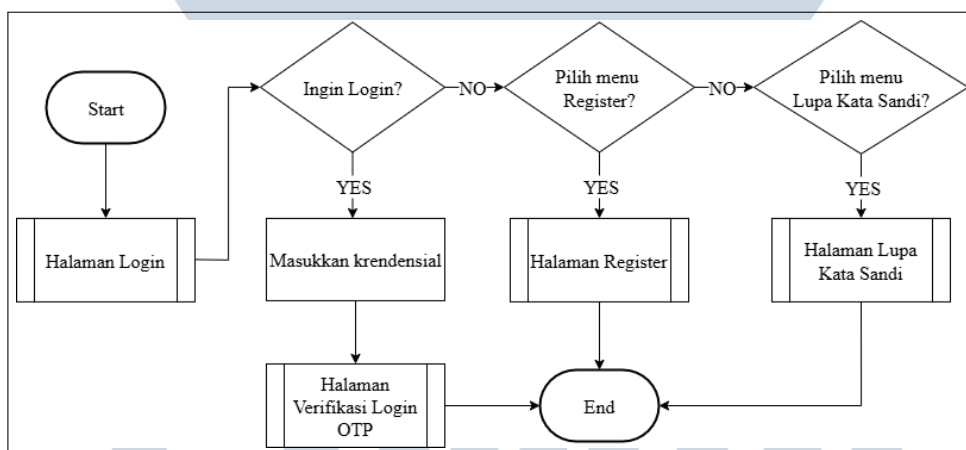
U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

B Flowchart

Berikut merupakan *flowchart* yang menyajikan representasi visual dari alur navigasi pengguna pada *website* Kini Usaha. Visualisasi ini disusun untuk menggambarkan struktur halaman dan fitur-fitur utama secara menyeluruh, sehingga mempermudah pemahaman terhadap mekanisme akses layanan. Selain itu, *flowchart* ini juga berfungsi sebagai panduan dalam memastikan keterjangkauan dan kemudahan penggunaan setiap elemen sistem oleh pengguna akhir.

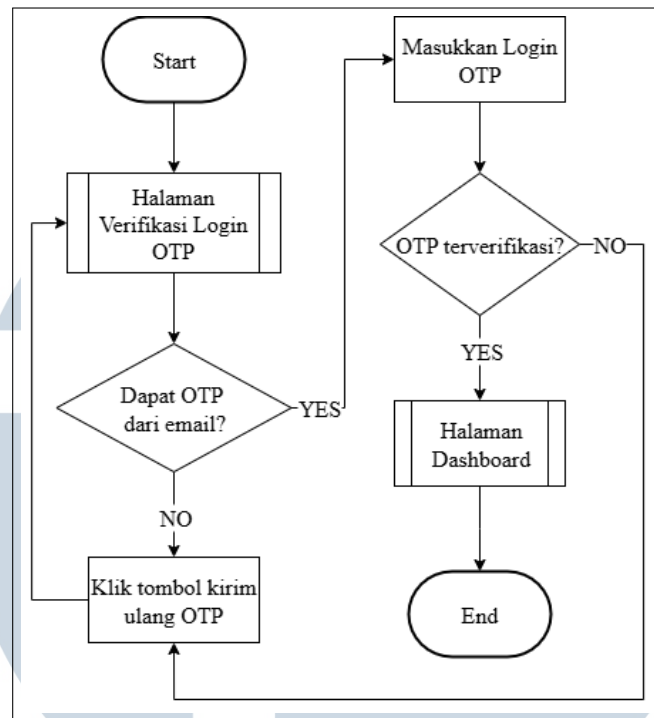
1. Flowchart Halaman Login dan Verifikasi Login OTP

Berikut ini merupakan *flowchart* dari halaman Login yang telah dikembangkan, seperti yang ditampilkan pada Gambar 3.2. Halaman Login merupakan halaman awal yang diakses oleh pengguna saat membuka *website* Kini Usaha untuk melakukan login, registrasi, pemulihan kata sandi.



Gambar 3.2. Flowchart halaman Login

Jika pengguna memilih untuk melakukan proses login, pengguna akan diminta untuk mengisi kredensial berupa *email* dan kata sandi. Kemudian, pengguna akan diarahkan ke halaman verifikasi OTP Gambar 3.3 yang dikirim melalui *email*. Apabila kode OTP berhasil diterima, pengguna dapat memasukkannya ke dalam sistem. Setelah berhasil diverifikasi, pengguna akan diarahkan menuju halaman Dashboard. Jika kode OTP tidak diterima atau salah, tersedia opsi “Kirim Ulang” untuk meminta kode baru hingga proses verifikasi berhasil dan autentikasi dinyatakan selesai.

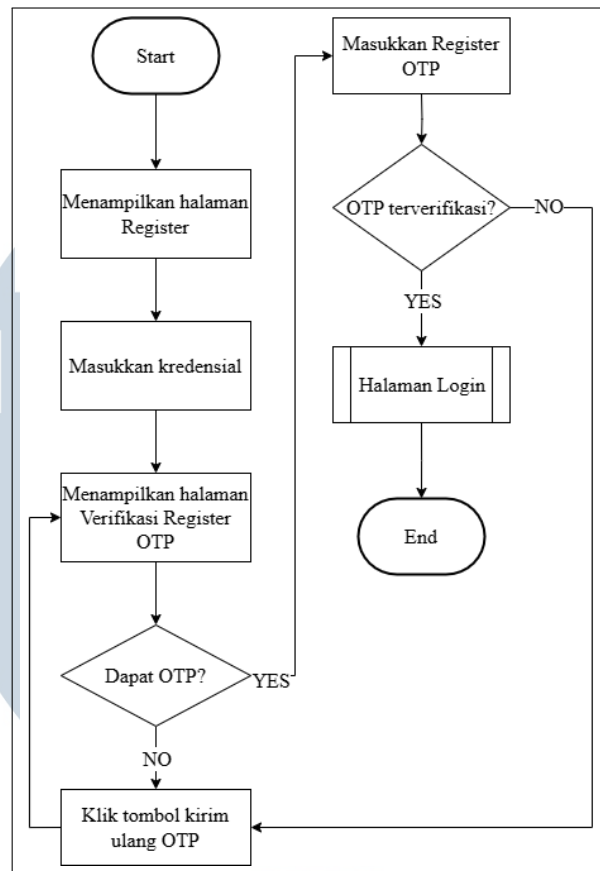


Gambar 3.3. Flowchart halaman Verifikasi Login OTP

2. Flowchart Halaman Register

Dari halaman Login, pengguna dapat memilih opsi untuk melakukan proses pendaftaran akun (Register). Alur proses ini dapat dilihat pada Gambar 3.4. Pada tahap ini, pengguna perlu untuk mengisi informasi pendaftaran berupa *email*, nomor *handphone*, dan kata sandi. Selanjutnya, sistem secara otomatis mengirimkan kode OTP ke *email* yang didaftarkan yang harus divalidasi agar proses verifikasi selesai. Jika kode OTP belum diterima, pengguna dapat mengakses fitur “Kirim Ulang”. Setelah berhasil diverifikasi, pengguna akan diarahkan kembali ke halaman Login untuk mulai masuk ke sistem menggunakan akun yang baru dibuat.

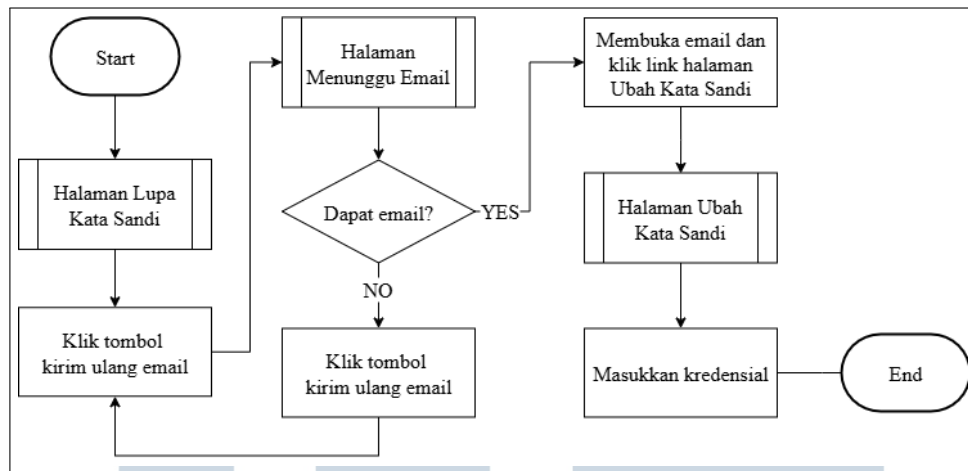
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.4. Flowchart halaman Register

3. Flowchart Halaman Lupa Kata Sandi

Jika pengguna mengalami kendala dalam mengakses akun dan memilih opsi Lupa Kata Sandi pada halaman Login, maka sistem akan mengarahkan pengguna ke proses pemulihan kata sandi melalui alur yang ditunjukkan pada Gambar 3.5. Pada halaman ini, pengguna diminta untuk memasukkan *email* yang terdaftar, lalu sistem akan mengirimkan *link* pemulihan ke alamat tersebut. Setelah mengakses *link* tersebut, pengguna diarahkan ke halaman pengaturan ulang kata sandi, di mana mereka diminta untuk membuat sandi baru serta mengisi CAPTCHA sebagai verifikasi. Jika proses berhasil, pengguna dapat kembali ke halaman Login untuk masuk menggunakan sandi yang baru diperbarui.

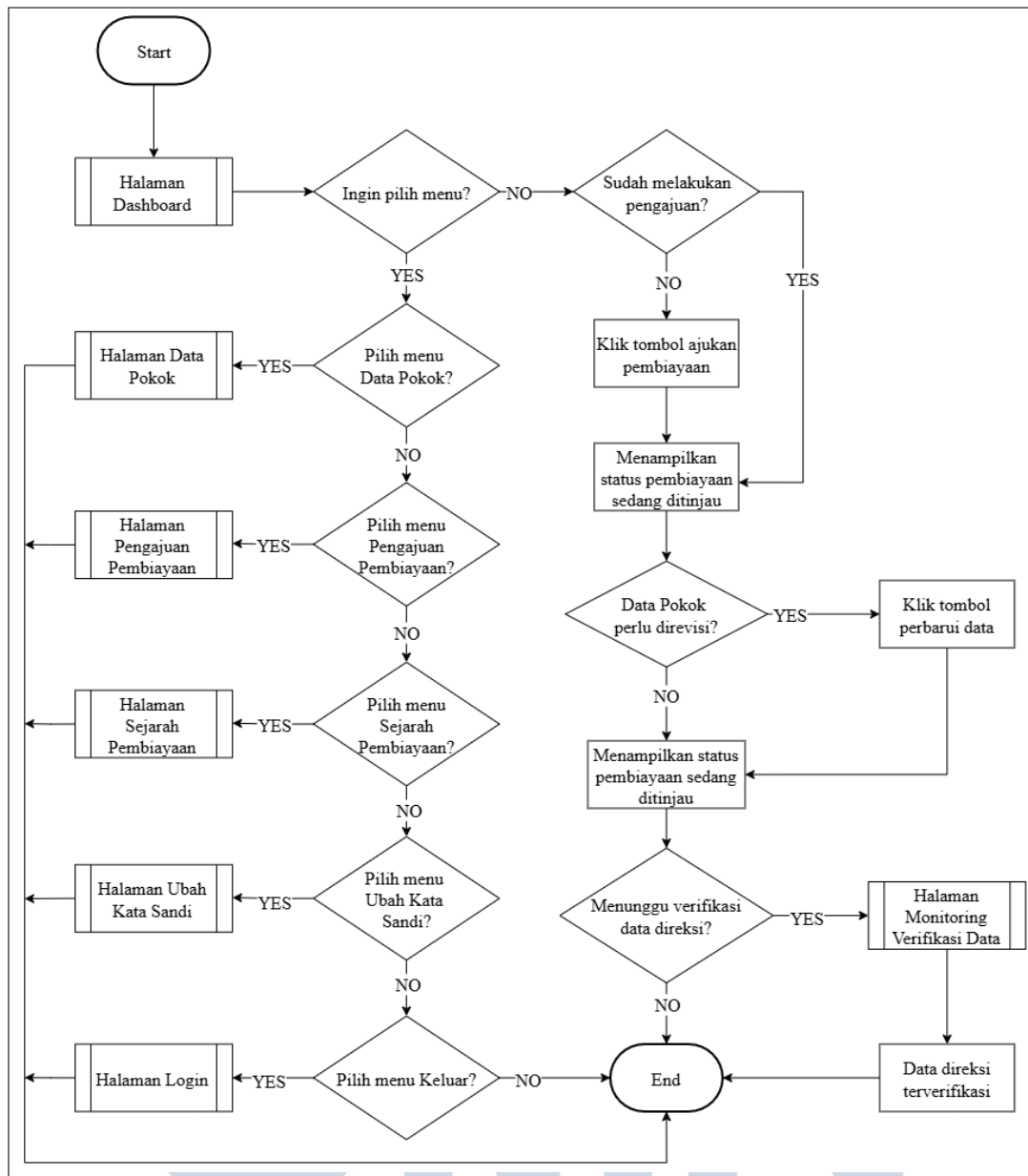


Gambar 3.5. Flowchart halaman Lupa Kata Sandi

4. Flowchart Halaman Dashboard

Halaman Dashboard merupakan titik awal yang diakses pengguna setelah berhasil masuk ke dalam sistem. Seperti ditunjukkan pada Gambar 3.6, apabila pengguna belum melakukan pengajuan, sistem akan mengarahkan pengguna untuk mengajukan pembiayaan melalui tombol "Ajukan Pembiayaan". Namun, jika pengajuan telah dilakukan, sistem akan menampilkan status bahwa pengajuan pembiayaan sedang dalam proses peninjauan.

