

BAB 3 PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Pelaksanaan kerja magang yang dilakukan yaitu sebagai *Fullstack Developer* yang berada di bawah posisi *engineering*. Pelaksanaan magang tersebut berada di bawah pengawasan Ibu Nadila sebagai CEO sekaligus menjadi supervisor yang mengarahkan dan membimbing untuk memastikan proyek dan tugas yang diberikan dapat dijalankan sesuai dengan ketentuan dan standar yang diberikan oleh perusahaan.

Selama proses pengembangan website, terdapat koordinasi yang dilakukan secara langsung melalui pertemuan lewat google meet maupun melalui pesan Telegram dengan supervisor. Tujuan dari koordinasi ini adalah untuk memberikan update mengenai progres pengerjaan website serta menerima masukan terkait perubahan yang dilakukan.

3.2 Tugas yang Dilakukan

Selama pelaksanaan magang, tugas utama yang dilakukan adalah melakukan pengembangan dan penyempurnaan antarmuka pengguna pada *website* *simplr*. *Simplr* merupakan platform *digital* yang dikembangkan oleh PT. Ahli Karya Improvi Teknologi, dengan tujuan utama untuk membantu perusahaan dalam mencari, merekrut, dan mengelola kandidat melalui teknologi kecerdasan buatan. Selama magang, penugasan difokuskan pada pengembangan berbagai fitur *frontend*. Seluruh pengembangan dilakukan untuk memastikan pengalaman pengguna yang memuaskan.

3.3 Uraian Pelaksanaan Magang

Pelaksanaan kerja magang diuraikan seperti pada Tabel 3.1.

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

Minggu Ke -	Pekerjaan yang dilakukan
1	Pengenalan lingkungan magang dan mendapatkan briefing terkait projek yang dilakukan
2-3	Mempelajari penggunaan <i>React</i> secara mandiri
4	Mempelajari <i>code project</i> yang telah dibuat sebelumnya
5-7	Mengerjakan slicing create assessment page pada company
7-13	Melakukan <i>improvements</i> dan <i>bug fixing</i> di berbagai halaman website

Pengembangan website ini menggunakan pendekatan SDLC Agile, Agile memungkinkan untuk bekerja secara iteratif dan kolaboratif melalui tahapan-tahapan seperti perencanaan, perancangan, development, pengujian, dan evaluasi dalam setiap sprint. Setiap fitur yang dikembangkan dievaluasi secara berkala dan disesuaikan berdasarkan masukan dari supervisor [3].

3.3.1 Perangkat yang digunakan

Berikut merupakan spesifikasi *software* dan *hardware* dalam proses pelaksanaan magang di PT Ahli Karya Improovi Teknologi

A Perangkat Lunak

Perangkat lunak yang digunakan dalam pengerjaan proyek pada PT Ahli Karya Improovi teknologi sebagai berikut:

1. *Visual Studio Code*
2. *Mockoon*
3. *Jira*
4. *Github*

B Perangkat Keras

Perangkat keras yang digunakan dalam pengerjaan proyek pada PT. Ahli Karya Improovi Teknologi sebagai berikut:

1. *Processor*: AMD Ryzen 7 3700x
2. Sistem Operasi: Windows 11 64-bit
3. RAM: 8 GB
4. SSD: 512 GB

3.3.2 Framework dan library yang digunakan

Pengembangan website *recruitment* selama kerja magang dilakukan dengan menggunakan *framework* dan *library*. Setiap *framework* dan *library* yang digunakan memiliki peran dan kegunaannya masing-masing. *Framework* dan *library* yang digunakan sebagai berikut:

1. Tailwind CSS

Tailwind CSS adalah sebuah framework CSS yang bersifat utility-first, yang artinya menyediakan class-class siap pakai untuk membangun antarmuka pengguna secara cepat dan efisien. Tidak seperti framework CSS konvensional yang menyediakan komponen siap pakai, Tailwind memungkinkan pengembang untuk langsung mengatur style pada elemen HTML melalui class utility yang spesifik. Pendekatan ini memberikan fleksibilitas tinggi dalam membangun desain yang konsisten tanpa perlu menulis CSS custom secara manual [4].

2. React

React adalah sebuah library JavaScript yang digunakan untuk membangun antarmuka pengguna (User Interface) secara efisien dan interaktif. React memungkinkan pengembang membuat komponen-komponen UI yang dapat digunakan kembali, sehingga memudahkan dalam mengelola dan mengembangkan aplikasi web yang kompleks. Salah satu keunggulan utama React adalah penggunaan Virtual DOM, sebuah mekanisme yang mempercepat proses rendering dengan hanya memperbarui bagian-bagian yang berubah di halaman web, sehingga meningkatkan performa aplikasi [5].

3.3.3 User Requirements

User requirements merupakan analisis yang dilakukan untuk mengidentifikasi kebutuhan pengguna. Melalui analisis ini sebagai berikut:

1. Membuat page *create assessment*, pengguna membutuhkan halaman khusus untuk membuat *assessment* baru. Di halaman ini, pengguna dapat mengatur judul *assessment*, deskripsi singkat, serta jenis dan durasi tes yang akan diberikan kepada kandidat
2. Pada fitur *payment integration* untuk *subscription website*, pengguna dari pihak perusahaan membutuhkan akses untuk melihat berbagai pilihan paket langganan yang tersedia dan melakukan pembelian paket yang sesuai dengan kebutuhannya. Sistem harus terintegrasi dengan layanan *payment gateway* agar pembayaran dapat dilakukan dengan aman dan cepat.

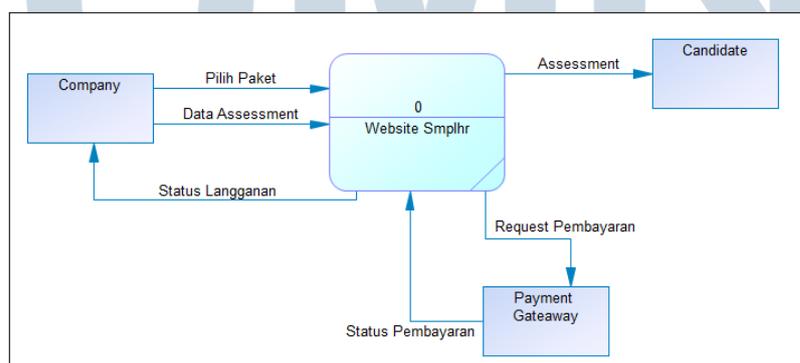
3.3.4 Perancangan Update Website Smplhr

Perancangan sistem digunakan untuk menggambarkan alur proses dari fitur-fitur yang dikembangkan selama magang. Perancangan sistem meliputi *flowchart* dan *flow data*. *Flowchart* digunakan untuk memvisualisasikan proses dari sistem secara detail, sedangkan *flow data* menunjukkan bagaimana data mengalir antar entitas eksternal, proses, dan penyimpanan data dalam sistem [6].