Selanjutnya, sistem akan memeriksa apakah data pokok perlu direvisi. Jika ya, pengguna akan diarahkan ke halaman Data Pokok untuk melakukan perbaikan. Apabila tidak ada revisi yang diperlukan, proses dilanjutkan ke tahap verifikasi oleh pihak direksi. Jika verifikasi masih dalam proses, pengguna dapat memantau statusnya melalui halaman Monitoring Verifikasi Data. Setelah data dinyatakan terverifikasi, proses dianggap selesai dan pengguna dapat mengajukan pembiayaan baru apabila diperlukan.



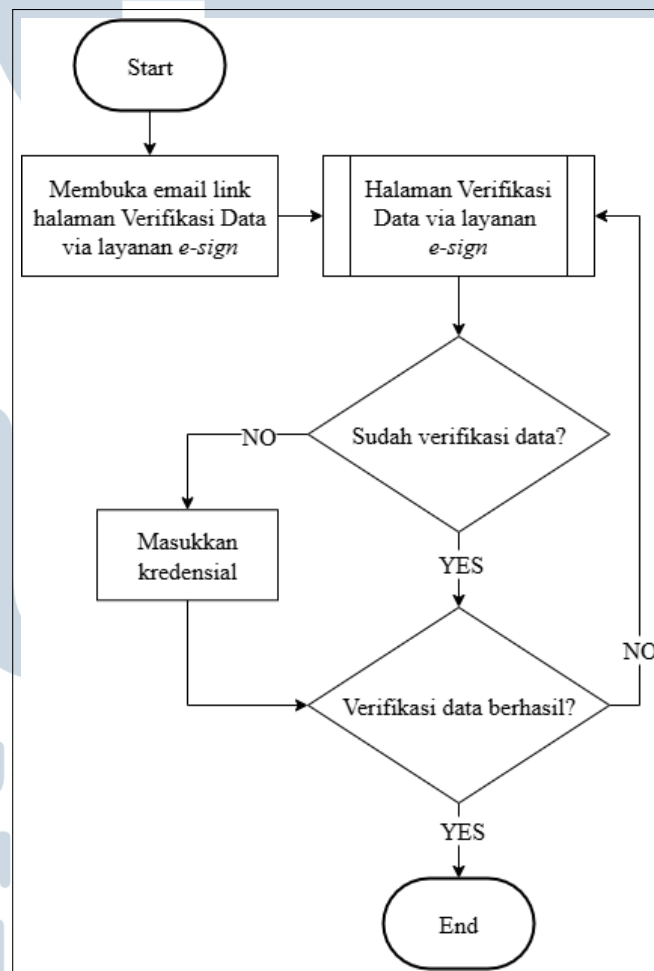
Gambar 3.6. Flowchart halaman Dashboard

UNIVERSITAS
MULTIMEDIA
NUSANTARA

5. Flowchart Halaman Verifikasi Data via Layanan E-Sign

Flowchart ini merupakan kelanjutan dari alur pada halaman Dashboard, khususnya saat data pokok telah lengkap dan pembiayaan sedang ditinjau, maka status berubah menjadi menunggu verifikasi data direksi. Terlihat pada Gambar 3.7, sistem mengirimkan *email* berisi *link* menuju halaman Verifikasi Data via layanan *e-sign*. Setelah *link* diakses, pengguna akan diarahkan ke halaman Verifikasi Data via layanan *e-sign* untuk mengisi kredensial.

Sistem kemudian memeriksa apakah proses verifikasi telah dilakukan dan berhasil. Jika belum, pengguna diminta untuk melengkapi data terlebih dahulu. Jika verifikasi berhasil, maka proses dianggap selesai dan sistem akan memperbarui status data menjadi “terverifikasi”.

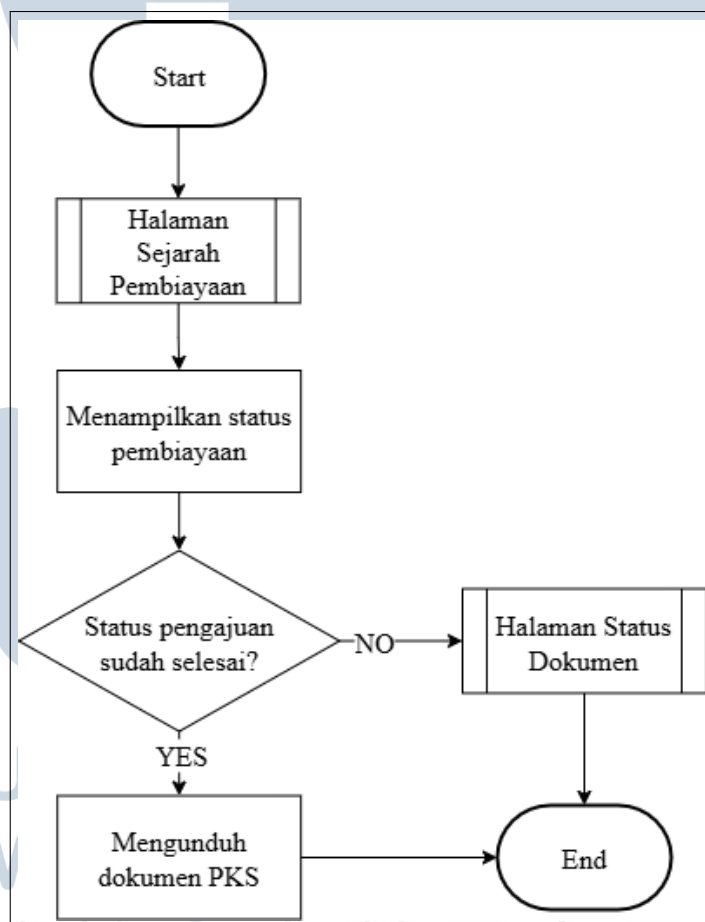


Gambar 3.7. Flowchart halaman Verifikasi Data via layanan e-sign

6. Flowchart Halaman Sejarah Pembiayaan

Halaman Sejarah Pembiayaan menampilkan status akhir dari pengajuan yang telah disetujui. Seperti ditunjukkan pada Gambar 3.8, alur dimulai dari halaman Sejarah Pembiayaan, di mana sistem akan menampilkan status pembiayaan. Jika status pengajuan telah selesai, maka pengguna dapat langsung mengunduh dokumen Perjanjian Kerja Sama (PKS).

Sebaliknya, jika belum selesai, pengguna akan diarahkan ke halaman Status Dokumen. Pada tahap ini, sistem memverifikasi apakah dokumen telah ditandatangani secara digital. Jika belum, pengguna dapat menekan tombol “Kirim Ulang” untuk memicu pengiriman *link* halaman Verifikasi Data Direksi kepada direksi melalui *email*.



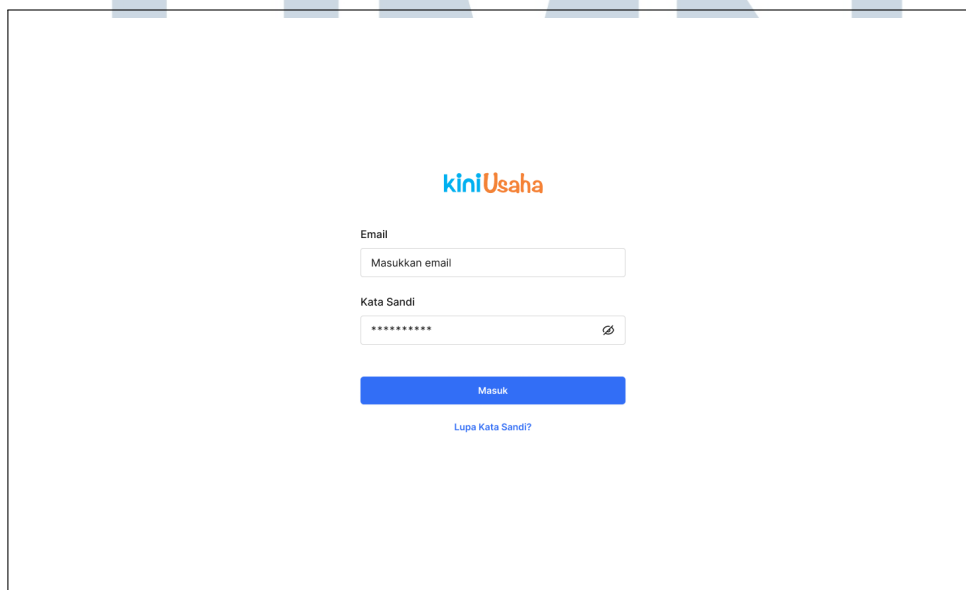
Gambar 3.8. Flowchart halaman Sejarah Pembiayaan

C Wireframe

Pada tahap pengembangan, perancangan *wireframe* disusun oleh tim UI/UX sebagai panduan awal dalam membentuk tampilan antarmuka pengguna. *Wireframe* ini menyajikan gambaran umum mengenai susunan elemen dan struktur halaman pada *website* Kini Usaha, yang berperan penting dalam merancang alur interaksi pengguna serta menetapkan komponen fungsional pada setiap bagian halaman. Adanya rancangan ini memungkinkan proses pengembangan *website* dilakukan dengan pendekatan yang lebih sistematis, efisien, dan selaras dengan kebutuhan pengguna.

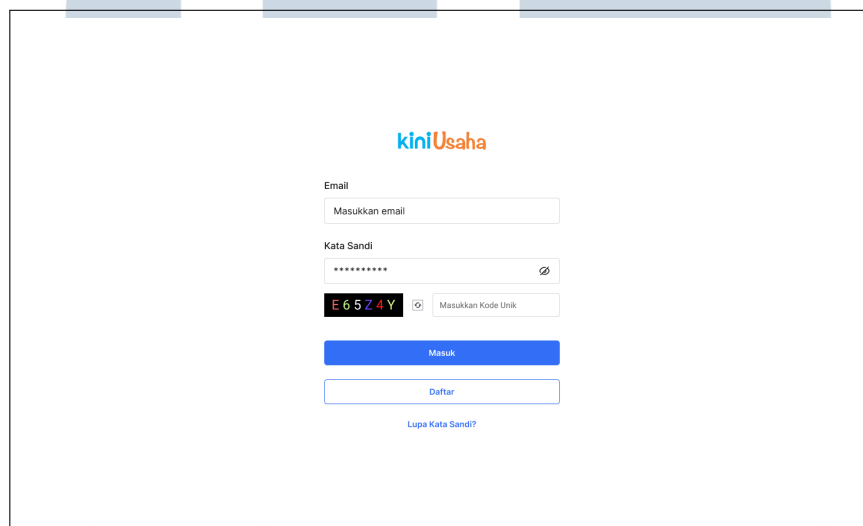
1. Wireframe Halaman Login

Wireframe website Kini Usaha sebelum pengembangan untuk halaman Login ditampilkan pada Gambar 3.9. Pada halaman ini, terdapat formulir yang terdiri dari dua kolom input, yaitu alamat *email* dan kata sandi. Pengguna diminta untuk mengisi data tersebut guna melakukan proses autentikasi. Apabila data yang dimasukkan sesuai, pengguna akan diarahkan ke halaman Login OTP Email. Sebaliknya, jika proses login gagal, maka akan ditampilkan pesan kesalahan yang sesuai. Selain itu, halaman ini juga menyediakan fitur lupa kata sandi untuk membantu pengguna melakukan pemulihan akun apabila diperlukan.

The image shows a wireframe for a login page. At the top center is the logo 'kiniUsaha' in blue and orange. Below it, the word 'Email' is followed by a text input field with the placeholder 'Masukkan email'. Underneath that, the text 'Kata Sandi' is followed by a password input field with a masked password '*****' and an eye icon to toggle visibility. A blue button labeled 'Masuk' is positioned below the password field. At the bottom, there is a link that says 'Lupa Kata Sandi?'.

Gambar 3.9. Wireframe halaman Login sebelum pengembangan

Setelah dilakukan pengembangan, *wireframe* halaman Login diperbarui, seperti yang ditampilkan pada 3.10. Pada versi terbaru ini, ditambahkan satu kolom input untuk CAPTCHA guna meningkatkan keamanan proses autentikasi, serta tombol daftar yang mengarahkan pengguna ke halaman Register bagi yang belum memiliki akun. Dengan demikian, formulir login kini terdiri dari tiga kolom input, antara lain alamat *email*, kata sandi, dan CAPTCHA. Meskipun terdapat penambahan input, alur autentikasi secara umum tidak mengalami perubahan, di mana pengguna tetap diarahkan ke halaman OTP Email setelah proses login berhasil.



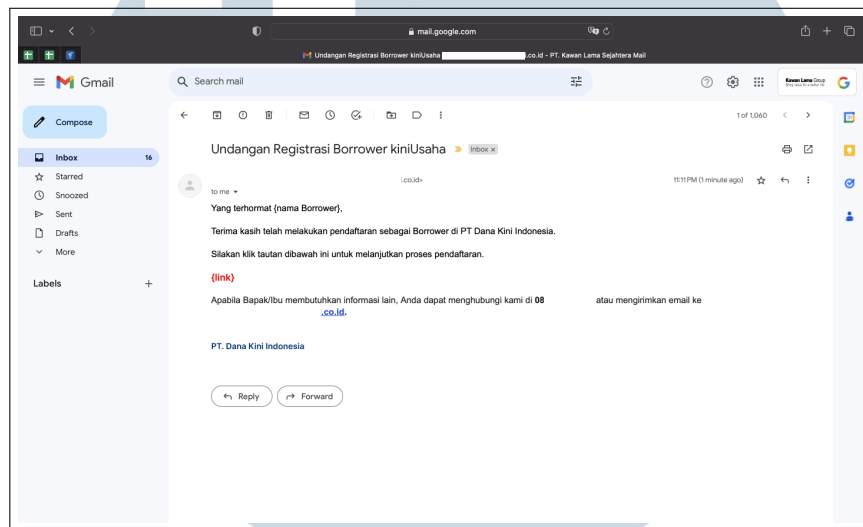
The wireframe shows a login form for 'kiniUsaha'. It includes an 'Email' section with a placeholder 'Masukkan email', a 'Kata Sandi' (Password) section with a placeholder '*****' and an eye icon, and a CAPTCHA section with a placeholder 'E 6 5 Z 4 Y' and a placeholder 'Masukkan Kode Unik'. Below these are two buttons: 'Masuk' (Login) and 'Daftar' (Register). A link 'Lupa Kata Sandi?' is also present.

Gambar 3.10. Wireframe halaman Login setelah pengembangan

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

2. Wireframe Halaman Register

Wireframe website Register Kini Usaha sebelum pengembangan ditampilkan pada Gambar 3.11. Pada tahap ini, akses ke halaman Register tidak disediakan secara langsung dari *website*, melainkan hanya dapat dibuka melalui *link* khusus yang dikirimkan ke alamat *email* calon pengguna.



Gambar 3.11. Wireframe halaman Register bagian 1 sebelum pengembangan

Tampilan halaman Register ditampilkan pada Gambar 3.12. Pada halaman Register, memuat formulir dengan tiga kolom input, yaitu alamat *email*, kata sandi, dan konfirmasi kata sandi.

Gambar 3.12. Wireframe halaman Register bagian 2 sebelum pengembangan

Setelah pengguna mengisi seluruh data dengan benar dan valid, sistem akan mengarahkan pengguna ke halaman pengisian data lanjutan, yang terdiri dari empat tab isian informasi, yaitu Data Personal, Data Alamat, Data Usaha, dan Upload, yang ditunjukkan pada Gambar 3.13. Jika pengguna sudah berhasil melakukan registrasi, pengguna akan langsung diarahkan ke halaman Login.

Gambar 3.13. Wireframe halaman Register bagian 3 sebelum pengembangan

Wireframe website Kini Usaha setelah pengembangan untuk halaman Register ditampilkan pada Gambar 3.14. Pada halaman ini, terdapat formulir yang terdiri dari empat kolom input, yaitu alamat *email*, nomor *handphone*, kata sandi, dan konfirmasi kata sandi.

< Kembali ke Login

kiniUsaha

Email *

Nomor Handphone *

Kata Sandi *

Konfirmasi Kata Sandi *

Lanjutkan Pendaftaran

Gambar 3.14. Wireframe halaman Register bagian 1 setelah pengembangan

Setelah data yang dimasukkan valid, pengguna akan diarahkan ke halaman verifikasi OTP melalui *email* dan SMS seperti yang ditunjukkan pada Gambar 3.15. Jika verifikasi OTP berhasil, maka proses registrasi selesai dan pengguna dapat melanjutkan ke halaman Login.

< Kembali

kiniUsaha

Masukkan kode OTP yang telah dikirimkan melalui email ke
 m*****x@gmail.com

 Tidak menerima kode? 01:39

Masukkan kode OTP yang telah dikirimkan melalui SMS ke 08*****222

 Tidak menerima kode? Kirim Ulang

Verifikasi

Gambar 3.15. Wireframe halaman Register bagian 2 setelah pengembangan

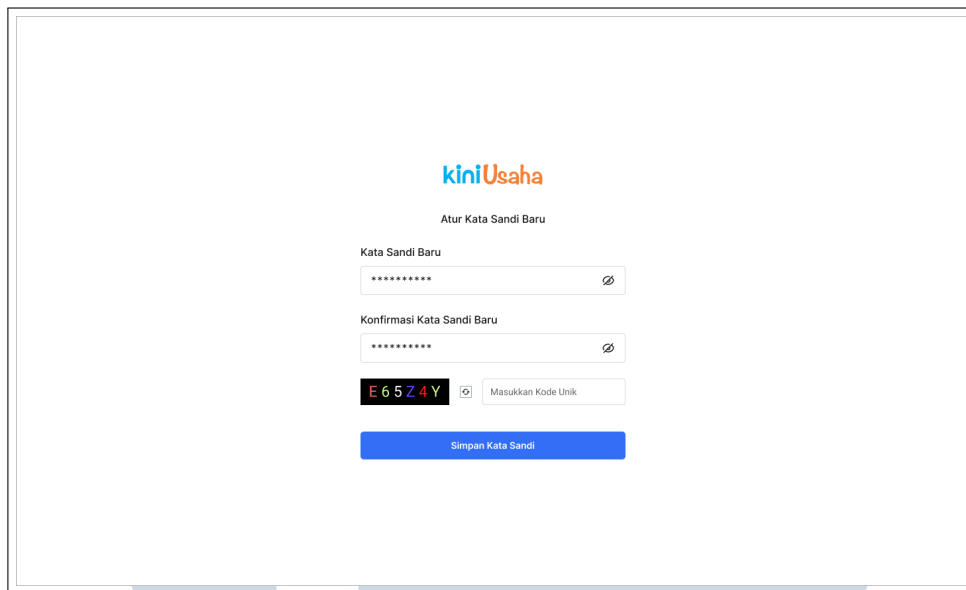
3. Wireframe Halaman Lupa Kata Sandi

Wireframe website Kini Usaha sebelum pengembangan untuk halaman Lupa Kata Sandi ditampilkan pada Gambar 3.16. Pada alur ini, pengguna diminta untuk memasukkan alamat *email* yang telah terdaftar. Selanjutnya, sistem akan mengirimkan *link* untuk mengatur ulang kata sandi melalui *email* tersebut. Setelah pengguna mengakses *link* tersebut, akan ditampilkan halaman Kata Sandi yang berisi formulir dengan dua kolom input, yaitu kata sandi baru dan konfirmasi kata sandi baru. Setelah proses pengaturan ulang kata sandi selesai, pengguna akan diarahkan kembali ke halaman Login.

The wireframe shows a web page for 'kiniUsaha' with the title 'Atur Kata Sandi Baru'. It contains two input fields for passwords, each with a strength indicator icon on the right. The first field is labeled 'Kata Sandi Baru' and the second is labeled 'Konfirmasi Kata Sandi Baru'. Below the fields is a blue button labeled 'Simpan Kata Sandi'.

Gambar 3.16. Wireframe halaman Forgot Password sebelum pengembangan

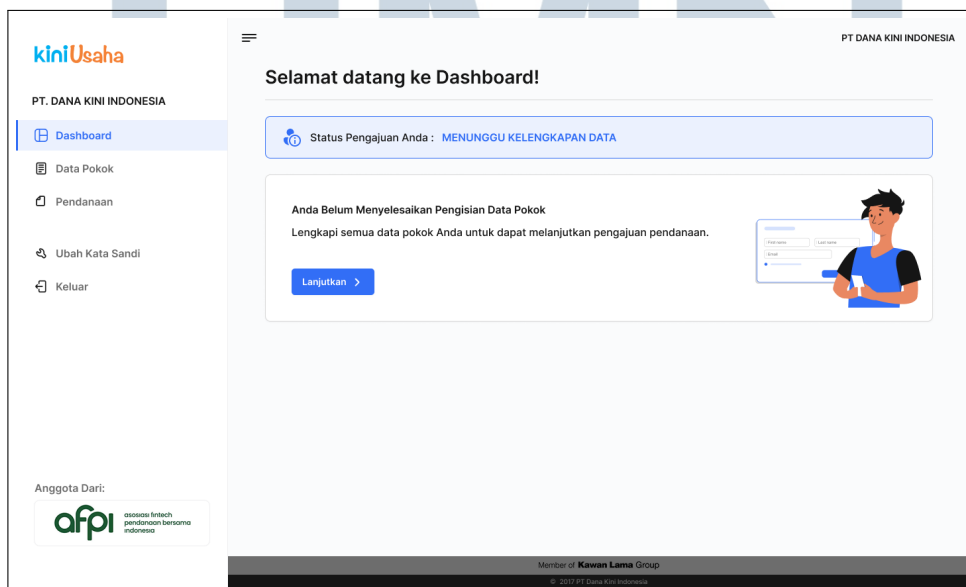
Sementara itu, tampilan halaman Lupa Kata Sandi setelah pengembangan ditampilkan pada Gambar 3.17. Perbedaan utama pada versi ini adalah penambahan kolom input CAPTCHA sebagai mekanisme pengamanan tambahan. Alur proses halaman ini tetap mengikuti struktur sebelumnya, yakni pengguna diminta untuk memasukkan alamat *email* yang telah terdaftar, kemudian sistem akan mengirimkan *link* untuk mengatur ulang kata sandi melalui *email*. Setelah mengakses *link* tersebut, pengguna akan diarahkan ke halaman Lupa Kata Sandi yang kini dilengkapi dengan input CAPTCHA. Setelah CAPTCHA berhasil divalidasi dan kata sandi baru disimpan, pengguna akan diarahkan kembali ke halaman Login.