A Data Flow Diagram

Gambar 3.1 memvisualisasikan aliran data utama antara entitas eksternal dengan sistem, sesuai ruang lingkup pekerjaan selama magang.

1. Data Flow Diagram Level 0

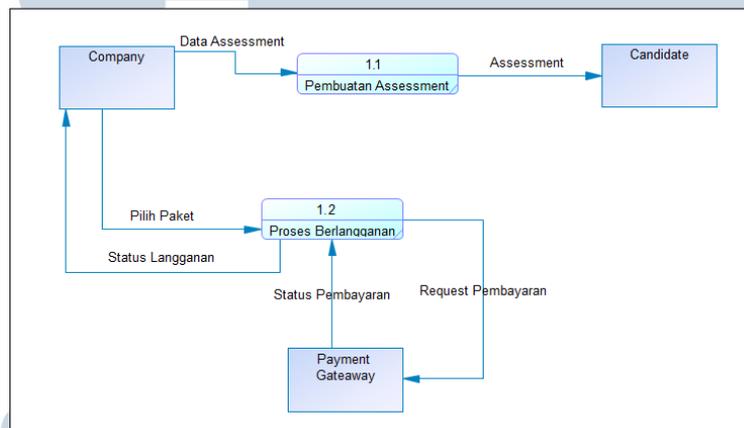


Gambar 3.1. Data Flow Diagram Level 0

Diagram flow data seperti pada Gambar 3.1, menunjukkan interaksi antara entitas eksternal dan sistem utama, yaitu Website Smplhr. Entitas eksternal

yang terlibat meliputi Company, Candidate, dan Payment Gateway. Company mengirimkan data *assessment* serta pilihan paket langganan ke sistem. Sistem kemudian memproses data tersebut dan mengirimkan *assessment* ke kandidat yang dituju. Di sisi lain, ketika Company memilih paket langganan, sistem akan mengirimkan permintaan pembayaran ke Payment Gateway, dan kemudian menerima status pembayaran dari pihak tersebut. Setelah menerima status pembayaran, sistem memperbarui status langganan dan menampilkannya kembali ke Company. DFD Level 0 ini memberikan gambaran umum mengenai batasan sistem dan bagaimana interaksi data terjadi antara sistem frontend dan entitas di luar sistem.

2. Data Flow Diagram Level 1



Gambar 3.2. Data Flow Diagram Level 1

Gambar 3.2 merupakan penguraian dari proses utama pada DFD Level 0, yaitu Website Smplhr. Diagram ini memecah proses menjadi dua bagian utama berdasarkan fitur frontend yang dikembangkan, yaitu "Pembuatan Assessment" dan "Proses Berlangganan". Company mengirimkan data *assessment* yang diproses dalam "Pembuatan Assessment" dan dikirim ke Candidate. Sementara itu, proses langganan dimulai ketika Company memilih paket, sistem mengirimkan permintaan pembayaran ke Payment Gateway, menerima status pembayaran, dan mengirimkan pembaruan status kembali ke Company. Pada DFD Level 1 ini tidak disertakan elemen *data store*, karena pengembangan difokuskan pada sisi frontend. Proses yang digambarkan hanya mencakup interaksi antar entitas eksternal dan antarmuka sistem, tanpa menyentuh aspek penyimpanan data atau manajemen database yang merupakan tanggung jawab sisi backend.

B List Improvements

Tabel 3.2 berikutnya merupakan daftar pengembangan dan perbaikan fitur yang telah dilakukan selama proses magang. Setiap pengembangan ditujukan untuk meningkatkan pengalaman pengguna.

Tabel 3.2. Tabel List *Improvements*

Nomor -	List Improvements
1	Membuat modal yang muncul saat pengguna menekan tombol “Share”, berisi tautan company page untuk dibagikan ke kandidat.
2	Menambahkan efek kursor tangan pada elemen interaktif untuk meningkatkan pemahaman pengguna bahwa elemen bisa diklik.
3	Menambahkan logika auto-fill nama kandidat pada kolom 'Share to' berdasarkan data kandidat yang sedang dibuka.
4	Mengubah format tampilan correct answer dari JSON indeks ke bentuk teks jawaban agar lebih mudah dipahami.
5	Mengembangkan fitur untuk membuat dan mengirim assessment secara langsung ke kandidat yang dipilih dari halaman Candidate.
6	Mengembangkan halaman pemilihan paket langganan dan mengintegrasikannya dengan gateway pembayaran Xendit

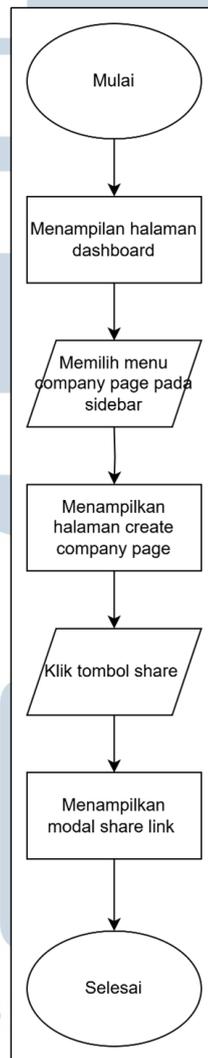
C Flowchart

Pembuatan flowchart berdasarkan list improvements yang akan dilaksanakan menggambarkan alur proses dari fitur yang akan dikembangkan. Diagram ini menunjukkan tahapan interaksi pengguna dengan sistem, dimulai dari tindakan awal hingga proses akhir yang terjadi pada sistem.

C.1 Penambahan Modal Link pada Tombol Share

Penambahan fitur modal yang muncul secara langsung setelah pengguna mengklik tombol “Share” bertujuan untuk memberikan akses cepat terhadap link yang dihasilkan dari proses penyimpanan data. Modal ini menampilkan

informasi berupa link yang dapat langsung disalin atau dibagikan kepada pihak terkait, sehingga memudahkan pengguna dalam melanjutkan proses distribusi atau komunikasi tanpa perlu mencari link tersebut secara manual. Dengan adanya fitur ini, alur kerja menjadi lebih efisien, meminimalkan kebingungan, dan memberikan informasi bahwa siap dibagikan. Hal ini juga mendukung pengalaman pengguna yang responsif dan terarah dalam penggunaan sistem.

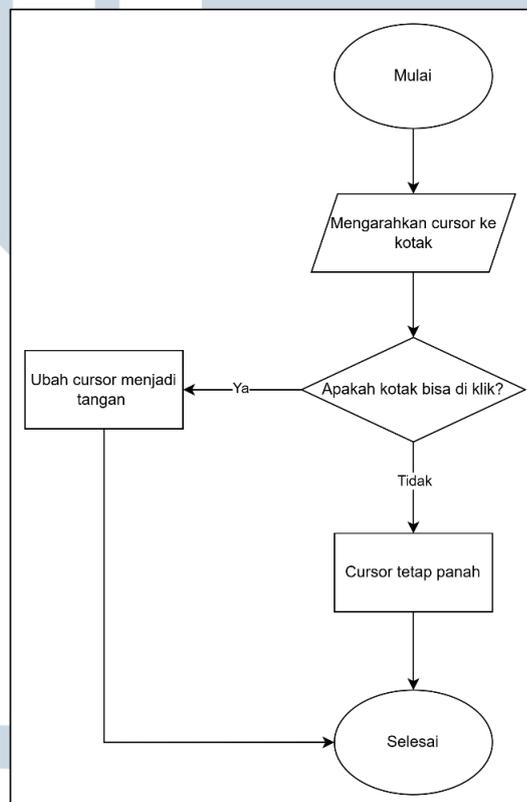


Gambar 3.3. Flowchart Penambahan Modal

Saat ini, halaman *career* page telah dibuat, namun terdapat kebutuhan tambahan dari pengguna untuk menyediakan fitur share. Seperti pada Gambar 3.3, website akan menampilkan halaman *dashboard* terlebih dahulu, kemudian pengguna dapat mengakses menu *company page* melalui *side bar*. Untuk menyebarkan *link*, pengguna cukup menekan tombol “Share” dan membagikan *link* tersebut.

C.2 Penambahan Cursor Pointer

Penggunaan kursor berbentuk tangan (pointer) saat pengguna mengarahkan kursor ke elemen berbentuk kotak yang dapat diklik bertujuan untuk memberikan isyarat visual bahwa elemen tersebut interaktif. Hal ini dilakukan untuk meningkatkan kejelasan antarmuka dan mengoptimalkan pengalaman pengguna (user experience), sehingga pengguna dapat dengan mudah memahami bahwa elemen tersebut memiliki aksi yang dapat dilakukan. Tanpa adanya indikator ini, pengguna bisa saja melewati elemen penting karena tidak menyadari bahwa kotak tersebut dapat diklik.

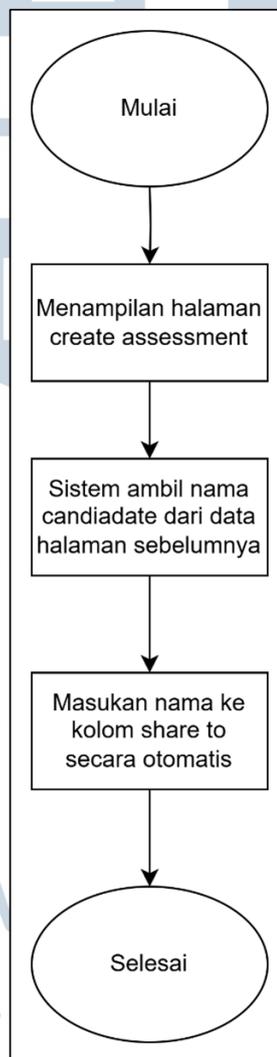


Gambar 3.4. Penambahan Cursor Pointer

Seperti pada Gambar 3.4, program akan menampilkan halaman dashboard terlebih dahulu. Pengguna kemudian dapat mengarahkan kursor ke kotak yang terdapat pada halaman dashboard. Jika elemen tersebut dapat ditekan, maka kursor akan berubah menjadi tangan yang menunjukkan bahwa elemen tersebut interaktif.

C.3 Penambahan Fitur Auto-fill

Fitur auto-fill pada kolom “Share to” yang secara otomatis mengisi nama kandidat saat membuat assessment dari menu Candidate bertujuan untuk mempermudah dan mempercepat proses input data oleh pengguna. Dengan fitur ini, pengguna tidak perlu lagi memilih atau mengetik ulang nama kandidat secara manual, karena sistem telah mengidentifikasi konteks kandidat yang sedang dibuka. Hal ini dapat mengurangi risiko kesalahan input, meningkatkan efisiensi kerja, dan menciptakan pengalaman pengguna yang lebih praktis.



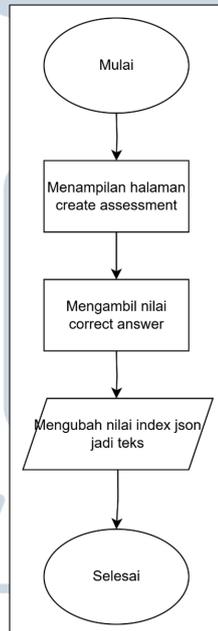
Gambar 3.5. Flowchart Nama Otomatis

Seperti pada Gambar 3.5, flowchart di atas menggambarkan alur sistem saat pengguna membuat assessment dari menu Candidate. Ketika pengguna membuka halaman detail kandidat dan memilih opsi “Create Assessment”, sistem secara

otomatis membaca data kandidat yang sedang dibuka dan langsung mengisi kolom “Share to” dengan nama kandidat tersebut. Proses *autofill* ini bertujuan untuk mempermudah pengguna dalam mengisi data, mengurangi kemungkinan kesalahan input, serta mempercepat alur kerja dalam pembuatan *assessment*. Dengan demikian, pengguna tidak perlu memilih nama kandidat secara manual, karena sistem sudah menampilkan informasi yang sesuai secara otomatis.