Gambar 3.17. Wireframe halaman Forgot Password setelah pengembangan

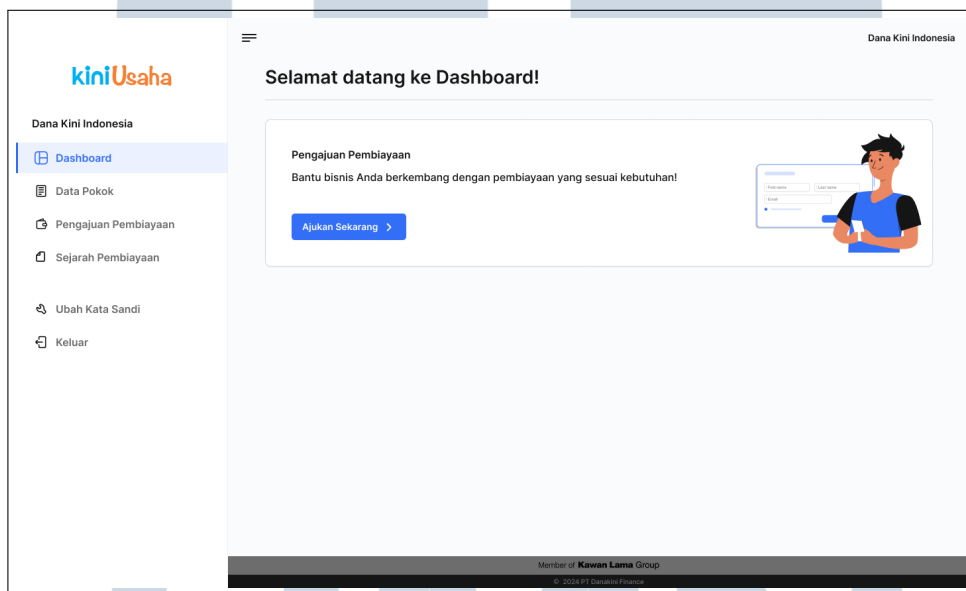
4. Wireframe Halaman Dashboard

Wireframe website Kini Usaha sebelum pengembangan untuk halaman Dashboard ditampilkan pada Gambar 3.18. Tampilan ini memuat tujuh *widget* berdasarkan status pengajuan, yaitu Processing, Dalam Peninjauan, Menunggu Perbaikan Data, Ajukan Pendanaan, Menunggu Verifikasi Data, Pendanaan Ditolak, dan Tinjau Pendanaan.



Gambar 3.18. Wireframe halaman Dashboard sebelum pengembangan

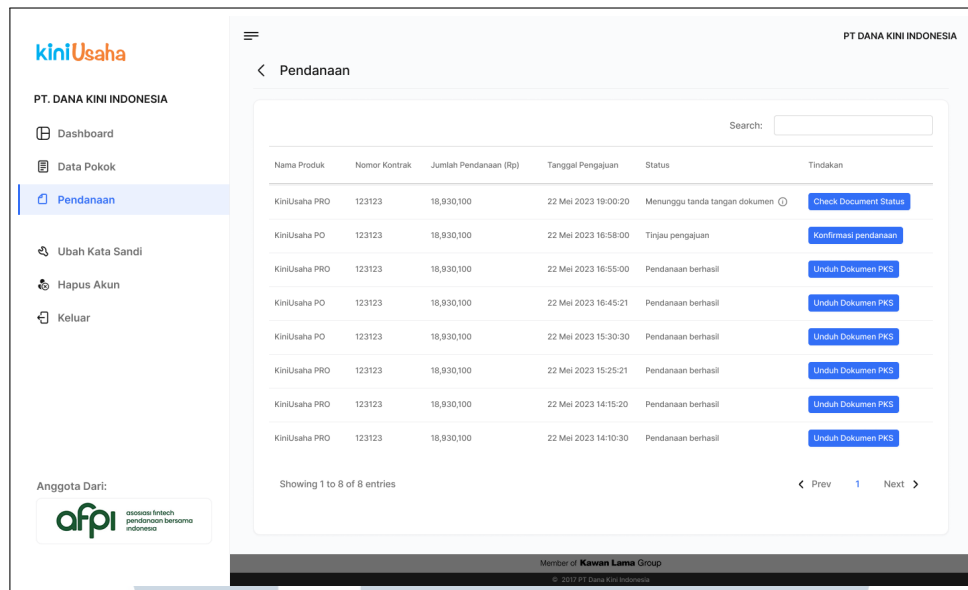
Wireframe website Kini Usaha setelah pengembangan untuk halaman Dashboard ditampilkan pada Gambar 3.19. Berbeda dengan yang sebelumnya, terdapat penyesuaian pada isi Dashboard. Jumlah widget informasi berdasarkan status pengajuan pengguna berkurang menjadi enam, yaitu Pengajuan Pembiayaan (halaman utama Dashboard), Menunggu Perbaikan Data, Processing, Menunggu Verifikasi Data, Ongoing Pengajuan, dan Pengajuan On Legal. Selain perubahan tersebut, terdapat tambahan *widget* informasi pembiayaan terakhir yang hanya menampilkan satu riwayat pembiayaan.



Gambar 3.19. Wireframe halaman Dashboard setelah pengembangan

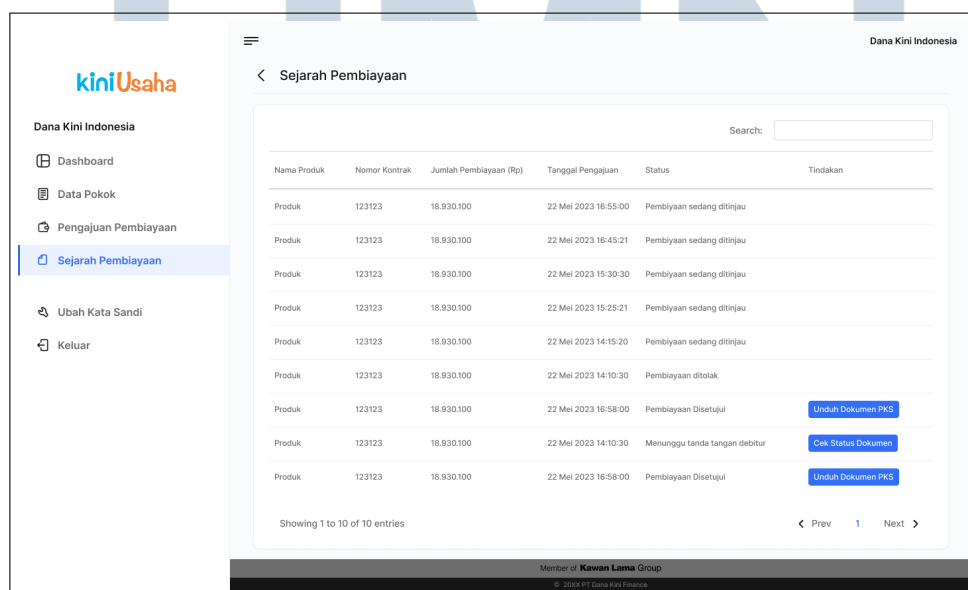
5. Wireframe Halaman Sejarah Pembiayaan

Wireframe website Kini Usaha sebelum pengembangan untuk halaman Sejarah Pembiayaan ditampilkan pada Gambar 3.20. Sebelumnya, halaman ini bernama Pendanaan, yang menampilkan informasi riwayat pengajuan pembiayaan pengguna dalam bentuk tabel. Tabel tersebut memiliki kolom *action*, yang berisi tombol untuk pengguna melakukan cek status dokumen, konfirmasi pendanaan, maupun unduh dokumen PKS.



Gambar 3.20. Wireframe halaman Sejarah Pembiayaan sebelum pengembangan

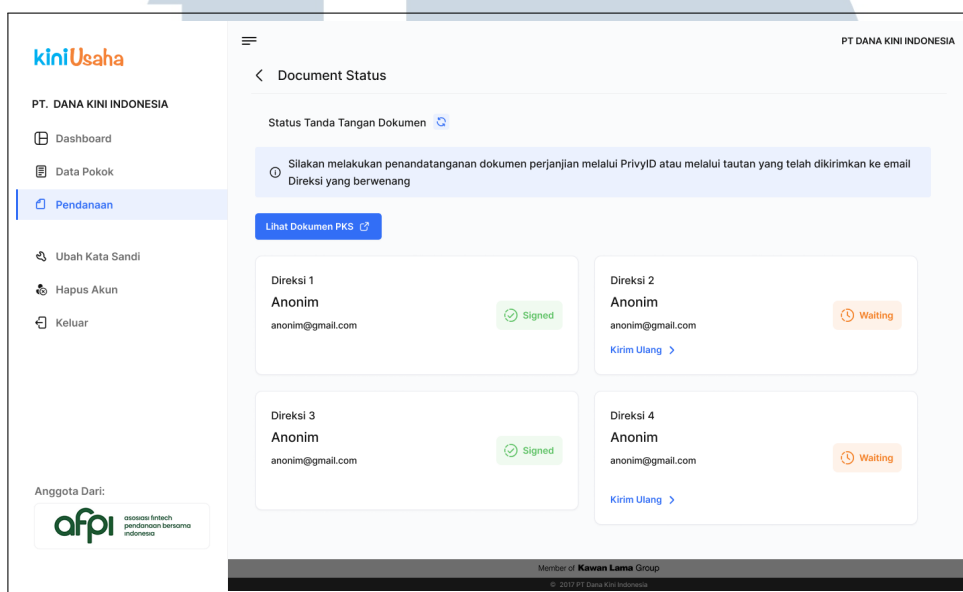
Wireframe website Kini Usaha setelah pengembangan untuk halaman Pendanaan ditampilkan pada Gambar 3.21, dengan perubahan nama menjadi Sejarah Pembiayaan untuk menyesuaikan istilah yang lebih relevan. Secara fungsi, halaman ini tetap menyajikan tabel riwayat pengajuan pembiayaan, namun kolom *action* disederhanakan menjadi hanya dua opsi, yaitu pengecekan status dokumen dan pengunduhan dokumen PKS.



Gambar 3.21. Wireframe halaman Sejarah Pembiayaan setelah telah pengembangan

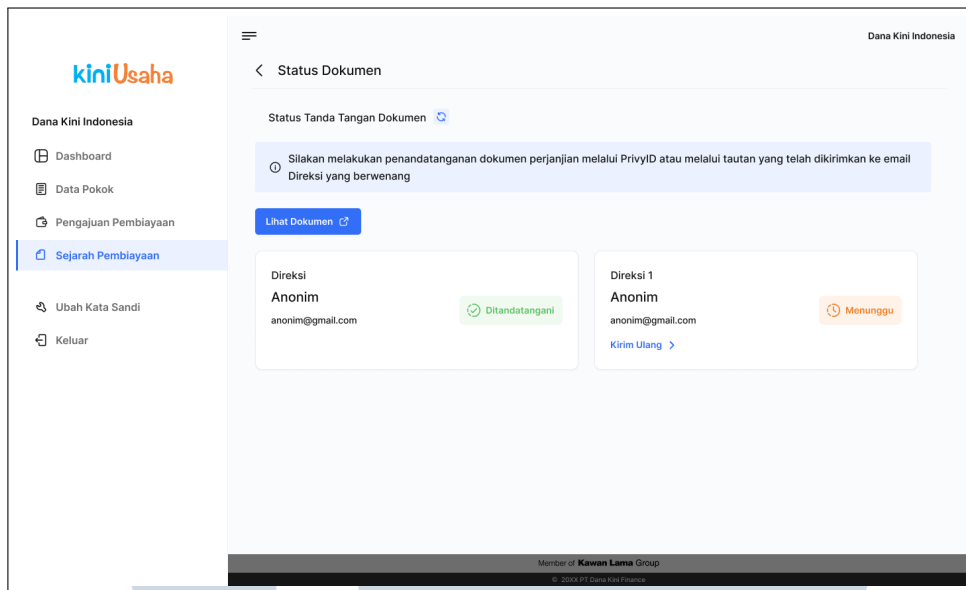
6. Wireframe Halaman Status Dokumen

Wireframe website Kini Usaha sebelum pengembangan untuk halaman Status Dokumen ditampilkan pada Gambar 3.22. Halaman ini menampilkan informasi mengenai status tanda tangan dokumen Perjanjian Kerja Sama (PKS) untuk satu direksi, yang menunjukkan apakah dokumen telah disetujui atau masih dalam proses penandatanganan.