C.4 Perubahan Tampilan Correct Answer Menjadi Format Teks

Perubahan format data pada bagian “Correct Answer” dari yang sebelumnya berupa JSON angka menjadi teks jawaban dilakukan dengan tujuan untuk mempermudah pengguna dalam memahami dan memilih jawaban yang benar secara langsung. Dengan menampilkan teks jawaban secara langsung, pengguna tidak perlu lagi mengingat atau mencocokkan angka dengan pilihan jawaban yang tersedia, sehingga proses input menjadi lebih efisien. Fitur ini meningkatkan kejelasan data yang ditampilkan dan meminimalisir potensi kesalahan dalam memilih jawaban yang benar saat membuat atau mengedit *assessment*.

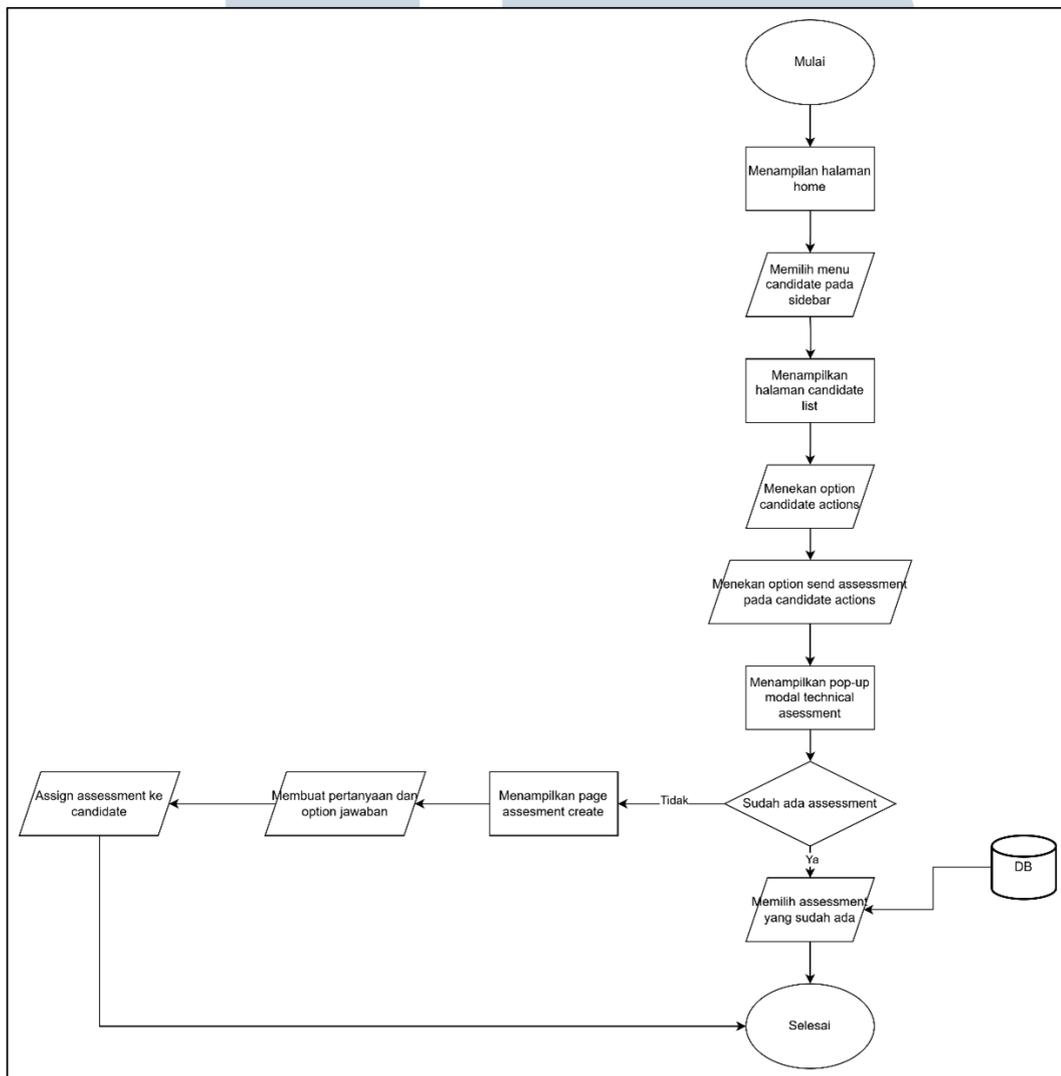


Gambar 3.6. Flowchart Konversi Nilai

C.5 Pembuatan Halaman Create Assessment

Penambahan fitur *create assessment* bertujuan untuk memberi *assessment* pada candidate yang mendaftar pada sebuah job. Dalam proses rekrutmen,

pengiriman *assessment* kepada candidate yang telah mendaftar pada sebuah *job* merupakan tahapan penting yang bertujuan untuk menyaring pelamar secara lebih objektif dan terukur. Setelah kandidat melewati proses seleksi awal seperti verifikasi dokumen dan pengecekan kesesuaian dengan kualifikasi dasar posisi, *assessment* dapat dikirim sebagai bagian dari tahapan seleksi lanjutan. HR dapat memilih satu atau beberapa kandidat, lalu mengirimkan *assessment* yang telah disiapkan sebelumnya secara otomatis melalui sistem.



Gambar 3.7. Flowchart send assessment

Pada Gambar 3.7 merupakan diagram flowchart untuk fitur send assessment. Proses dimulai dari pengguna (HR) yang mengakses halaman utama (home), kemudian memilih menu “candidate” pada sidebar untuk melihat daftar kandidat yang telah melamar. Setelah halaman daftar kandidat ditampilkan, pengguna

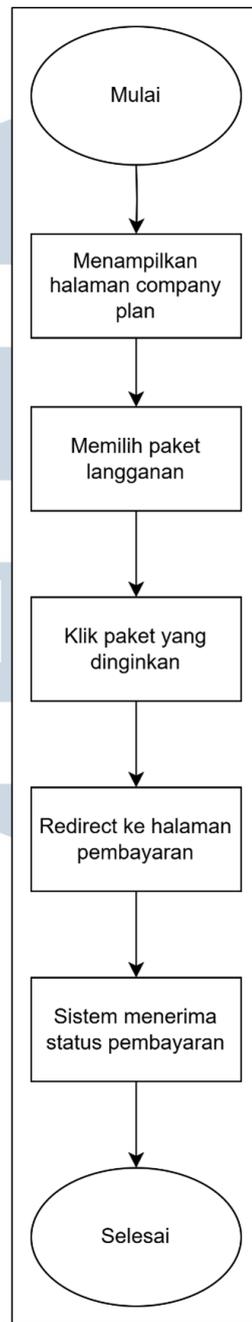
menekan opsi “candidate actions”, lalu memilih opsi “send assessment” pada menu tersebut. Langkah ini akan memunculkan pop-up modal untuk mengirim assessment kepada kandidat.