Gambar 3.22. Wireframe halaman Status Dokumen sebelum pengembangan

Wireframe website Kini Usaha setelah pengembangan untuk halaman yang sama ditampilkan pada Gambar 3.23. Tidak terdapat perubahan dari sisi alur maupun fungsionalitas utama. Perbedaan utamanya terletak pada penyesuaian bahasa, di mana seluruh elemen dalam halaman telah disesuaikan menggunakan Bahasa Indonesia secara konsisten, termasuk istilah dan label yang digunakan.



Gambar 3.23. Wireframe halaman Status Dokumen setelah pengembangan

7. Wireframe Halaman Verifikasi Data via Layanan *E-Sign*

Wireframe website Kini Usaha sebelum pengembangan untuk halaman Verifikasi Data via layanan *e-sign* ditunjukkan pada Gambar 3.24. Halaman ini diakses melalui tautan yang dikirimkan melalui *email* kepada pihak direksi sebagai bagian dari proses penandatanganan dokumen digital. Halaman ini memungkinkan direksi untuk mengisi data kredensial, seperti ID dan password yang diperlukan untuk proses verifikasi serta penandatanganan digital dokumen.

Secara tampilan, halaman ini tidak mengalami perubahan dari versi sebelumnya. Namun, pada tahap pengembangan dilakukan integrasi dengan API yang baru untuk menangani proses verifikasi dan penandatanganan secara digital. Dengan integrasi ini, proses menjadi lebih terotomatisasi dan terjamin keamanannya, tanpa mengubah struktur maupun tampilan antar muka halaman.

kiniUsaha

Nama Direksi

Jenis Badan Hukum

Nama Badan Hukum

☐ Saya adalah Direksi yang bertanggungjawab dan menyatakan bahwa data-data yang telah diisi diatas merupakan data yang sebenarnya dan data yang valid

☐ Menyatakan setuju dan memberi kuasa kepada PT Dana Kini Indonesia untuk melakukan proses permintaan informasi perkreditan pada Lembaga Pengelola Informasi Perkreditan (LPIP) dan/atau Lembaga lain di bidang pengecekan dan verifikasi data untuk kepentingan pengajuan pendanaan ini.

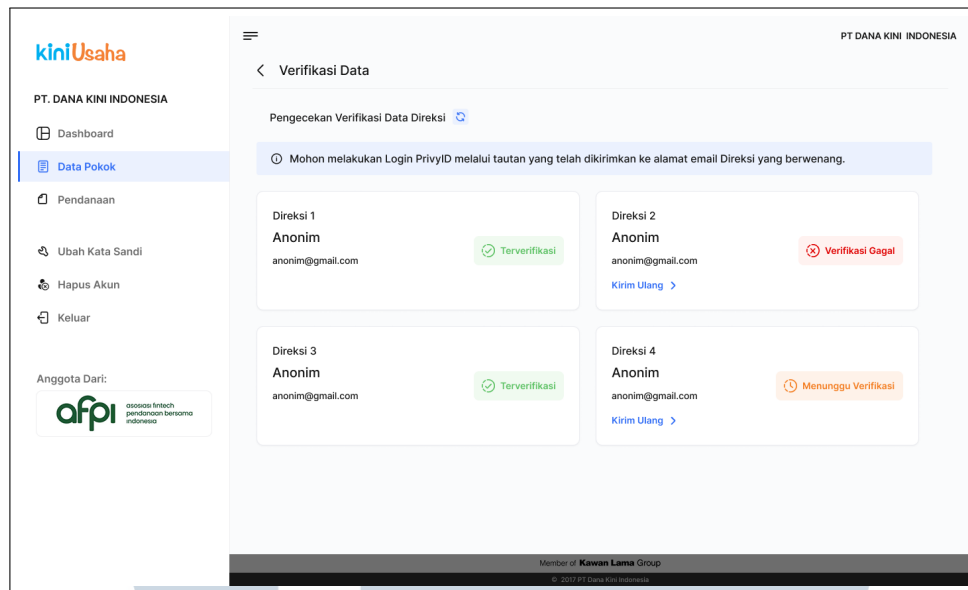
Lanjutkan

Gambar 3.24. Wireframe halaman Verifikasi Data via layanan e-sign

8. Wireframe Halaman Monitoring Verifikasi Data

Wireframe website Kini Usaha sebelum pengembangan untuk halaman Verifikasi Data ditampilkan pada Gambar 3.25. Halaman ini menampilkan status proses verifikasi data direksi, yang dapat berupa status “Terverifikasi”, “Menunggu Verifikasi”, atau “Verifikasi Gagal”. Alur awalnya dimulai dari halaman Data Pokok, di mana sistem akan mengirimkan *email* permohonan verifikasi kepada pihak terkait.

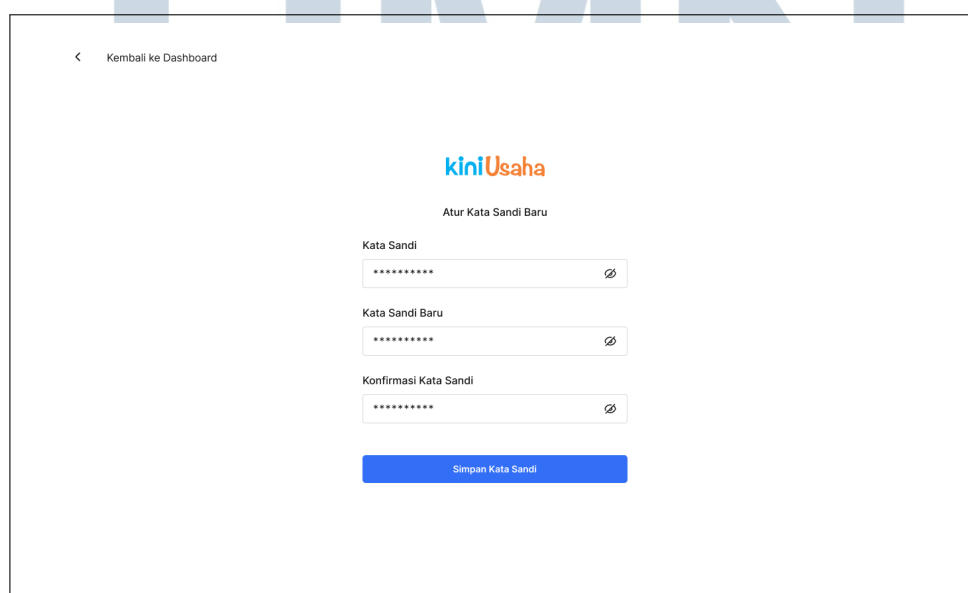
Selanjutnya, pengguna dapat mengakses halaman ini melalui Data Pokok maupun halaman Dashboard untuk memantau status verifikasi tersebut. Selain itu, tersedia tombol yang berfungsi untuk melakukan *trigger* ulang pengiriman *email* permohonan verifikasi data direksi apabila dibutuhkan. Setelah dilakukan pengembangan, tidak terdapat perubahan pada alur maupun tampilan halaman ini, namun proses verifikasi telah diintegrasikan dengan API baru untuk meningkatkan efisiensi dan keandalan sistem.



Gambar 3.25. Wireframe halaman Verifikasi Data sebelum pengembangan

9. Wireframe Halaman Ubah Kata Sandi

Wireframe halaman Ubah Kata Sandi sebelum pengembangan ditampilkan pada Gambar 3.26. Halaman ini memungkinkan pengguna mengganti kata sandi untuk menjaga keamanan akun dan dapat diakses melalui *sidebar*. Meskipun tampilan tidak berubah signifikan, integrasi API dan alur sistem diperbarui. Setelah ubah kata sandi, pengguna tetap dalam sesi aktif dan diarahkan ke halaman Dashboard.



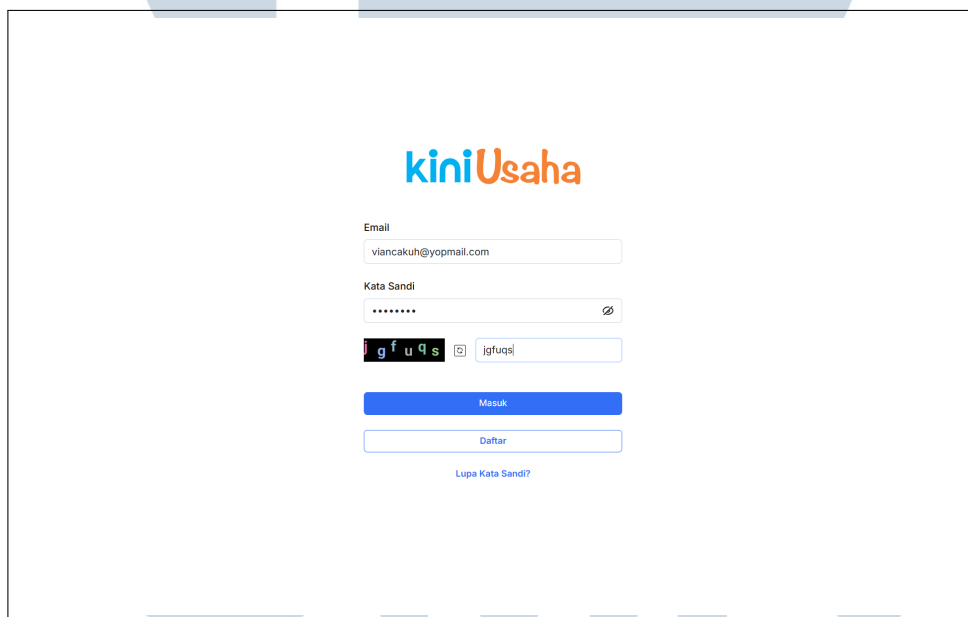
Gambar 3.26. Wireframe halaman Ubah Kata Sandi

3.3.2 Hasil Implementasi

Berikut merupakan hasil implementasi *website* Kini Usaha yang telah dikembangkan berdasarkan persyaratan dan ketentuan fungsional yang telah ditetapkan:

1. Halaman Login

Halaman ini menjadi pintu masuk utama pengguna saat membuka *website* Kini Usaha. Pengguna dapat mengaksesnya untuk masuk menggunakan kredensial akun yang telah terdaftar. Selain itu, pada halaman ini juga tersedia *link* untuk mengakses halaman Register bagi pengguna baru, serta halaman Lupa Kata Sandi bagi pengguna yang mengalami kendala dalam mengakses akun mereka, seperti yang ditunjukkan pada Gambar 3.27.



Gambar 3.27. Halaman Login

Proses autentikasi mencakup pengisian *email*, kata sandi, dan CAPTCHA. Setelah data yang diinput berhasil divalidasi, pengguna akan diarahkan ke halaman Verifikasi OTP yang dikirimkan melalui *email*, ditunjukkan pada Gambar 3.28.

kiniUsaha

Masukkan kode OTP yang telah dikirimkan melalui email ke v*****h@yopmail.com

1 2 3 4 5 6

Masuk

Tidak menerima kode? [Kirim Ulang](#)

Gambar 3.28. Halaman Verifikasi Login OTP

Jika dalam 3x percobaan OTP yang dikirimkan masih salah, pengguna akan diberikan pilihan untuk mendapatkan OTP melalui *email* atau SMS, ditunjukkan pada Gambar 3.29. Sebaliknya, jika kode OTP berhasil diverifikasi, pengguna akan langsung diarahkan ke halaman Dashboard.

kiniUsaha

Kami akan mengirimkan 6 digit kode autentikasi. Silakan pilih salah satu metode di bawah ini

☐ Email ke v*****h@yopmail.com

☐ SMS ke 08*****456

Gambar 3.29. Halaman ubah metode pengiriman kode OTP

Potongan kode untuk halaman Login ketika *trigger* OTP dapat dilihat pada Kode 3.1.

```
1 public function triggerOTP($type, $phone, $email)
2 {
3     $userOTPIdentifier = $email;
4
5     $data = [
6         'userOTPIdentifier' => $userOTPIdentifier,
7         'userOTPSendType' => $type,
8         'userOTPSendTo' => $type == 2 ? $email : $phone,
9         'userOTPStepProcess' => 'Login2FACompany',
10        'source' => 'OPS',
11    ];
12
13    $apiHelper = new ApiHelper();
14    $requestOTP = $apiHelper->post($this->gateway . $this
15    ->api, $data);
16    return $requestOTP;
17 }
```

Kode 3.1: Potongan kode halaman Login ketika trigger OTP

2. Halaman Register

Halaman ini memungkinkan pengguna baru untuk membuat akun pada sistem Kini Usaha. Pengguna diminta mengisi sejumlah data seperti alamat *email*, nomor handphone, dan kata sandi, yang kemudian divalidasi agar sesuai dengan ketentuan yang berlaku, seperti yang ditunjukkan pada Gambar 3.30.