Selanjutnya, sistem akan memverifikasi apakah assessment untuk posisi tersebut sudah tersedia di database atau belum. Jika assessment sudah ada, pengguna cukup memilih salah satu assessment dari daftar yang tersedia dan langsung melakukan assign assessment ke kandidat. Namun, jika belum ada assessment yang tersedia, sistem akan mengarahkan pengguna ke halaman pembuatan assessment. Di sana, pengguna harus membuat pertanyaan beserta pilihan jawabannya terlebih dahulu. Setelah pertanyaan dibuat, assessment dapat di-assign ke kandidat untuk dikerjakan. Proses ini kemudian diakhiri setelah assessment berhasil dikirimkan, baik menggunakan template yang sudah ada maupun hasil pembuatan baru. Flowchart ini memberikan gambaran terstruktur mengenai bagaimana proses teknis pengelolaan dan pengiriman assessment berlangsung secara efisien dan terdokumentasi dalam system.

C.6 Penambahan Halaman Payment Integration

Penambahan halaman payment integration merupakan bagian penting dalam pengembangan sistem untuk mendukung kebutuhan pengguna dalam melakukan proses langganan layanan. Tujuan utama dari halaman ini adalah untuk memberikan kemudahan dan fleksibilitas bagi pengguna dalam memilih jenis langganan yang sesuai dengan kebutuhan mereka. Di halaman ini, pengguna dapat melihat berbagai pilihan paket langganan yang tersedia, di mana setiap paket memiliki keunggulan, fitur, dan batasan yang berbeda-beda.

Setelah pengguna menentukan paket yang diinginkan, sistem akan mengarahkan mereka secara otomatis ke halaman pembayaran yang telah terintegrasi dengan pihak ketiga, yaitu Xendit. Xendit merupakan penyedia layanan pembayaran digital yang memungkinkan proses transaksi dilakukan dengan aman, cepat, dan terpercaya. Dengan adanya halaman ini, proses berlangganan menjadi lebih mudah bagi pengguna untuk segera mengakses fitur-fitur yang sesuai dengan paket yang telah dipilih.



Gambar 3.8. Flowchart Payment Integration

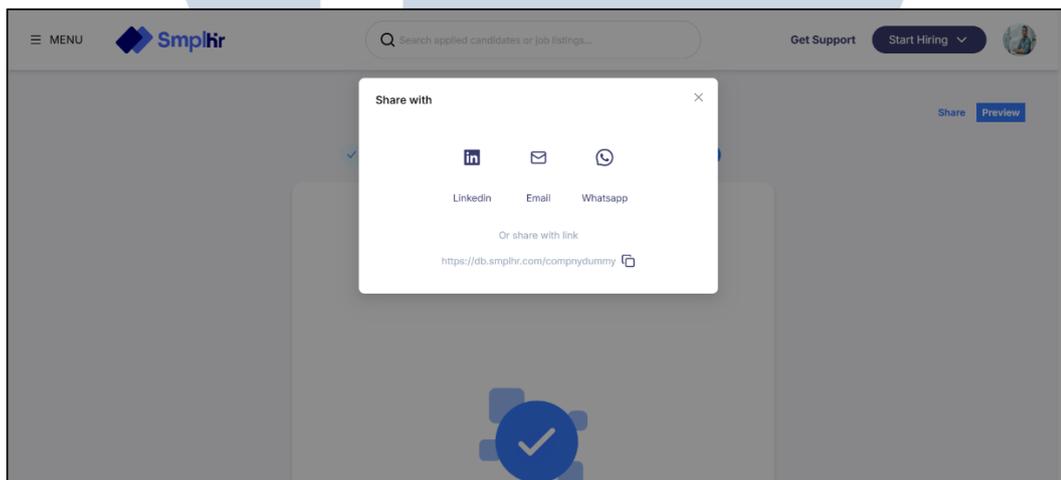
Flowchart di atas menggambarkan alur proses pembayaran yang terjadi saat pengguna melakukan langganan pada sistem. Proses dimulai ketika pengguna membuka halaman *Plan* dan memilih salah satu jenis subscription yang tersedia. Setelah pilihan dilakukan, sistem menampilkan detail paket yang dipilih beserta harga, kemudian pengguna menekan tombol “Subscribe”. Selanjutnya, sistem secara otomatis melakukan redirect ke halaman pembayaran milik Xendit. Di sana, pengguna dapat menyelesaikan transaksi menggunakan berbagai metode

pembayaran yang tersedia, seperti kartu kredit, e-wallet, atau transfer bank. Setelah proses pembayaran selesai, Xendit akan mengirimkan status pembayaran kembali ke sistem melalui webhook. Jika pembayaran berhasil, sistem akan mengaktifkan langganan pengguna sesuai paket yang dipilih. Namun, jika pembayaran gagal, sistem akan menampilkan pesan kegagalan dan memberikan opsi untuk melakukan pembayaran ulang.

D Implementasi fitur

D.1 Fitur Modal Link

Gambar 3.9 berikut menunjukkan modal yang muncul setelah tombol “Share” ditekan.



Gambar 3.9. Fitur Modal Link Setelah Tombol Share Ditekan

Seperti pada Gambar 3.9, setelah pengguna mengakses halaman company page dan menekan tombol share, maka akan muncul modal yang berfungsi untuk menampilkan link company page untuk dibagikan.

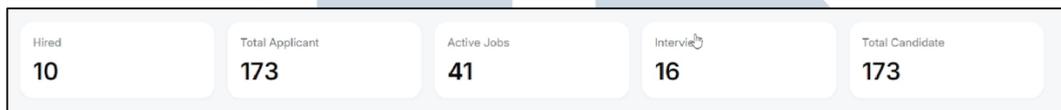
D.2 Fitur Cursor Pointer

Gambar 3.10 berikut menunjukkan kursor default berupa panah.

Hired 10	Total Applicant 173	Active Jobs 41	Interview 16	Total Candidate 173
-------------	------------------------	-------------------	-----------------	------------------------

Gambar 3.10. Tampilan sebelum Penambahan Cursor Pointer

Sebelum dilakukan pengembangan, elemen yang dapat diklik seperti kotak, masih menggunakan kursor default tanda panah. Hal ini dapat membuat pengguna tidak langsung menyadari bahwa elemen tersebut dapat diklik, dapat dilihat pada Gambar 3.10.



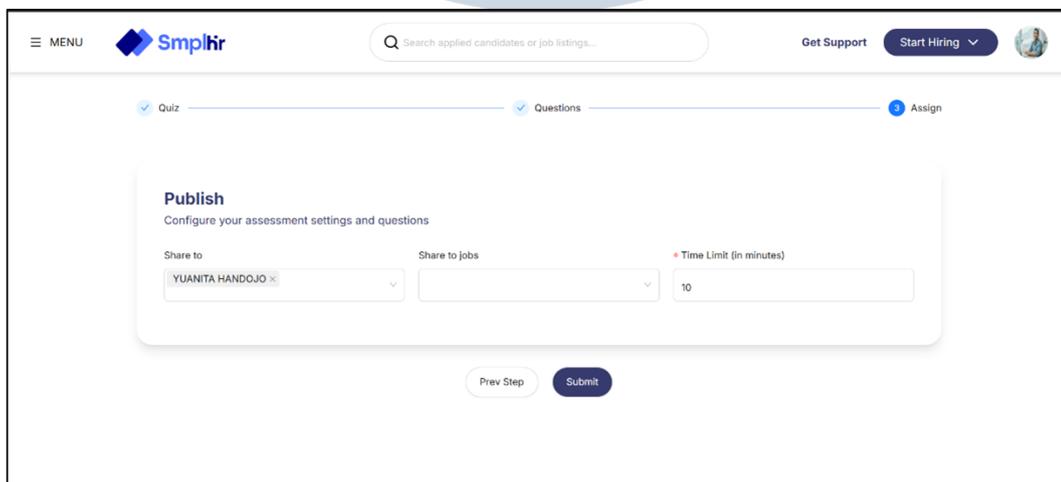
Hired	Total Applicant	Active Jobs	Interview	Total Candidate
10	173	41	16	173

Gambar 3.11. Tampilan setelah Penambahan Cursor Pointer

Setelah dilakukan pengembangan, seperti pada Gambar 3.11, kursor pada elemen yang dapat diklik diubah menjadi berbentuk tangan (pointer) ketika diarahkan ke area tersebut. Perubahan ini memberikan kejelasan bahwa elemen tersebut dapat diklik.

D.3 Fitur Autofill

Gambar 3.12 berikut menunjukkan kolom “Share To” yang otomatis terisi nama kandidat saat membuat assessment dari halaman kandidat.



Publish
Configure your assessment settings and questions

Share to:

Share to jobs:

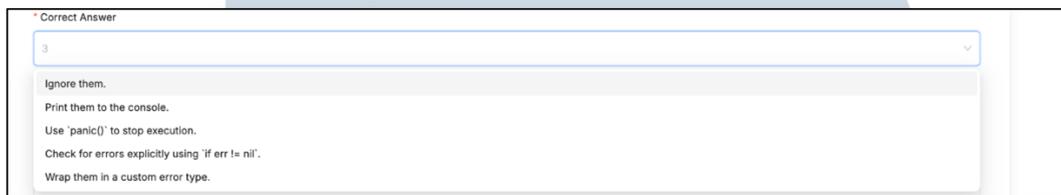
Time Limit (in minutes):

Gambar 3.12. Penambahan Fitur Autofill

Sebelum dilakukannya pengembangan, kolom “share to” pada halaman tersebut masih harus diisi manual oleh pengguna. Setelah dilakukan penyesuaian, bisa dilihat pada Gambar 3.12, secara otomatis sistem mengisi nama kandidat yang sedang dibuka ke dalam kolom penerima assessment.

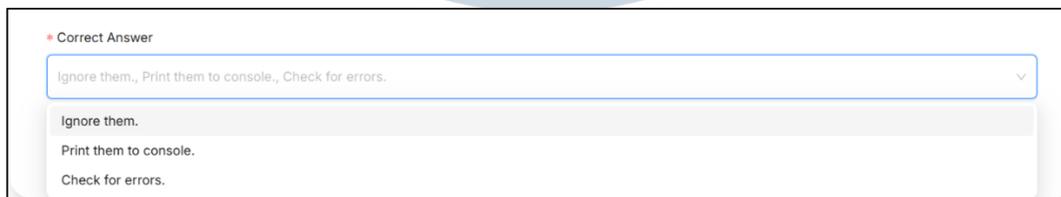
D.4 Perubahan Nilai Correct Answer

Pada awalnya yang bisa dilihat pada gambar 3.12, data correct answer yang ditampilkan masih menggunakan format JSON berupa angka, yang merepresentasikan indeks pilihan jawaban. Format ini kurang efektif bagi pengguna karena mereka harus mencocokkan angka tersebut dengan pilihan jawaban secara manual, hal ini dapat menimbulkan kebingungan.



Gambar 3.13. Tampilan sebelum Perubahan Nilai

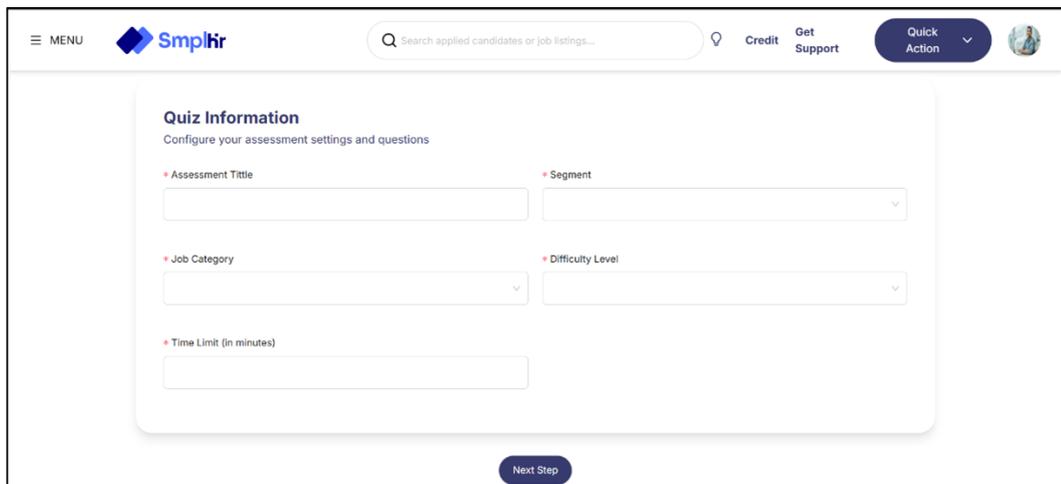
Setelah dilakukan pengembangan yang dapat terlihat di Gambar 3.14, nilai *correct answer* ditampilkan langsung ke dalam teks jawaban. Perubahan ini membuat pengguna dapat memilih jawaban tanpa mencocokkan dengan indeks jawaban.