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

Gambar 3.30. Halaman Register

Setelah pengguna berhasil mengisi data pada formulir registrasi, sistem akan mengirimkan kode OTP ke *email* dan nomor handphone yang telah didaftarkan. Selanjutnya, pengguna akan diarahkan ke halaman Verifikasi OTP, sebagaimana ditunjukkan pada Gambar 3.31. Apabila kode OTP berhasil diverifikasi, maka pengguna akan dialihkan ke halaman Login untuk dapat masuk ke dalam sistem.

Gambar 3.31. Halaman Verifikasi Register OTP

Potongan kode untuk halaman Register ketika *trigger* OTP dapat dilihat pada Kode 3.2.

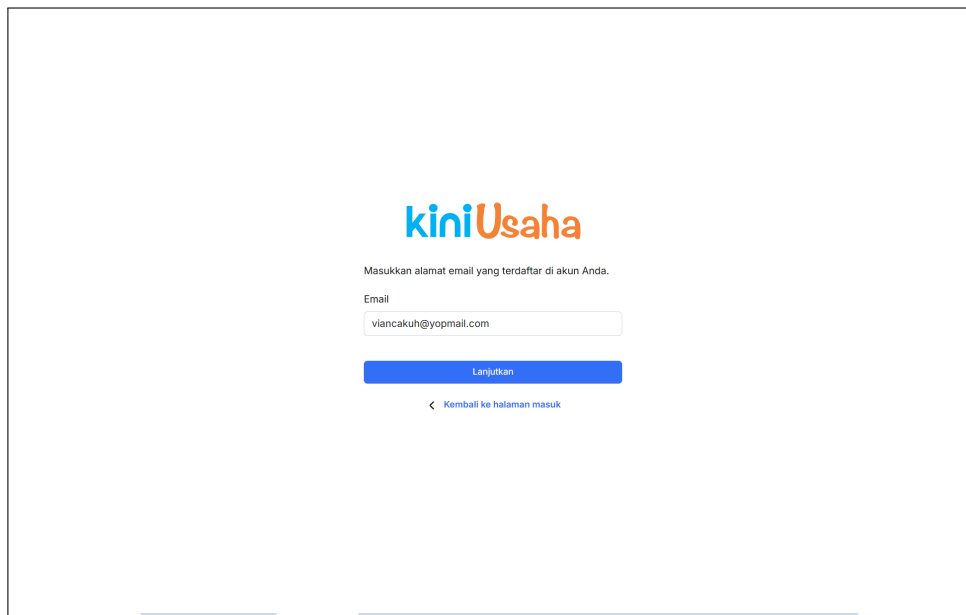
```
1 public function triggerOTP($type , $phone , $email)
2 {
3     $userOTPIdentifier = Session::get('companyDebiturID')
4     ;
5     $data = [
6         'userOTPIdentifier' => $userOTPIdentifier ,
7         'userOTPSendType' => $type == 1 ? 1 : 2,
8         'userOTPSendTo' => $type == 1 ? $phone : $email ,
9         'userOTPStepProcess' => 'CompanyRegister' ,
10        'source' => 'OPS' ,
11    ];
12
13    $apiHelper = new ApiHelper();
14    $requestOTP = $apiHelper->post($this->gateway . $this
15    ->api , $data);
16    return $requestOTP;
17 }
```

Kode 3.2: Potongan kode halaman Register ketika trigger OTP

3. Halaman Lupa Kata Sandi

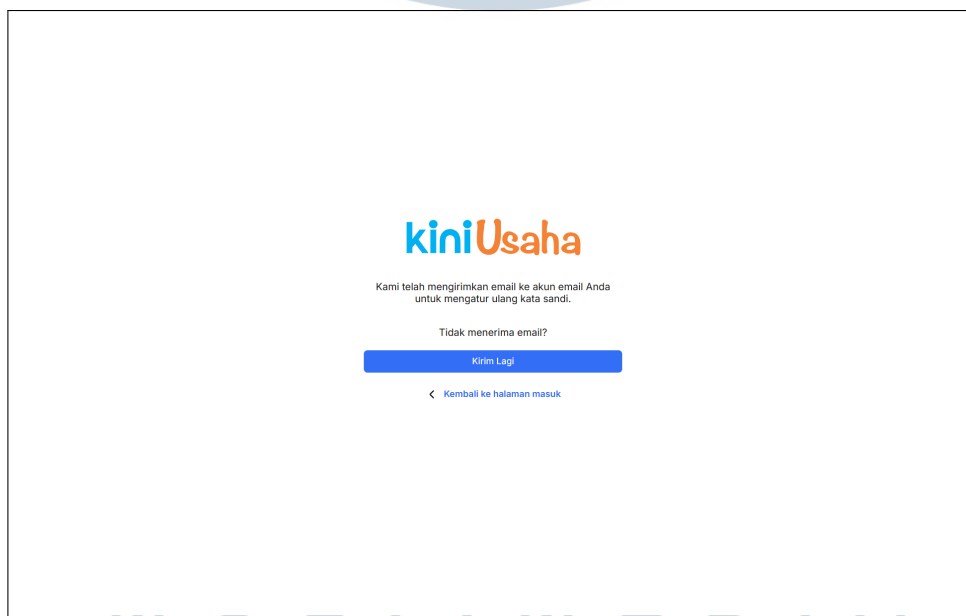
Halaman ini disediakan untuk membantu pengguna yang mengalami kendala dalam mengakses akun karena lupa kata sandi. Melalui proses pemulihan ini, pengguna dapat mengatur ulang sandi dengan menggunakan *link* verifikasi yang dikirimkan melalui *email*.

Tampilan awal halaman ini memungkinkan pengguna untuk memasukkan alamat *email* yang terdaftar, seperti ditunjukkan pada Gambar 3.32.



Gambar 3.32. Tampilan awal halaman Lupa Kata Sandi untuk memasukkan email

Setelah itu, pengguna dapat menekan tombol untuk memicu pengiriman *email* pemulihan, seperti yang terlihat pada Gambar 3.33.



Gambar 3.33. Halaman konfirmasi untuk mengirim ulang link pemulihan

Jika *link* berhasil dikirim dan diakses melalui *email*, pengguna akan diarahkan ke halaman pengisian ulang kata sandi, ditampilkan pada Gambar 3.34.

Gambar 3.34. Halaman utama untuk ubah kata sandi

Potongan kode untuk halaman Lupa Kata Sandi dapat dilihat pada Kode 3.3.

```

1 public function forgotPassword(Request $request)
2 {
3     try {
4         $forgotModel = new ForgotPasswordModel();
5
6         $forgotModel->company_debitor_email = $request->
input('forgot_email');
7         $forgotModel->company_debitor_ip_address =
$request->ip();
8
9         $validateModel = $forgotModel->validate();
10
11         if ($validateModel->fails()) {
12             foreach ($validateModel->messages()->
getMessages() as $field_name => $messages) {
13                 $error_array[] = $messages;
14             }
15
16             $this->responseError['message'] =
$error_array;
17             return json_encode($this->responseError);
18         }

```

```

19
20     $apiHelper = new ApiHelper();
21     $generateList = $apiHelper->post($this->gateway .
    $this->api, $forgotModel->toArray());
22     if ($generateList['statuscode'] == 200) {
23         $timeRemain = 120;
24         $minutesDef = floor(($timeRemain / 60) % 60);
25         $secondsDef = $timeRemain % 60;
26         $minutes = $minutesDef;
27         $seconds = sprintf('%02d', $secondsDef);
28         $time = $minutes . ':' . $seconds;
29         $generateList['data']['seconds'] =
    $timeRemain;
30         $generateList['data']['time'] = $time;
31     }
32
33     return json_encode($generateList);
34 } catch (\Exception $e) {
35     $this->responseError['message'] = $e->getMessage
    ();
36     return json_encode($this->responseError);
37 }
38 }
39

```

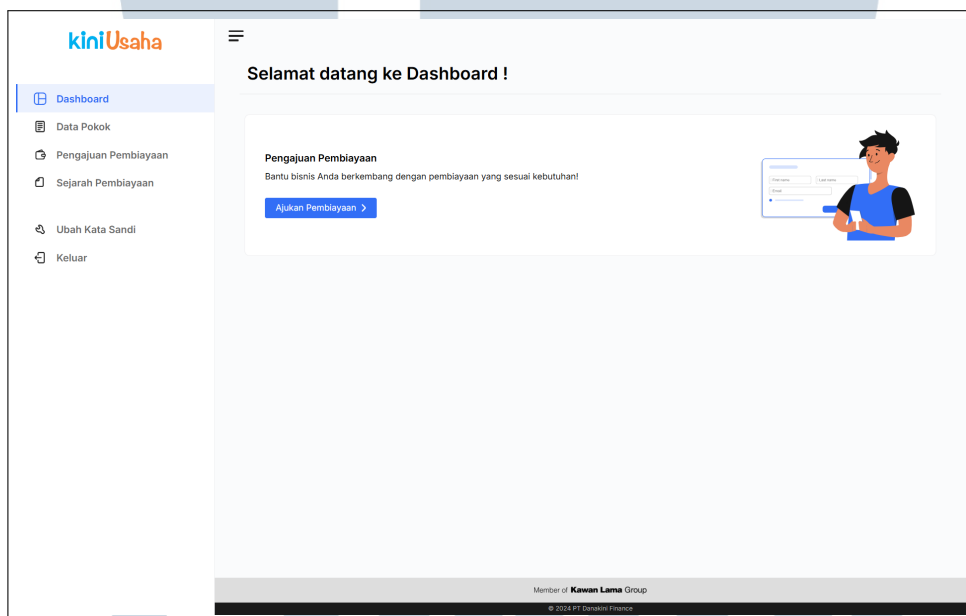
Kode 3.3: Potongan kode halaman Lupa Kata Sandi



4. Halaman Dashboard

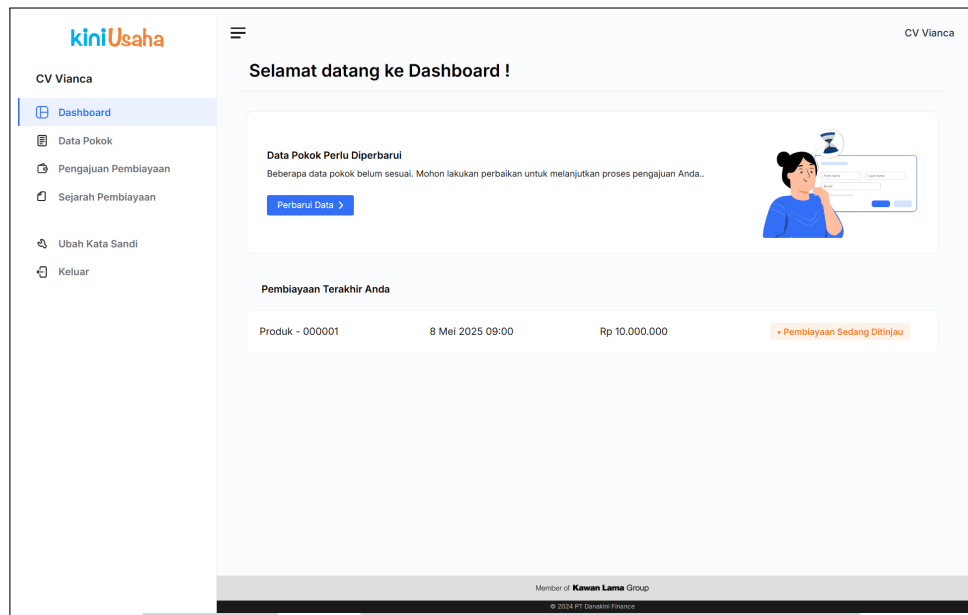
Setelah berhasil login, pengguna diarahkan ke halaman ini sebagai pusat navigasi utama. Halaman Dashboard dilengkapi dengan *sidebar* yang berfungsi sebagai menu navigasi untuk mengakses berbagai fitur lainnya, seperti halaman Data Pokok, Pengajuan Pembiayaan, Sejarah Pembiayaan, Ubah Kata Sandi, maupun Keluar (*logout*).