Gambar 3.14. Tampilan setelah Perubahan Nilai

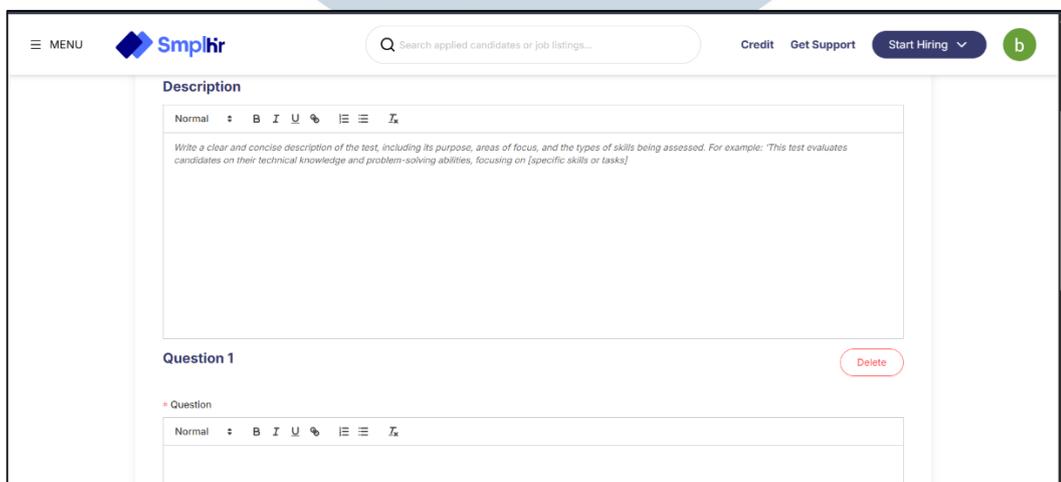
D.5 Penambahan Fitur Create Assessment

Pada Gambar 3.15 di bawah ini menunjukkan page *assessment create* setelah pengguna memilih untuk buat assessment melalui modal di halaman candidate sebelumnya. Di page ini pengguna dapat mengisi judul assessment, memilih category job, segment, dan kesulitan assessmentnya, serta durasi pengerjaannya.



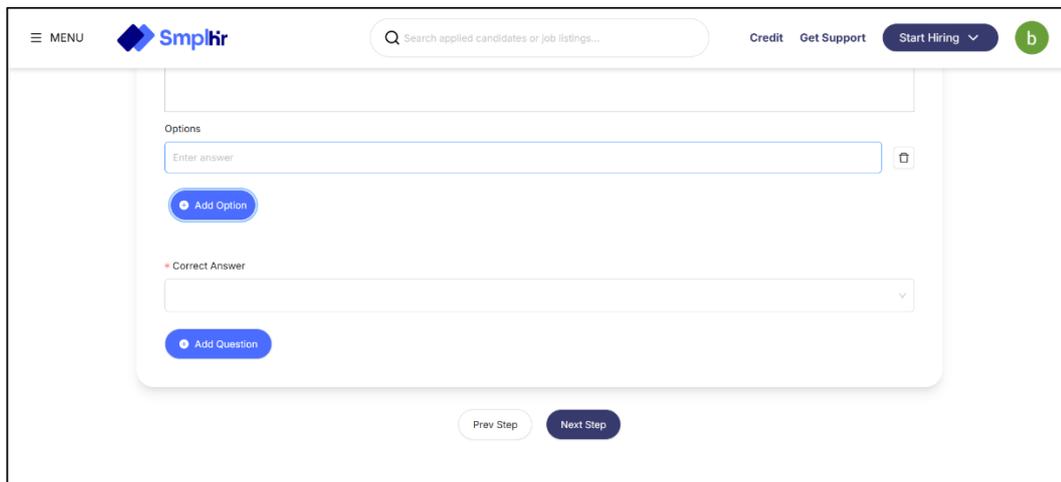
Gambar 3.15. Fitur Create Assessment

Pada Gambar 3.16 di bawah ini pengguna dapat mengisi deskripsi *assessment* ini tentang apa. Deskripsi ini berfungsi sebagai penjelasan mengenai topik, tujuan, dan konteks dari *assessment* tersebut, sehingga *candidate* dapat memahami gambaran umum sebelum dikerjakan.



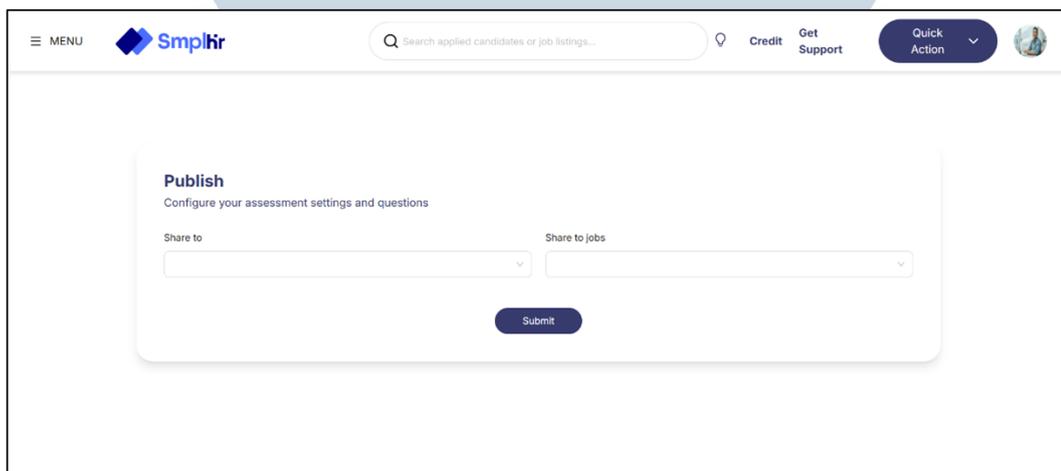
Gambar 3.16. Fitur Deskripsi

Pada Gambar 3.17 di bawah ini, digunakan pengguna untuk membuat pertanyaan beserta pilihan jawabannya.