Di bagian tengah halaman, ditampilkan berbagai *widget* informatif yang menyesuaikan dengan kondisi terkini pengguna. Jika pengguna baru saja login dan belum memiliki pengajuan pembiayaan, maka akan ditampilkan tampilan awal Dashboard seperti yang terlihat pada Gambar 3.35.



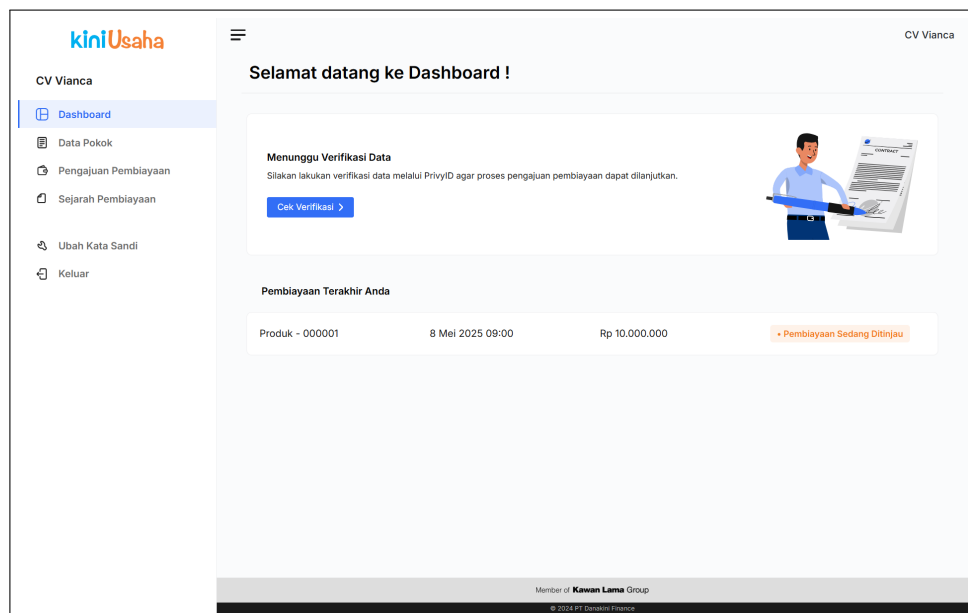
Gambar 3.35. Tampilan awal Dashboard setelah login

Apabila terdapat data pokok yang perlu direvisi, sistem akan menampilkan informasi dan ajakan untuk memperbaiki data tersebut, seperti yang ditunjukkan pada Gambar 3.36.



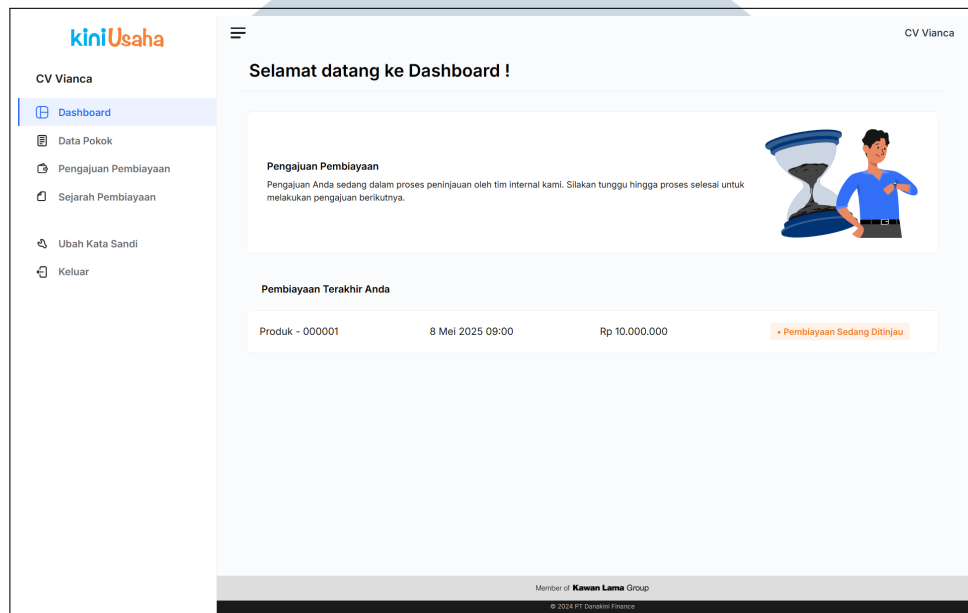
Gambar 3.36. Dashboard dengan informasi data pokok perlu direvisi

Jika pengguna sedang menunggu proses verifikasi data, maka sistem akan menampilkan informasi dan tombol untuk melanjutkan proses verifikasi, seperti pada Gambar 3.37.



Gambar 3.37. Dashboard dengan informasi menunggu verifikasi data

Apabila pengguna telah mengajukan pembiayaan dan sedang dalam proses peninjauan, maka sistem akan menampilkan informasi tersebut, seperti pada Gambar 3.38.



Gambar 3.38. Dashboard dengan informasi pembiayaan sedang ditinjau

Potongan kode untuk halaman Dashboard dapat dilihat pada Kode 3.4.

```

1 public function index()
2 {
3     try {
4         $userId = Session::get('userinfo')['userId'];
5
6         $apiHelper = new ApiHelper();
7         $generateList = $apiHelper->get($this->gateway .
8         $this->api, $userId);
9         if ($generateList['statuscode'] == 200) {
10             $data = [
11                 'userInfo' => Session::get('userinfo'),
12                 'dashboardData' => [],
13                 'isDashboard' => true,
14                 'breadcrumb' => 'dashboard',
15                 'regexPattern' => $this->regexPattern,
16                 'dashboardCard' => $generateList['data'],
17                 'lastFinancing' => $generateList['data']['
18                 'loan_application'] ?? '',

```

```

17         'accessMenuList' => Session::get('
18         userinfo')['menuList'] ?? [],
19         ];
20
21         return view('dashboard', $data);
22     } else {
23         return redirect(route('result.show', ['type'
24         => 'db']) . '#1');
25     }
26 } catch (\Exception $e) {
27     return redirect(route('result.show', ['type' => '
28     db']) . '#2');
29 }

```

Kode 3.4: Potongan kode halaman Dashboard

5. Halaman Sejarah Pembiayaan

Halaman ini menyajikan riwayat pengajuan pembiayaan yang telah dilakukan oleh pengguna, beserta status terkini dari masing-masing pengajuan, yang ditunjukkan pada Gambar 3.39. Pengguna juga dapat mengakses dokumen-dokumen terkait apabila proses telah selesai.

Nama Produk	Nomor Kontrak	Jumlah Pembiayaan (Rp)	Tanggal Pengajuan	Status	Tindakan
Produk	000007	70.000.000	08 Mei 2025 17:10:00	Lanjutkan proses pembiayaan	
Produk	000006	60.000.000	08 Mei 2025 16:23:14	Pencairan dana berhasil	Unduh Dokumen PKS
Produk	000005	50.000.000	08 Mei 2025 14:45:00	Menunggu tanda tangan debitur	Cek Status Dokumen
Produk	000004	40.000.000	08 Mei 2025 13:00:00	Pembiayaan ditolak	
Produk	000003	30.000.000	08 Mei 2025 11:30:00	Pembiayaan sedang ditinjau	
Produk	000002	20.000.000	08 Mei 2025 10:15:00	Data pokok membutuhkan revisi	
Produk	000001	10.000.000	08 Mei 2025 09:00:00	Pembiayaan sedang ditinjau	

Showing 1 to 7 of 7 entries

Gambar 3.39. Halaman Sejarah Pembiayaan

Potongan kode untuk halaman Sejarah Pembiayaan dapat dilihat pada Kode 3.5.

```

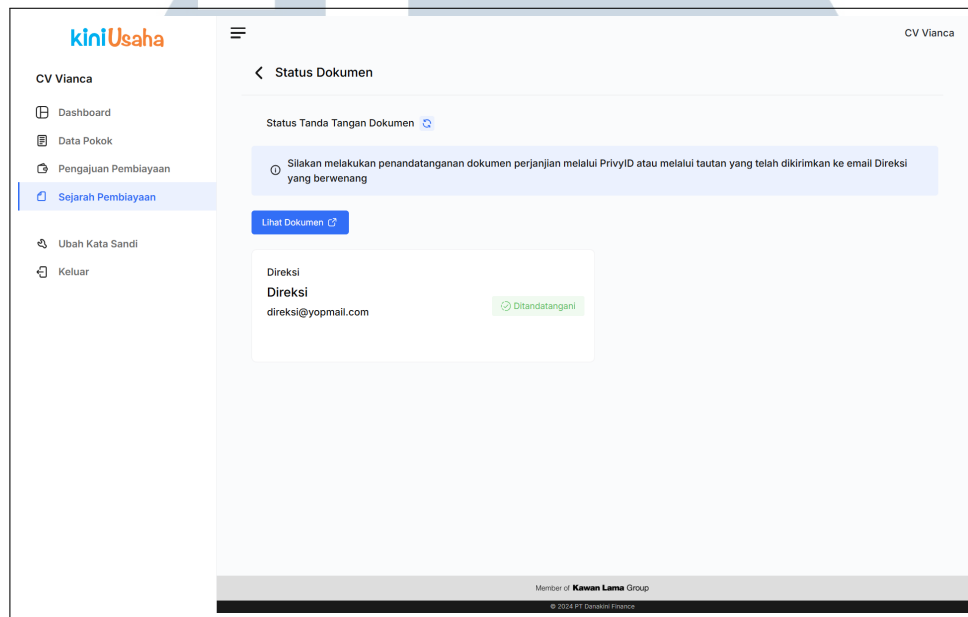
1      <div class="container-fluid p-0">
2          <div class="row">
3              <div class="d-flex">
4                  <div class="w-100">
5                      <div class="row">
6                          <div class="white-bg-corner mx-auto p
7                          -4">
8                              <div class="table-container col
9                              -12">
10                                 <table id="funding-table"
11                                 class="table table-medium
12                                 table-responsive row-border w-100">
13                                     <thead>
14                                         <tr>
15                                             <th>Nama Produk </
16                                             th>
17                                             <th>Nomor Kontrak
18                                             </th>
19                                             <th>Jumlah
20                                             <th>Tindakan </th>
21                                             <th>Tanggal
22                                             <th>Status </th>
23                                             <th>Tindakan </th>
24                                         </tr>
25                                         </thead>
26                                         <tbody>
27                                         </tbody>
28                                     </table>
29                                 </div>
30                                 </div>
31                             </div>
32                         </div>
33                     </div>
34                 </div>
35             </div>
36         </div>
37     </div>

```

Kode 3.5: Potongan kode halaman Sejarah Pembiayaan

6. Halaman Status Dokumen

Pada halaman ini, pengguna dapat memantau status penandatanganan dokumen pembiayaan dan mengunduh dokumen PKS. Halaman ini juga menyediakan tombol untuk mengirim ulang *link* penandatanganan jika diperlukan, seperti yang ditunjukkan pada Gambar 3.40.