Gambar 3.17. Fitur Pertanyaan Beserta Jawabannya

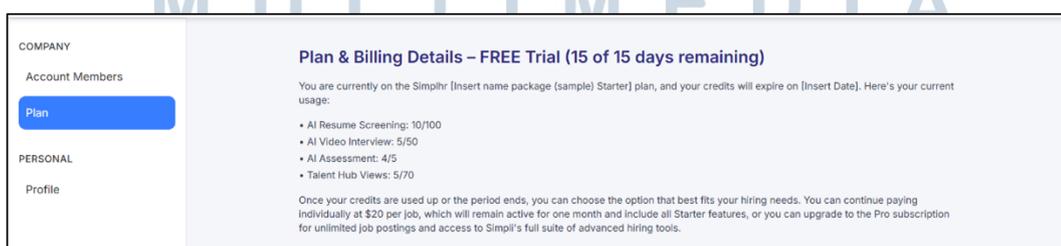
Pada Gambar 3.18, di bawah ini pengguna dapat mengassign *assessment* yang telah dibuat tadi berdasarkan *candidate* yang dipilih dan jenis job nya.



Gambar 3.18. Fitur Assign Assessment

D.6 Penambahan Fitur Payment integration

Berikut merupakan hasil penambahan halaman payment integration yang memungkinkan pengguna memilih paket layanan



Gambar 3.19. Fitur Informasi Langgan

	<input checked="" type="radio"/> Billed monthly <input type="radio"/> Billed for 3 months <input type="radio"/> Billed yearly		
Compare plans <small>Choose your workspace plan according to your organisational plan</small>	Starter <small>pay-as-you-go</small> \$20 <input type="button" value="Buy"/>	Pro \$119 /Month <input type="button" value="Buy"/>	Enterprise Custom <input type="button" value="Contact sales"/>
Active job post	Up to 3 jobs <small>Active for 1 month</small>	Unlimited	Unlimited
AI resume screening	Up to 10 CVs <small>Active for 1 month</small>	Unlimited	Unlimited
AI video interview	Up to 10 candidates <small>Active for 1 month</small>	Unlimited	Unlimited

Gambar 3.20. Daftar Pilihan Paket

Pada Gambar 3.19, menampilkan informasi langganan yang sedang dipakai pengguna, termasuk jumlah penggunaan fitur-fitur. Informasi ini bertujuan untuk menginformasikan pengguna terkait batas penggunaan layanan yang tersedia dalam paket langganan yang aktif saat ini.

Sementara itu pada Gambar 3.20, menampilkan daftar pilihan paket langganan dalam bentuk tabel, yakni *starter*, *pro*, dan *enterprise*. Masing-masing paket memiliki kelebihan dan batasan fitur yang berbeda, sehingga pengguna dapat memilih paket yang paling sesuai dengan kebutuhan rekrutmen mereka.

3.4 Testing

Pengujian dilakukan secara manual dengan mencoba secara langsung seluruh fitur yang telah dikembangkan. Setiap fitur diuji berdasarkan skenario penggunaan umum untuk memastikan bahwa alur interaksi berjalan sesuai yang diharapkan dan tidak menimbulkan *error* pada sisi tampilan. Pengujian ini menggunakan metode *blackbox testing*, yaitu pengujian yang berfokus pada fungsi dan output sistem tanpa melihat struktur internal atau kode sumbernya [7]. Hasil pengujian dapat dilihat pada tabel 3.3.

Tabel 3.3. Hasil Pengujian Blackbox Pada Aplikasi

No	Fitur yang diuji	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Modal Share Link	Modal muncul setelah menekan tombol “Share” pada halaman company page	Modal muncul dengan link yang dapat disalin	Berhasil
2	Cursor Pointer pada Elemen Interaktif	Kursor berubah menjadi pointer saat diarahkan ke elemen yang dapat diklik	Kursor berubah menjadi pointer sesuai elemen interaktif	Berhasil
3	Auto-Fill Kolom “Share To”	Nama kandidat terisi otomatis saat create assessment dari halaman kandidat	Kolom terisi otomatis sesuai nama kandidat yang dipilih	Berhasil
4	Tampilan Correct Answer dalam Format Teks	Jawaban benar ditampilkan dalam bentuk teks, bukan angka indeks	Teks jawaban benar ditampilkan sesuai pilihan yang telah dibuat	Berhasil
5	Pembuatan Assessment	Pengguna dapat membuat assessment dan mengirim ke kandidat	Form assessment dapat diisi dan dikirim ke kandidat	Berhasil
6	Integrasi Halaman Payment Integration	Pengguna diarahkan ke halaman pembayaran setelah memilih paket langganan	Redirect berjalan dan halaman Xendit muncul sesuai paket yang dipilih	Berhasil

3.5 Kendala yang Ditemukan

Selama proses pengembangan, salah satu kendala utama yang dihadapi adalah belum terbiasa dalam menggunakan *React*, yang merupakan framework utama dalam pengembangan sistem. Hal ini menyebabkan proses implementasi beberapa fitur menjadi lambat, karena membutuhkan waktu untuk memahami alur kerja komponen yang saling terhubung. Selain itu, proses pengembangan juga sempat terhambat akibat ketidakcocokan *library* dan dependensi yang digunakan dalam proyek, yang menyebabkan beberapa komponen tidak dapat dijalankan dengan semestinya atau menimbulkan error saat proses build. Di lain sisi, implementasi awal dari beberapa tampilan terkadang tidak sepenuhnya sesuai dengan desain yang telah ditentukan.

3.6 Solusi atas Kendala yang Ditemukan

Untuk mengatasi kendala tersebut, dilakukan pembelajaran secara mandiri dan mendalam mengenai *React*. Selain itu, dilakukan juga penyesuaian versi *library* agar sama dengan konfigurasi proyek saat ini, Penyesuaian tampilan juga dilakukan secara bertahap berdasarkan feedback dari supervisor, dengan tujuan agar hasil akhir sesuai dengan rancangan desain yang diharapkan.