Gambar 3.40. Halaman Status Dokumen

Potongan kode untuk halaman Status Dokumen dapat dilihat pada Kode 3.6.

```
1 public function documentStatus($id)
2 {
3     try {
4         $data = explode('|', $this->decryptAcc($id));
5         $losId = $data[0];
6
7         $apiHelper = new ApiHelper();
8         $generateList = $apiHelper->get($this->gateway .
9         $this->api, $losId);
10        $data = [
11            'userInfo' => Session::get('userinfo'),
12            'data' => $generateList['data'] ?? [],
13            'doc_url' => $generateList['data']['doc_url'],
14        ],
15        'code' => $losId ,
```

```

14         'regexPattern' => $this->regexPattern ,
15         'breadcrumb' => 'Status Dokumen',
16         'breadcrumburl' => route('funding.index'),
17     ];
18
19     return view('funding.doc_status', $data);
20 } catch (\Exception $ex) {
21     $this->responseError['message'] = $ex->getMessage
22     ();
23     return redirect(route('result.show', ['type' => '
24     fund']) . '#dcs');
25 }

```

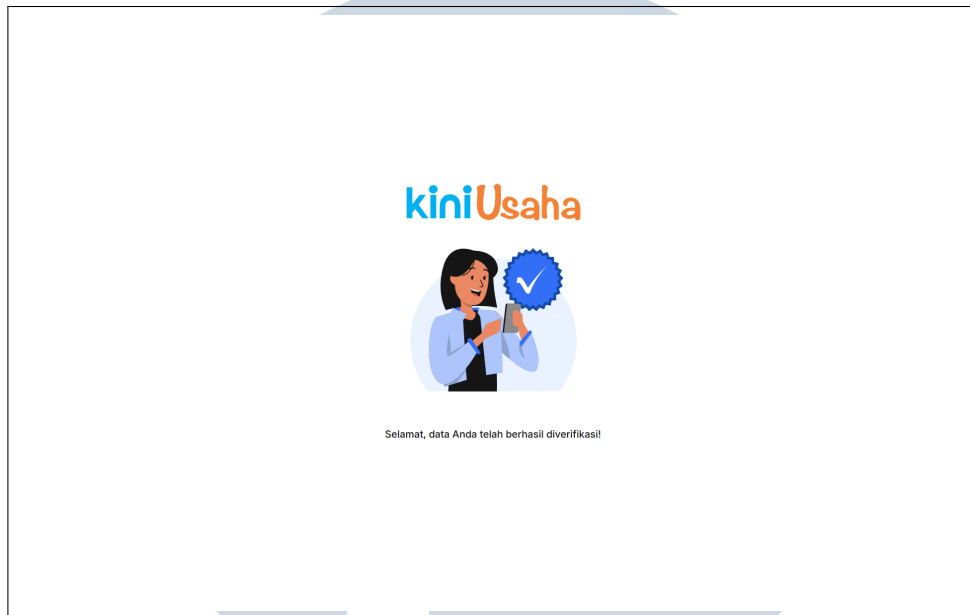
Kode 3.6: Potongan kode halaman Status Dokumen

7. Halaman Verifikasi Data via Layanan E-Sign

Halaman ini diakses oleh pihak direksi melalui *link* yang dikirim via *email*, dan digunakan untuk mengisi informasi yang diperlukan dalam proses penandatanganan dokumen secara digital melalui layanan *e-sign*, seperti yang ditunjukkan pada Gambar 3.41.

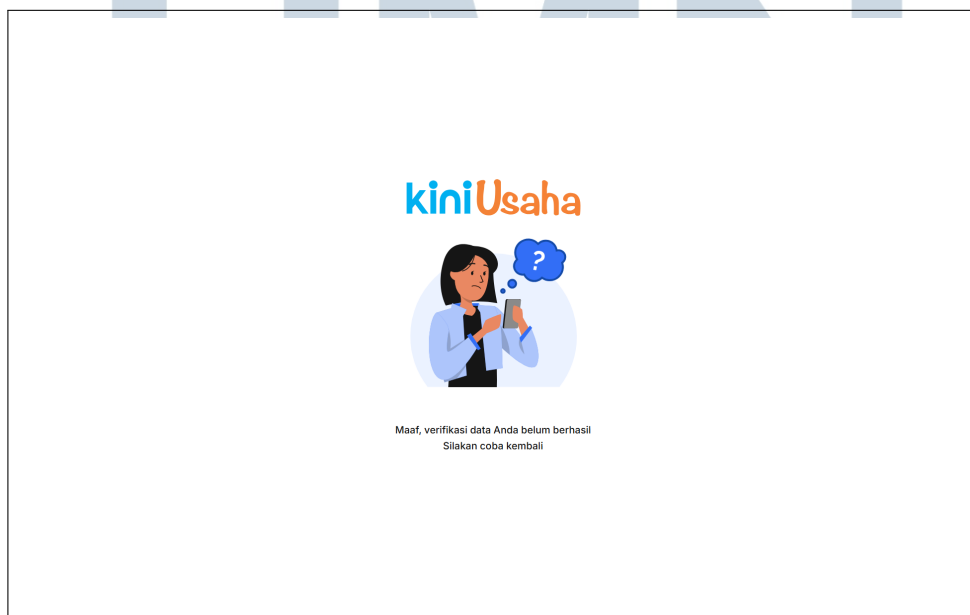
Gambar 3.41. Halaman Verifikasi Data via layanan e-sign

Apabila verifikasi data berhasil, pengguna diarahkan ke halaman konfirmasi, dapat dilihat pada Gambar 3.42.



Gambar 3.42. Halaman Verifikasi Data berhasil

Sebaliknya, jika verifikasi data gagal, maka pengguna akan diarahkan ke halaman yang berisi informasi bahwa verifikasi data gagal, seperti yang ditunjukkan pada Gambar 3.43.



Gambar 3.43. Halaman Verifikasi Data gagal

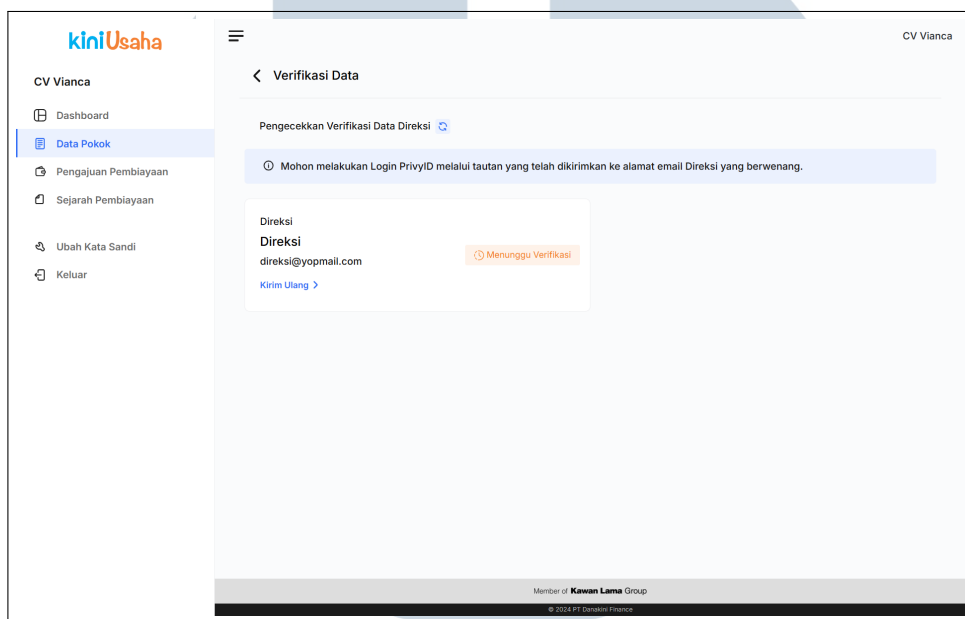
Potongan kode untuk halaman Verifikasi Data via layanan *e-sign* dapat dilihat pada Kode 3.7.

```
1 public function index(Request $request)
2 {
3     try {
4         $code = $request->query('code');
5         $id = json_decode($this->decryptAcc($code));
6
7         $dataCust = [
8             'user_id' => $id->user_id,
9             'los_id' => $id->los_id,
10        ];
11
12        $apiHelper = new ApiHelper();
13        $generateList = $apiHelper->get($this->gateway .
14        $this->api, $dataCust);
15        if ($generateList['statuscode'] == 200) {
16            $data = [
17                'regexPattern' => $this->regexPattern,
18                'data' => $generateList['data'],
19                'urlekyc' => $this->gateway,
20            ];
21
22            Session::put('customerData', $data);
23
24            if ($generateList['data']['submit'] == true)
25            {
26                return redirect($this->gateway);
27            }
28
29            return view('customer_verification', $data);
30        } else {
31            return redirect(route('result.show', ['type'
32            => 'ekyc']) . '#1');
33        }
34    } catch (\Exception $e) {
35        return redirect(route('result.show', ['type' => '
36        ekyc']) . '#2');
```

Kode 3.7: Potongan kode halaman Verifikasi Data via layanan e-sign

8. Halaman Monitoring Verifikasi Data

Halaman ini ditujukan bagi pengguna untuk memantau status verifikasi data direksi, yang dapat dilihat pada Gambar 3.44. Status yang ditampilkan meliputi "terverifikasi", "menunggu verifikasi", atau "gagal verifikasi".



Gambar 3.44. Halaman Monitoring Verifikasi Data

Potongan kode untuk halaman Monitoring Verifikasi Data dapat dilihat pada Kode 3.8.

```
1 public function borrowerVerification()
2 {
3     $userId = Session::get('userinfo')['userId'];
4
5     $apiHelper = new ApiHelper();
6     $generateList = $api->get($this->gateway . $this->api
7     , $userId);
8     if ($generateList['statuscode'] == 200) {
9         $data = array(
10             "userinfo" => Session::get('userinfo'),
11             "verificationData" => $generateList['data'],
12             "breadcrumb" => "Verifikasi Data",
13             "breadcrumburl" => "/master-data",
14             "regexPattern" => $this->regexPattern
15         );
16     }
```

```

16         return view('master_data.borrower_verification',
17             $data);
18     }
19 }

```

Kode 3.8: Potongan kode halaman Monitoring Verifikasi Data

9. Halaman Ubah Kata Sandi

Halaman ini memungkinkan pengguna untuk mengganti kata sandi akun mereka. Sistem akan melakukan pengecekan terhadap riwayat kata sandi melalui validasi yang ada, dan memastikan bahwa kata sandi baru belum pernah digunakan sebelumnya, ditunjukkan pada Gambar 3.45.

Gambar 3.45. Halaman Ubah Kata Sandi

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

Potongan kode untuk halaman Ubah Kata Sandi dapat dilihat pada Kode 3.9.

```
1      $(document).ready(function() {
2          $("#form-input").submit(function(e) {
3              e.preventDefault();
4
5              if ($(this).valid()) {
6                  const data = kebabToSnakeKeys($("#form-input")
7                      ).serialize();
8
9                  $.ajax({
10                      url: "{{ route('password.update') }}",
11                      type: 'POST',
12                      data: data,
13                      dataType: 'json',
14                      cache: false,
15                      beforeSend: function() {
16                          showLoadBar();
17                      },
18                      success: function(response) {
19                          hideLoadBar();
20                          if (response.statuscode == 200) {
21                              sweetURLOK("Berhasil", response.
22                                  message, "success", "/",
23                                  "Kembali ke Dashboard");
24                          } else {
25                              sweetURLOK("Oops", response.
26                                  message, "error", "", "Kembali");
27                          }
28                      },
29                      error: function(error) {
30                          hideLoadBar();
31                          console.log(error);
32                      }
33                  });
34      });
35  });
```

Kode 3.9: Potongan kode halaman Ubah Kata Sandi

3.4 Kendala dan Solusi yang Ditemukan

3.4.1 Kendala

Selama pelaksanaan kegiatan magang di PT Dana Kini Indonesia, terdapat beberapa kendala yang dihadapi dalam proses pengembangan *website*, antara lain:

1. Kesulitan dalam memahami alur sistem dan struktur kode yang sudah ada karena tidak tersedianya dokumentasi teknis yang memadai, sehingga proses adaptasi menjadi lebih lama.
2. Struktur kode yang ditulis menggunakan pendekatan Vanilla JavaScript dan manipulasi *Document Object Model* (DOM) secara langsung menyebabkan kode menjadi kurang modular dan sulit dipelihara.
3. Proses integrasi API tidak berjalan optimal karena implementasi teknis kurang selaras dengan kebutuhan sistem akibat kurangnya pemahaman terhadap *user requirements* yang diberikan.

3.4.2 Solusi

Untuk mengatasi kendala-kendala tersebut, ditemukan beberapa solusi, antara lain:

1. Melakukan pembacaan kode secara bertahap disertai dengan diskusi langsung bersama atasan atau anggota tim guna memperoleh pemahaman menyeluruh terhadap cara kerja sistem.
2. Menggunakan library jQuery guna menyederhanakan manipulasi elemen *Document Object Model* (DOM) dan meningkatkan modularitas serta kemudahan pemeliharaan kode.
3. Dilakukan komunikasi yang lebih terstruktur dengan tim *Back End*, termasuk penyepakatan format data dan koordinasi rutin, sehingga pengembangan antarmuka dapat disesuaikan dengan kebutuhan sistem dan proses integrasi berjalan lebih lancar.